Addressing Data-Complexity for Imbalanced Data-sets: A Preliminary Study on the Use of Preprocessing for C4.5

Julián Luengo
Dept. of Computer Science and A.I.,
University of Granada
julianlm@decsai.ugr.es

Alberto Fernández
Dept. of Computer Science and A.I.,
University of Granada
alberto@decsai.ugr.es

Francisco Herrera
Dept. of Computer Science and A.I.,
University of Granada
herrera@decsai.ugr.es

Abstract

In this work we analyse the behaviour of the C4.5 classification method with respect to a bunch of imbalanced data-sets. We consider the use of two metrics of data complexity known as "maximum Fishers discriminant ratio" and "nonlinearity of 1NN classifier", to analyse the effect of preprocessing (oversampling in this case) in order to deal with the imbalance problem.

In order to do that, we analyse C4.5 over a wide range of imbalanced data-sets built from real data, and try to extract behaviour patterns from the results. We obtain rules that describe both good or bad behaviours of C4.5 in the case of using the original data-sets (absence of preprocessing) and when applying preprocessing.

These rules allow us to determine the effect of the use of preprocessing an to predict the response of C4.5 to preprocessing from the data-set's complexity metrics prior to its application, and then establish when the preprocessing would be useful.

***Keywords**: C4.5, Classification, Data complexity, Imbalanced Data-sets, Oversampling.*

1. Introduction

This contribution is focused on the framework of imbalanced data-sets, also known as class imbalance problem, which refers to the case where one class, usually the one that contains the concept to be learnt (the positive class), is under represented in the data-set [7]. This issue is present in many real-world classification tasks and has been defined as a current challenge of the Data Mining community [15].

It is well-known that the prediction capabilities of classifiers are strongly dependent on the problem's characteristics. An emergent field, that uses a set of complexity measures applied to the problem to describe its difficulty, has recently arisen. These measures quantify particular aspects of the problem which are considered complicated to the classification task [3]. Studies of data complexity metrics applied to particular classification's algorithms can be found in [3, 6, 5, 14].

We are interested in analysing the relationship between the imbalanced data and two data complexity measures, which were computed using the original data-sets (not preprocessed). The first one is known as *maximum Fishers discriminant ratio*, which is based on the geometry, topology and density of manifolds. The second one is the *nonlinearity of 1NN classifier*, which measures the overlap in feature values from different classes. We have considered a well-known classifier, the C4.5 decision tree [13], which has been used in previous analysis of imbalanced data [4].

In order to deal with the problem of imbalanced data-sets balancing the distribution of training examples in both classes, we will make use of a preprocessing technique, the hybridization of the "Synthetic Minority Over-sampling Technique" (SMOTE) with the Wilson's Edited Nearest Neighbour Rule (ENN) [4]. SMOTE-ENN forms new minority class examples by interpolating between several minority class examples that lie together and then removes any example from the training set misclassified by its three nearest neighbours.

We have selected a large collection of data-sets with different degrees of imbalance from UCI repository [2] for developing our analysis. We have analysed the intervals of values of the two data complexity measures in which C4.5

performs good or bad, considering the use of SMOTE-ENN or not. We have formulated a rule for such intervals in both cases, comparing the support (number of data-sets included in the interval) and average AUC for the rules without preprocessing with those obtained using SMOTE-ENN. We can determine when the use of SMOTE-ENN is a necessity in order to obtain a good performance in the framework of imbalanced data-sets.

This contribution is organized as follows. First, Section 2 introduces two data-complexity measures used. Next, Section 3 presents the problem of imbalanced data-sets, describing its features and the metric we have employed in this context. Section 4 contains the experimental framework for the study and the analysis of the results for C4.5. Finally, Section 5 summarizes and concludes the work.

2. Data Complexity Measures

In this section we introduce the two metrics we have used in this contribution, first proposed in [9]. Their description is detailed below:

F1: maximum Fishers discriminant ratio. Fishers discriminant ratio for one feature dimension is defined as:

$$f = \frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2}$$

where μ_1, μ_2, σ_1^2, σ_2^2 are the means and variances of the two classes respectively, in that feature dimension. We compute f for each feature and take the maximum as measure F1. For a multidimensional problem, not all features have to contribute to class discrimination. The problem is easy as long as there exists one discriminating feature. Therefore, we can just take the maximum f over all feature dimensions in discussing class separability. F1 is a measure of *overlap in feature values from different classes*.

N4: nonlinearity of 1NN classifier. This is the nonlinearity measure, as defined by Linear Programming. Hoekstra and Duin [10] proposed a measure for the nonlinearity of a classifier with respect to a given data-set. Given a training set, the method first creates a test set by linear interpolation (with random coefficients) between randomly drawn pairs of points from the same class. Then the error rate of the classifier (trained by the given training set) on this test set is measured. Here we use such a nonlinearity measure for the linear classifier defined for the measure L1 (see [9]). In the case of N4, error is calculated for a nearest neighbour classifier. This measure is for the alignment of the nearest-neighbour boundary with the shape of the gap or overlap between the convex hulls of the classes. N4 is a measure of *geometry, topology and density of manifolds*.

3. Imbalanced Data-sets in Classification

Standard classification algorithms are usually biased towards the majority class trying to maximize the overall accuracy and their performance is poor on imbalanced data-sets. Thus these measures can lead to erroneous conclusions over imbalanced data-sets since they do not take into account the proportion of examples for each class. Therefore, in this work we use the AUC metric [11], defined as

$$AUC = \frac{1 + TP_{rate} - FP_{rate}}{2}, \qquad (1)$$

where TP_{rate} and FP_{rate} are the percentage of correctly and wrongly classified cases belonging to the positive class respectively.

Finally, we must stress the concept of the Imbalance Ratio (IR), which is defined as the ratio of the number of instances of the majority class and the minority class [12] and that has been used as a metric in this framework to categorise the data according to their degree of imbalance [8].

4. Experimental Study

In this study, our aim is to show the effect of the use of SMOTE-ENN in the characterisation of the behaviour of C4.5 in the scenario of imbalanced data-sets by means of the F1 and N4 data complexity measures.

In the remaining of this section, we will first present the experimental framework and all the parameters employed in this study in Subsection 4.1 and then we will show how the IR has little relationship with the behaviour of C4.5 in Subsection 4.2. The empirical study for C4.5 in imbalanced data-sets with the data complexity measures separately is presented in Subsection 4.3. In Subsection 4.4 we analyse the collective evaluation of the set of rules.

4.1. Experimental Set-Up

To carry out the different experiments we consider a *5-folder cross-validation model*, i.e., 5 random partitions of data with a 20%, and the combination of 4 of them (80%) as training and the remaining one as test. For each data-set we consider the average results of the five partitions.

We have selected forty-four data-sets from UCI repository [2]. The data are summarized in Table 1, showing the number of examples (#Ex.), attributes (#Atts.), name of each class (minority and majority), class attribute distribution and IR. For every binary data-set generated, we compute the F1 and N4 metrics before preