Local Neighbourhood Extension of SMOTE for Mining Imbalanced Data

Tomasz Maciejewski and Jerzy Stefanowski
Institute of Computing Science
Poznań University of Technology
60-965 Poznań, Poland
tomek.maciejewski@gmail.com, jerzy.stefanowski@cs.put.poznan.pl

***Abstract*—In this paper we discuss problems of inducing classifiers from imbalanced data and improving recognition of minority class using focused resampling techniques. We are particularly interested in SMOTE over-sampling method that generates new synthetic examples from the minority class between the closest neighbours from this class. However, SMOTE could also overgeneralize the minority class region as it does not consider distribution of other neighbours from the majority classes. Therefore, we introduce a new generalization of SMOTE, called LN-SMOTE, which exploits more precisely information about the local neighbourhood of the considered examples. In the experiments we compare this method with original SMOTE and its two, the most related, other generalizations Borderline and Safe-Level SMOTE. All these pre-processing methods are applied together with either decision tree or Naive Bayes classifiers. The results show that the new LN-SMOTE method improves evaluation measures for the minority class.**

I. Introduction

Some real-life data mining problems involve learning classifiers from *imbalanced data*, which means that one of the classes (further called a *minority class*) includes much smaller number of examples than the others (further referred to as *majority classes*). Typical such problems are medical diagnosing dangerous illness, analysing financial risk, detecting oil spills in satellite images, predicting technical equipment failures or information filtering [1], [2]. Class imbalance constitutes a difficulty for most learning algorithms, which are biased toward learning and recognition of the majority classes. As a result, minority examples tend to be misclassified.

Learning from imbalanced data has received growing research interest in the last decade and several specialized methods have been proposed (see [2], [3] for a review). In this paper we are interested in pre-processing methods on the data level. They are classifier-independent and consist in transforming an original data distribution to change the balance between classes. The simplest re-sampling techniques are random *over-sampling* which replicates examples from the minority class and random *under-sampling* which randomly eliminates examples from the majority classes until a required degree of balance between classes is reached. However, random under-sampling may potentially remove some important examples and simple over-sampling may also lead to overfitting. Thus, *focused methods* like SMOTE [1], one-side-sampling [4], NCR [5] or SPIDER [6], which attempt on taking into account internal characteristics of regions around minority class examples were introduced and experimentally verified.

The most popular among them is SMOTE [1], which considers each example from the minority class and generates new synthetic examples along the lines between it and some of randomly selected its k nearest neighbours from the minority class. Although experiments confirmed its usefulness [1], [7], some of assumptions behind this technique could be still questioned. In its generalization called Borderline-SMOTE [8] authors focused their attention on oversampling around examples located in the borderline between classes. It is also claimed that SMOTE may overgeneralize the minority class region without considering a distribution of neighbours from the majority classes [6], [9]. A new technique, called Safe-Level-SMOTE, has been just introduced to solve it [9]. However, we think that both solutions may be still unsatisfactory with respect to *local neigbourhood* of examples and therefore we propose a new generalization called LN-SMOTE.

The main aim of our paper is to introduce and to experimentally evaluate the LN-SMOTE method. Our additional aim is to carry out an extensive comparative study of the proposed method and the two most related generalizations: Borderline and Safe-Level SMOTE, as previously they were studied on few data sets only [8], [9].

II. Related Works

We discuss the most related focused methods only; for more extensive reviews see, e.g, [2], [3].

Several authors showed the simple uninformed random under-sampling or over-sampling were not sufficiently good at improving recognition of imbalanced classes. In particular, random over-sampling simply adds copies of the minority examples which makes some sub-regions of the minority class very specific leading to overfitting while learning classifiers.

SMOTE (Synthetis Minority Oversampling TEchnique) was proposed by Chawla et al. to overcome this problem by a special approach to generate new synthetic examples [1]. As the authors said, this method generates artificial examples based on the feature space similarities between original examples of the minority class. Its main idea is to take each example of the minority class and to introduce synthetic examples along the lines between it and its selected nearest neighbours also from the minority class. While looking for these nearest neighbours,

978-1-4244-9925-0/11/$26.00 ©2011 IEEE

the distance is calculated with the Euclidean distance metric for numerical features and the Value Distance Metric [10] for the qualitative features.

More precisely, let the training set S contain examples from the minority class P and other classes N. For each example $p_i \in P$ find its k nearest neighbours x from class P. Depending on the other parameter of this method – the amount of over-sampling – a given number of examples from these k nearest neighbours is randomly selected. Synthetic minority class examples are generated in the direction of each. For numerical features the new synthetic example is constructed as follows: compute the difference between features describing the example p_i and x – one of the selected k-nearest neighbours; multiply this feature vector difference by δ – a random number between 0 and 1; and add it to the feature vector p_i creating a new vector $x_{new} = p_i + (x - p_i) \cdot \delta$. For qualitative features create a new example with the most common feature values among k nearest neighbours. This technique of over-sampling generalizes the decision regions for the minority class. As a result, larger and less specific regions of this class are learned without causing overfitting. This should help learned classifier to better generalize. Experiments carried out in [1] with C4.5 trees, Ripper rules and Naive Bayes classifiers showed that SMOTE improved recognition of the minority class. Moreover, its combination with under-sampling the majority class can achieve better results than other under-sampling methods – see e.g. [1], [7].

Although SMOTE proved to be successful in these experiments it also has some shortcomings, which we further discuss. Firstly the way of identifying minority examples to be seeds for over-sampling could be problematic. In SMOTE all examples from this class are considered. However, they are not equally important for learning classifiers. In particular, it concerns examples at the *decision border* between classes and the ones located nearby as they are more liable to be misclassified while examples located inside the class region may be easier to be learned. For instance, edited approaches to instance based learning focus on *borderline examples* while some safer examples could be discarded. One-side-sampling [4] also differently processes examples depending on their type. Moreover, in the selective filtering method SPIDER [6] borderline examples are over-sampled in larger amount than the ones located inside the minority class.

Han et al. introduced a method Borderline-SMOTE where only the borderline examples of the minority class are over-sampled [8]. These examples are identified in the following way. For each example $p \in P$ the set of its k nearest neighbours is determined. Among these neighbours a number of examples from the majority class is calculated (denoted as $SN(p,k)$). Finally, the borderline examples (called DANGER in [8]) are those p that satisfy formula $k/2 \leq SN(p,k) < k$. Other examples from the minority class are treated as safe if $0 \leq SN(p,k) < k/2$ or as noise if $SN(p,k) = k$. It is assumed that it is not necessary to strengthen regions around the safe minority examples (as they are already well enough recognized) and noise examples (all examples around them are in the majority class). The DANGER examples are only fed to SMOTE for generating synthetic examples around them.

Two versions of Borderline-SMOTE were proposed in [8]. The first version generates synthetic examples from each example in the DANGER set and its nearest neighbours belonging to the minority class P only (as in original SMOTE). Borderline-SMOTE2 considers also the nearest neighbours from the majority classes N. Then, the difference between the minority examples and its nearest majority class neighbour is multiplied by a random number between 0 and 0.5 - so the synthetic example is located closer to the minority class. Experiments with four data sets from UCI showed that Borderline methods used with C4.5 trees improved sensitivity and F measure for the minority class over the original SMOTE and simple random over-sampling. The Borderline-SMOTE2 achieved better value of sensitivity than its first version.

Another shortcoming of SMOTE is the *overgeneralization* problem as it blindly generalizes the regions of the minority class without regard to the majority class. This strategy is particularly problematic in the case of skewed class distribution where the minority class is very sparse with respect to the majority class. In such a case SMOTE generation of synthetic examples may increase the occurrence of overlapping between classes. So, some adaptive strategies have been proposed to overcome this limitation. In [11] Japkowicz and Wang proposed ASMO method which included testing for data sparsity, special clustering of the minority class and synthetic sample generation using both minority and majority classes. Similar but more general, idea which took into account data decomposition was presented in [12].

In this paper we focus on the newest proposal called Safe-Level-SMOTE [9] as it is directly related to our approach. In this method the presence of the majority examples is taken into account before generating synthetic examples by calculating a special coefficient called a *safe level*. For each minority class example, it is defined as the number of other minority class examples among its k nearest neighbours. If that value is equal or close to 0, given example is interpreted as noise. On the other hand, if it is closer to k, then this example could be located in a safe region of the minority class. The key idea is to direct generation of new synthetic examples closer to the safer regions. More precisely, let p be the minority class example being a seed for over-sampling, then k nearest examples also belonging to the minority class P are determined. As in the original SMOTE, at least one of these neighbours is selected – it is denoted as n. For both examples p and n their k nearest examples in the full training data S are found to calculate their safe levels denoted as $sl(p)$ and $sl(n)$ respectively. Given them, the *safe level ratio* coefficient is defined as *sl-ratio* $= sl(p)/sl(n)$. The rest of the method goes along the 5 cases:

1) If $sl(p) = 0$ and $sl(n) = 0$, both examples p and n are treated as noisy outliers and no synthetic example is generated.
2) If $sl(p) > 0$ and $sl(n) = 0$, then n is interpreted as noise. The synthetic example will be generated far from n just by duplicating p.

3) If $sl\text{-}ratio = 1$, both p and n have similar nature of neighbours and the new synthetic example will be generated along the line joining them in the same way as in the original SMOTE.
4) If $sl\text{-}ratio > 1$, then p is located in safer minority region than n and the new synthetic example will be generated closer to p, i.e. δ parameter in SMOTE will be generated in the range $[0, 1/sl\text{-}ratio]$.
5) If $sl\text{-}ratio < 1$, then it is contrary to above point and the new example will be generated in the range $[1 - sl\text{-}ratio, 1]$.

As in case (1) no synthetic example is generated, it is possible that overall oversampling ratio might be slightly less than the required level. However, it is justified as it happens for two noisy examples only – between them no new examples should be added.

Experiments with C4.5 and Naive Bayes on two UCI data shown that this method outperformed both SMOTE and Borderline-SMOTE [9].

Finally, we notice other interesting extensions of SMOTE as SMOTEBoost, combinations with data cleaning techniques, LLE-based SMOTE or Surrounding-SMOTE. However, as they are not strictly related to our approach we skip their presentation; see, e.g. [2] for more details.

III. Analysing Local Neighbourhood in Extended SMOTE

In section II we notice that the original SMOTE did not take into account the distribution of examples from the majority class. Although the Safe-Level SMOTE uses such information, it may be still insufficient to overcome this problem. In particular, if the minority class is decomposed into several sub-regions of rather small cardinalities. This situation refers to problem of *small disjuncts* and is claimed to be a more important source of difficulty for learning classifiers from imbalanced data than the imbalance ratio itself [13].

For example, let us consider the situation where two rather rare subgroups of the minority class are surrounded by majority class examples – see Figure 1. They are distant enough from other examples from the minority class. Let the seed be an example in the lower subgroup. If the parameter k of the method is greater than the number of other minority class examples inside this subgroup (e.g $k = 5$), then the next minority class neighbours will be examples from the other subgroup. If the safe level ratios of examples from both subgroups could be similar, the new synthetic examples could be generated along the line joining the examples from these groups. So, it could be still located inside the area occupied by examples from the majority class (see the star mark in Figure 1). Thus, this strategy may still lead to overlapping between classes. In our opinion, similar increasing of overlapping may also occur for seed minority examples located in wider borders areas between classes [14].

The above undesirable situation results from the SMOTE strategy for looking k nearest neighbours that belong to the minority class only. If the seed example is not located in the dense area of this class, then some of these neighbours could be rather distant with respect to other majority class examples also surrounding this seed. We think that considering more *local neighbourhood* of the seed minority example could give better approximation of presence majority class examples. So, looking for too distant examples should be avoided. By the local neighbourhood we understand the typical $k\text{-}NN$ paradigm, i.e. determining k really nearest neighbours in the training set, including also the majority class ones. As a consequence the synthetic example could be generated between examples belonging to two different classes. We call our generalization LN-SMOTE emphasizing this consideration of the local neighbourhood.

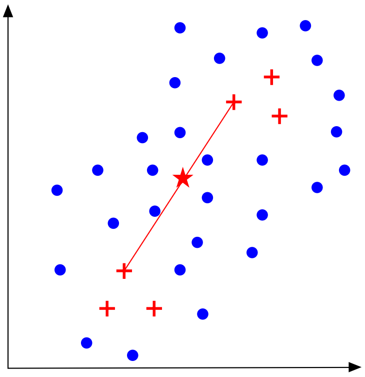

Fig. 1. Problem of overgeneralization when the minority class is decomposed

We also want to use and adapt the idea of local safe levels to consider presence of other majority neighbours. However, staying with the original Safe-Level strategy is not sufficient if the other example n belongs to the majority class. For example, let us assume that seed example p is an outlier and its selected majority class neighbour n does not have any other neighbours from the minority class. Safe level $sl(p) = 0$, however $sl(n) = 1$ due to presence of this example p in the local neighbourhood of n. The safe level ratio is 0 and following case (5) the synthetic example is generated exactly in the position of n, which introduces inconsistency in data.

Therefore, we decided to modify the way of calculating safe levels for the majority class neighbours. If the seed example p is identified within k nearest neighbours of n, it is not included into the $sl(n)$ but we look for the next $k + 1$ neighbour (see the pseudocode of LN-SMOTE).

Yet another problem with generating synthetic example between the minority seed p and its majority neighbour n is to direct its position rather closer to the minority class than to the majority class represented by n. Similarly to solutions from Borderline-SMOTE2 and SPIDER methods we want to restrict the range of interval where the new example could be randomly located. So, in some cases of safe levels we do not consider right boundary of the interval as 1 but as threshold $\tau < 1$ (see function RANDOMGAP). However, unlike in the above methods the threshold value is not fixed but it is determined dynamically depending on the safe level of considered majority example. If $sl(n)$ is relatively low, it means that n is surrounded by many examples from the majority class. So, a new example should be placed rather closer to the p. If n is surrounded by reasonable amount of minority examples, so the value of $sl(n)$ is higher, a new example could be located closer to n. It allows us to control the level of minority class expansion in a dynamic way, taking into account local distribution of examples. LN-SMOTE pseudo-code is given below.

```
function LN-SMOTE
    $SEEDS \leftarrow \{x : x \in S \wedge \textsc{Class}(x) = P\}$
    $OUT \leftarrow S$                          ▷ copy input dataset to results
    for $i \leftarrow 1 \ldots |SEEDS|$ do
        $p \leftarrow x_i \in SEEDS$                ▷ choose the seed
        $NN \leftarrow k$ nearest neighbours of $p$
        for $j \leftarrow 1 \ldots o$ do             ▷ oversampling ratio
            $s \leftarrow \textsc{CreateSynthetic}(p, NN, OUT)$
        end for
    end for
    return $OUT$
end function

function $\textsc{CreateSynthetic}(p, NN, OUT)$
    $n \leftarrow$ randomly selected nearest neighbour from $NN$
    if $\textsc{CanCreate}(p, n)$ then
        $x_{new} \leftarrow \textsc{Clone}(p)$
        for all $a \in \textsc{Attributes}(S)$ do
            if $\textsc{IsQuantitative}(a)$ then
                $\delta \leftarrow \textsc{RandomGap}(p, n)$
                $diff \leftarrow n(a) - p(a)$
                $x_{new}(a) \leftarrow p(a) + \delta \cdot diff$
            else if $\textsc{IsQualitative}(a)$ then
                $x_{new}(a) \leftarrow \textsc{MostFrequent}(p \cup NN, a)$
            end if
        end for
        add $x_{new}$ to $OUT$ set
    end if
end function

function $\textsc{CanCreate}(p, n)$    ▷ checks if $p$ and $n$ can be used to create new example between them
    $slp \leftarrow \textsc{SafeLevel}(p)$
    $sln \leftarrow \textsc{SafeLevel}(n, p)$
    return $slp \neq 0$ or $sln \neq 0$
end function

function $\textsc{SafeLevel}(p)$
    $neighbours \leftarrow k$ nearest neighbours of $p$
    return $|x : x \in neighbours \wedge \textsc{Class}(x) = P|$
end function

function $\textsc{SafeLevel}(n, p)$
    $neighbours \leftarrow k$ nearest neighbours of $n$
    if $\textsc{Class}(n) \neq P$ and $p \in neighbours$ then
        replace $p$ in $neighbours$ by $k + 1$ neighbour of $n$
    end if
    return $|x : x \in neighbours \wedge \textsc{Class}(x) = P|$
end function

function $\textsc{RandomGap}(p, n)$    ▷ returns $\delta$ based on safe level of $p$ and $n$
    $slp \leftarrow \textsc{SafeLevel}(p)$
    $sln \leftarrow \textsc{SafeLevel}(n, p)$
    $\delta \leftarrow 0$
    if $sln = 0$ and $slp > 0$ then
        return $\delta$
    else
        $sl\text{-}ratio \leftarrow \frac{slp}{sln}$
        if $sl\text{-}ratio = 1$ then
            $\delta = \textsc{Random}(1)$
        else if $sl\text{-}ratio > 1$ then
            $\delta = \textsc{Random}(\frac{1}{sl\text{-}ratio})$
        else
            $\delta = 1 - \textsc{Random}(sl\text{-}ratio)$
        end if
    end if
    if $\textsc{Class}(n) \neq P$ then
        $\delta = \delta \cdot \frac{sln}{k}$
    end if
    return $\delta$
end function
```

Moreover, we consider yet another version of our method, called LN-SMOTE-2. Following good experiences with combination of over-sampling with under-sampling (see e.g. [7], [1], [6]) we remove difficult noisy examples from the majority class in the first step before applying LN-SMOTE to the modified data. These examples are ones having only minority examples within their local 3 nearest neighbours.

We checked also other strategies of undersampling of the majority class, i.e. ENN and Tomek links [7], but the strategy described above gave the best results.

IV. Experiments

The aim of experiments is to compare the new proposed LN-SMOTE with related methods: the original version of SMOTE, Borderline-SMOTE and Safe-Level-SMOTE. We prepared our own implementation of these methods in Java using classes from the WEKA environment[1]. All these pre-processing methods are combined with two algorithms for inducing classifiers: the decision trees following Quinlan's C4.5 (Java implementation available in WEKA under the name J4.8) and Naive Bayes (also coming from WEKA). We chose them as decision trees are known to be sensitive to class imbalance and they were often used in studies of SMOTE and their extensions [7], [9], [1], [8]. Naive Bayes was also considered in many of above studies as another classifier. It is said to be less sensitive to imbalance. We also considered a basic approach with classifiers induced directly

[1] WEKA is available at http://www.cs.waikato.ac.nz/ml/weka

TABLE I
CHARACTERISTICS OF EVALUATED DATA SETS (ATT - NUMBER OF ATTRIBUTES, CLASS - IDENTIFIER OF THE MINORITY CLASS, MIN AND MAJ NUMBER OF EXAMPLES IN CLASSES)

Dataset	ATT	CLASS	MIN	MAJ	IMB(%)
Balance scale	4	B	49	576	7.84
Breast cancer	9	recurrence	85	201	29.72
Cleveland	13	3	35	268	11.55
CMC	9	long term	333	1140	22.61
Ecoli	7	imU	35	301	10.42
Flags	28	white	17	177	8.76
German credit	20	bad	300	700	30.00
Haberman	3	die	81	225	26.47
Hepatitis	19	die	32	123	20.65
Pima	8	positive	268	500	34.90
Postoperative	8	home	24	66	26.67
Solar flare	12	F	43	1023	4.03
Transfusion	4	donated	178	570	23.80
Yeast	8	ME2	51	1433	3.44

TABLE II
CONFUSION MATRIX FOR PERFORMANCE EVALUATION

	Predicted Positive	Predicted Negative
True Positive	TP	FN
True Negative	FP	TN

TABLE III
PRESENCE OF *danger* AND *noisy* EXAMPLES.

Dataset	MIN	D3	D5	D7	N3	N5	N7
Balance scale	49	0	4	18	49	45	31
Breast cancer	85	50	56	63	23	11	6
Cleveland	35	13	19	25	22	16	10
CMC	333	207	225	232	94	60	43
Ecoli	35	20	20	21	7	5	3
Flags	17	10	11	11	7	5	5
German credit	300	191	221	228	71	32	14
Haberman	81	51	53	57	20	14	8
Hepatitis	32	21	22	20	6	4	4
Pima	268	169	165	161	42	23	14
Post-operative	24	13	19	22	11	5	2
Solar flare	43	25	28	31	18	14	12
Transfusion	178	97	100	103	41	27	20
Yeast	51	25	28	32	24	20	18

from imbalanced data without any pre-processing to obtain a kind of baseline for comparing various SMOTE versions.

Our experiments were carried out on 14 data sets coming from UCI repository. Their basic characteristics is listed in Table I. The imbalance ratio (IMB in Table I) is calculated as a ratio of the number of minority examples to the total number of examples in a data set. We chose these data as they are characterized by varying degree of imbalance and they were often used in related experimental studies. Data sets with higher imbalanced ratio, as Slovenia breast cancer, were also chosen as they contained many noisy or borderline minority class examples. Some of these data sets originally included more than two classes, however, to focus more on minority vs majority characteristics and to simplify calculations we decided to aggregate all majority classes into one. In our opinion this aggregation does not influence the work of compared algorithms.

Another issue is choosing evaluation measures. As the overall classification accuracy is biased towards the majority classes [2], in most of the studies on imbalanced data, measures defined for two-class classification are considered, where typically the class label of the minority class is called positive and the class label of the majority class is negative. The performance of the classifiers is presented in a confusion matrix as in Table II.

Following the literature, we choose two kinds of measures. Firstly, we will consider:

$$Precision = TP/(TP + FP)$$

$$Recall = TP/(TP + FN)$$

These measures are aggregated to F measure:

$$F = ((1+\beta)^2 \cdot Recall \cdot Precision)/(\beta^2 \cdot Recall + Precision)$$

In our experiments we set $\beta = 1$ given equal importance to both precision and recall. Another type of measure often used is the $G\text{-}Mean$ defined as a geometric mean:

$$G\text{-}Mean = \sqrt{Sensitivity \cdot Specificity},$$

where *Sensitivity* is defined in the same way as recall and the specificity characterizes recognition of the negative class as:

$$Specificity = TN/(TN + FP)$$

We also remark that ROC technique with calculating Area Under Curve (AUC) could be also applied to evaluate classifiers on imbalanced data. However, we stayed with F measure and G-mean as we focused our experiments mainly on the tree-based classifier which is deterministic one. All these measures were estimated by stratified 10-fold cross validation repeated 3 times. Each time examples in data sets were reordered using different seeds. As a result, each compared method was evaluated in 30 repetitions of train and test folds. In further tables we present mean values of evaluation measures calculated over these repetitions.

Finally, let us discuss tuning parameters of methods. As to tree classifiers we used standard options with choosing unpruned version to strengthen the minority class. For SMOTE methods we have two parameters: k – number of neighbours and o – amount of oversampling (expressing how many times the minority class should be increased by oversampling)[2]. In related papers on SMOTE and its extensions k was set to 5 and o was usually stepwise changed to 5. In our experiments we decided to study wider range of these parameters. In case

[2] In the paper introducing original version of SMOTE [3] it was expressed in percentage, e.g. 500%

TABLE IV
F-MEASURE FOR THE MINORITY CLASS FOR ALL COMPARED METHODS USED TOGETHER WITH C4.5 TREE CLASSIFIER

	None	SMO	BS1	BS2	SLS	LN1	LN2
BS	0.00	9.29	8.40	11.33	8.58	16.54	16.08
BC	39.83	43.83	43.02	44.37	45.15	43.83	45.64
CL	19.29	26.71	25.27	28.33	26.03	29.27	29.70
CM	40.81	41.64	42.05	44.16	41.64	44.95	45.94
EC	58.86	64.31	62.38	64.02	63.98	62.01	66.96
FL	30.89	44.51	41.35	42.68	43.15	39.46	42.03
GC	45.51	50.30	49.98	51.01	50.02	50.91	50.46
HA	30.36	43.70	41.84	43.58	40.08	44.56	42.59
HE	49.20	52.10	53.94	53.00	57.10	58.57	57.86
PI	62.05	65.51	65.68	65.61	65.02	65.13	65.06
PO	5.84	22.03	22.86	19.06	20.56	20.42	19.44
SF	28.79	27.84	28.85	29.93	28.68	31.60	33.08
TR	47.27	48.80	50.05	51.12	48.94	49.19	50.30
YE	35.02	39.64	42.23	42.02	40.07	41.39	42.58

TABLE V
G-MEAN FOR ALL COMPARED METHODS WITH THE TREE CLASSIFIER

	None	SMO	BS1	BS2	SLS	LN1	LN2
BS	0.00	23.36	23.93	27.58	21.10	43.93	41.21
BC	53.27	57.17	55.77	57.93	58.10	57.18	58.73
CL	31.46	45.89	38.84	45.86	41.41	46.25	50.04
CM	57.33	60.33	60.25	62.92	60.29	63.23	64.71
EC	73.28	83.77	76.89	84.26	80.71	80.67	82.88
FL	41.61	61.46	58.53	56.31	59.14	53.80	58.06
GC	58.83	63.00	62.70	63.55	62.81	63.57	63.11
HA	44.53	59.00	57.15	59.20	55.78	59.94	58.13
HE	63.17	66.43	66.21	67.75	69.81	70.05	69.97
PI	69.84	72.73	72.80	72.60	72.47	72.33	72.56
PO	8.23	26.44	26.25	26.47	25.86	27.52	26.67
SF	39.28	45.71	46.77	47.03	45.78	50.85	57.33
TR	60.79	63.18	65.53	65.53	63.71	63.95	63.67
YE	50.17	66.23	67.08	70.64	61.04	64.70	66.03

of k we tested the following values 3, 5 and 7. However, for o we decided to check more values of oversampling depending on data characteristics – for data with imbalanced ratio larger than 15% we tested following o values 1, 2, 3, 4, 5 and 6 while in case of data with the smaller imbalance ratio we additionally considered o values equal to 10, 15, 20 and 25. Moreover, we found an additional value which led to balancing cardinalities of both minority and majority classes. We prepared a batch procedure for testing all these combinations of k and o parameters for a given method and data. Among these results we chose this combination which led to the best value of F measure for the minority class (as it was also the main criterion in [9], [8]). For this combination of parameters we also calculated the remaining measures. As for each method this best configuration was found separately on each data set, the chosen values for o and k may vary between methods and data sets.

Before these experiments we performed a simpler analysis of data sets. For each of them we used $k-NN$ analysis of each minority class example and considered is as noisy or danger (borderline) as defined in Borderline SMOTE (see section II). Results are presented in Table III, where **Dk** denotes number of examples identified as danger with k neighbours and **Nk** corresponds to the number of noisy examples.

First of all, one can notice that all studied data sets are rather difficult with respect to classifier ability for recognizing the minority class. In particular, in balance scale data all minority class examples are noisy with $k = 3$ and nearly all for larger neighbourhood. Some other data sets, e.g. cleveland or yeast, contain more noisy examples than dangerous ones. The number of noisy examples is quite high comparing to the size of the minority class. Another interesting observation is that for the typical neighbourhood ($k = 3$ or partly 5) most of the data sets do not contain any safe example or their number is relatively small comparing to noisy or danger examples, see e.g. flags data where 17 examples are divided between 10 in borderline zone and 7 noisy ones – so there are no safe examples. Quite similar distribution occurs for solar flare with 43 examples from the minority class distributed as 25 danger and 19 noisy ones. In our opinion these values confirm that imbalance ratio is not the only source of difficulty but 'the nature' of examples makes the learning problems very difficult.

Then, in Table IV we present results of F value for all compared methods used with the tree classifier. We use the following notation: None - baseline without any pre-processing of input data, SMO – original SMOTE, BS – two versions of Borderline-SMOTE, SLS – Safe-Level SMOTE and LN denotes our proposed method, where LN1 is the basic oversampling with local neighbourhood and LN2 is its combination with undersampling . Moreover, here and in further tables we will use abbreviations of the full names of the data sets to keep the width of tables in formatting requirements. Values of sensitivity of the minority class and G-means are presented in Tables VI V, respectively[3].

In case of F measure the new introduced method LN-SMOTE achieved very good results. Results of the basic version of LN-SMOTE were the best among all compared methods for 3 of all 14 data sets (and 3 times it is the second in the order) and results of LN-SMOTE2 were the best for the next 7 data sets. One can also notice that Borderline SMOTE 2 was the next method with respect to the high F-values.

In order to globally compare performance of a pair of methods on all data sets we used the Wilcoxon Signed Ranks Test – a nonparametric test for significant differences between paired observations – see details of its calculations described in [15]. We considered all pairs of methods and for each of them we present in Tables VII and VIII p value for the calculated Wilcoxon statistics. If this value is smaller than the confidence level α = 0.05, the method from the column is superior to the method in the corresponding row.

Results of Wilcoxon test for F value clearly show that the

[3] All these values are presented in a range between 0 and 100

TABLE VI
SENSITIVITY OF THE MINORITY CLASS FOR ALL COMPARED METHODS WITH TREE CLASSIFIER

	None	SMO	BS1	BS2	SLS	LN1	LN2
BS	0	15.33	13.33	17.5	12.5	34.83	35.83
BC	37.5	44.72	42.82	47.18	45.83	47.31	50.69
CL	21.67	34.5	27.78	34.44	30	34.72	38.06
CM	39.76	49.58	47.67	54.67	48.86	53.8	59.21
EC	61.11	77.5	69.44	75.17	71.39	66.39	74.17
FL	38.33	55	55	48.33	53.33	46.67	51.67
GC	45	54.33	52.89	56.33	53.22	55.44	55.78
HA	26.76	48.89	45.28	51.44	43.24	53.89	47.96
HE	51.67	56.94	53.61	60.39	62.5	59.17	58.89
PI	60.8	74.87	76.48	77.49	72.88	75.38	72.62
PO	6.11	23.11	22.78	22.78	22.22	23.33	22.22
SF	21.67	29	29.5	29.83	29	33.67	39.5
TR	42.3	47.35	53.73	50.46	49.42	49.4	46.59
YE	31.89	52.33	51.67	53.67	42.33	46.56	48.44

TABLE VII
F-MEASURE WILCOXON TEST – p VALUES

	None	SMO	BS1	BS2	SLS	LN1	LN2
None	–	0.00	0.00	0.00	0.00	0.00	0.00
SMO	1.00	–	0.64	0.06	0.78	0.07	0.02
BS1	1.00	0.38	–	0.02	0.50	0.06	0.00
BS2	1.00	0.95	0.98	–	0.95	0.29	0.03
SLS	1.00	0.24	0.52	0.06	–	0.06	0.01
LN1	1.00	0.94	0.95	0.73	0.95	–	0.10
LN2	1.00	0.99	1.00	0.97	1.00	0.91	–

TABLE VIII
G-MEAN WILCOXON TEST – p VALUES

	None	SMO	BS1	BS2	SLS	LN1	LN2
None	–	0.00	0.00	0.00	0.00	0.00	0.00
SMO	1.00	–	0.43	0.06	0.23	0.07	0.04
BS1	1.00	0.60	–	0.03	0.23	0.20	0.03
BS2	1.00	0.95	0.98	–	0.97	0.89	0.50
SLS	1.00	0.79	0.79	0.04	–	0.12	0.07
LN1	1.00	0.85	0.87	0.25	0.87	–	0.47
LN2	1.00	0.97	0.98	0.52	0.89	0.56	–

TABLE IX
F-MEASURE FOR THE MINORITY CLASS FOR ALL COMPARED METHODS USED TOGETHER WITH NAIVE BAYES CLASSIFIER

	None	SMO	BS1	BS2	SLS	LN1	LN2
BS	0.00	14.07	13.07	13.35	14.12	14.26	14.15
BC	47.72	53.13	51.36	51.26	52.74	52.53	52.52
CL	35.58	43.24	42.24	42.33	42.65	44.54	44.01
CM	42.25	47.73	47.73	47.48	47.68	47.57	47.61
EC	62.23	61.96	63.59	64.80	63.91	61.08	60.91
FL	35.84	28.72	31.58	34.84	32.84	33.44	33.57
GC	54.46	61.47	61.93	61.91	61.74	62.07	61.85
HA	30.10	44.02	43.45	44.44	43.36	43.61	43.95
HE	63.69	68.74	69.15	69.87	69.50	66.73	68.66
PI	63.48	66.68	66.24	66.05	66.59	66.58	66.36
PO	4.44	21.82	20.25	26.45	21.48	25.89	22.57
SF	37.83	37.01	38.24	37.60	37.76	38.77	38.47
TR	27.44	48.32	48.73	48.83	48.53	48.75	48.72
YE	29.65	32.52	34.87	34.91	34.97	34.35	34.24

new proposed LN-SMOTE2 outperforms all other method. Then, the basic version LN-SMOTE1 is also very good, although its superiority is smaller with respect to this test. However, one can notice that difference between LN-SMOTE1 and the next method Borderline2 is not sufficiently significant. Finally, according to this test Borderline1 and SafeLevel (which are the next in an order with respect to F value) are not significantly better than the original SMOTE.

Looking at the sensitivity alone (Table VI) we can say that LN-SMOTE still gives good results, however, other methods also lead to high values. Even SMOTE achieved the good value of sensitivity for some data sets. However we can interpret it that the SMOTE and Borderline1 methods are focused on recognizing the minority class without paying enough attention to majority classes (as these methods also received worse G-means or F-values for these data). Wilcoxon test for G-means still shows that LN-SMOTE (in particular version 2) outperforms the many of other methods, however now there is no significant difference with Borderline2.

Summary of experiments with Naive Bayes are presented in Tables IX and X. Firstly, we should stress that this classifier performed better than the tree classifies as evaluation values are usually higher than in the previous tables. Moreover, differences between compared methods are smaller. LN-SMOTE is still a winner for many data set (but now rather its version LN1 is winning more times), however the Wilcoxon test showed that some of these differences were not significant (due to the page limit we have to skip presentation detailed tables). In particular, we noticed that Borderline2 gave very similar results and it was also a winner for some of remaining data. On the other hand, this generalization is also exploiting the closest majority neighbours similarly to our concept of the local neighbourhood, so we could say that this aspect led to better results than other methods. Considering SMOTE, Borderline1 and Safe-Level (which performed worse than local neighbourhood based methods) the Wilcoxon test showed no difference between their performance.

V. CONCLUSIONS

In this paper we studied SMOTE based oversampling methods. Following a critical discussion on shortcoming of the original version of SMOTE and its generalizations Borderline and Safe-Level we introduced a new method LN-SMOTE. We paid a particular attention to studying the local neighbours of the seed minority example. Moreover, in this method we adapt and modify the idea of calculating safe-levels describing the distribution of other examples in the nearest neighbourhood.

The LN-SMOTE was compared with original SMOTE and

TABLE X
G-MEAN FOR ALL COMPARED METHODS WITH THE NAIVE BAYES CLASSIFIER

	None	SMO	BS1	BS2	SLS	LN1	LN2
BS	0.00	0.83	3.45	2.43	2.78	0.39	0.28
BC	59.68	65.36	63.65	63.75	64.99	64.89	64.75
CL	59.07	75.03	70.69	72.69	71.06	72.87	77.08
CM	59.38	66.39	66.36	66.18	66.32	66.47	66.30
EC	86.34	87.50	88.00	89.92	87.55	87.65	87.59
FL	56.23	54.03	57.10	62.41	59.35	60.84	60.75
GC	65.10	72.27	72.37	72.62	72.09	72.67	72.67
HA	42.24	59.50	58.87	59.55	59.01	59.16	59.44
HE	75.65	78.81	78.69	80.70	78.85	78.03	80.60
PI	71.01	73.10	72.85	72.94	73.24	73.05	73.12
PO	4.71	29.52	27.68	33.85	29.69	33.68	29.60
SF	73.50	77.49	77.72	77.29	77.63	77.78	77.75
TR	42.15	64.21	65.46	65.45	65.51	65.43	65.32
YE	61.82	71.87	74.44	74.47	73.27	73.03	79.46

Borderline and Safe-Level, on several imbalanced data sets. The results, discussed in the previous section, show that the new method outperformed the compared methods with respect to F measure and G-means. The difference was higher for the tree classifier than for Naive Bayes. However, the Bayesian classifier generally better recognized the minority classes. Then, the Borderline2 was always the next well performing method. We can remark that Borderline2 also considers majority class examples inside k nearest neighbours, so this observation confirm the use of local neighbourhood instead of the typical oversampling SMOTE strategy.

Analysing results of two currently known SMOTE extensions, Borderline and Safe-Level, we can say that Safe-Level is not performing so well as it was reported in [9] – but this previous study was done on few data sets only. What is also surprising the differences between Borderline1 or Safe-Level methods and original SMOTE were not so high, as we could expect reading the literature. Indeed these differences were not significant with respect to the Wilcoxon test. Both these methods achived better results with Naive Bayes classifier, but performed worse with J48.

Another interesting observation is the level of applied oversampling in all these methods. For some data sets the best F values were obtained when the oversampling ratio o was greater than k (see balance scale with $k = 5$ and $o = 20$, or Ecoli with $k = 3$ and $o = 10$). Let us remind that it was recommended in [1] to tune o no higher than k. However, in our experiments for data sets with very high imbalance ratio, we noticed that the best results were obtained while the oversampling ratio was sometimes 4–5 times higher than the number of analyzed neighbours.

We also inform that we carried out additional experiments, where we looked in a different way for the best combinations of o and k. We chose o parameter leading to the best F value for the original SMOTE only (run with $k = 5$). Then, this combination was used for the other compared methods (all of them were run together with tree classifiers). The results of F value in these experiments still confirmed that LN-SMOTE performed better than other methods.

Yet another observation is that nearly all of studied data sets are highly noisy and contain many borderline examples without too many safe regions of the minority class. In future research we want to carry out experiments with artificial data where we could change and explore the impact of noisy, borderline or safe examples from the minority class on the classifier performance (see similar recent experiments with other re-sampling methods [14]).

Another future research could concern studying the new distance measure more specific for handling nominal attributes and trying to more automatically adapt oversampling amount depending on the distribution of examples around the seed minority class example.

Acknowledgments: The research has been supported by the Ministry of Science and Higher Education, grant no. N N519 441939.

REFERENCES

[1] Chawla, N., Bowyer, K., Hall, L., Kegelmeyer, W.: SMOTE: Synthetic Minority Over-sampling Technique. J. of Artifical Intelligence Research, 16, 2002, 341–378.

[2] He H., Garcia E.: Learning from imbalanced data. IEEE Transactions on Data and Knowledge Engineering, vol. 21 (9), 2009, 1263–1284.

[3] Chawla N.: Data mining for imbalanced datasets: An overview. In: Maimon O., Rokach L. (eds.) The Data Mining and Knowledge Discovery Handbook, Springer, 2005, 853–867.

[4] Kubat M., Matwin S.: Addresing the curse of imbalanced training sets: one-side selection. In: Proc. of the 14th Int. Conf. on Machine Learning, 1997, 179–186.

[5] Laurikkala J.: Improving identification of difficult small classes by balancing class distribution. Tech. Report A-2001-2, University of Tampere, 2001.

[6] Stefanowski J., Wilk Sz.: Improving rule based classifiers induced by MODLEM by selective pre-processing of imbalanced data. In: Proc. of the RSKD Workshop at the ECML/PKDD Conf., 2007, 54–65.

[7] Batista, G., Prati, R., Monard, M.: A study of the behavior of several methods for balancing machine learning training data. ACM SIGKDD Explorations Newsletter, vol. 6 (1), 2004, 20–29.

[8] Han H., Wang W., Mao B.: Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning. In Proc. ICIC, Springer LNCS 3644, 2005, 878–887.

[9] Bunkhumpornpat C., Sinapiromsaran K., Lursinsap C.: Safe-Level-SMOTE: Safe Level Synthetic Over-Sampling TEchnique for Handling the Class Imbalanced Problem. In: Proc. PAKDD 2009, Springer LNAI 5476, 2009, 475–482.

[10] Cost S., Salzberg S.: A Weighted Nearest Neighbor Algorithm for Learning with Symbolic Features. Machine Learning Journal, vol. 10 no. 1, 1993, 1213–1228.

[11] Wang, B.X., Japkowicz, N.: Imbalanced Data Set Learning with Synthetic Examples. Presented at the IRIS Machine Learning Workshop, Ottawa, June 9, 2004.

[12] Cieslak D.A., Chawla N.V.: Start globally, optimize locally, predict globally: Improving performance on imbalanced data. In: Proc. the 8th IEEE Int. Conference on Data Mining ICDM, 2009, 143–152.

[13] Jo T., Japkowicz N.: Class Imbalances versus small disjuncts. ACM SIGKDD Explorations Newsletter, vol. 6 (1), 2004, 40–49.

[14] Napierala K., Stefanowski J., Wilk Sz.: Learning from Imbalanced Data in Presence of Noisy and Borderline Examples. In. Proc. 7th Int. Conf. RSCTC 2010, Springer, LNAI vol. 6086, 2010, 158-167.

[15] Demšar, J., Statistical comparisons of classifiers over multiple data sets. The Journal of Machine Learning Research, vol. 7, 2006, 1–30.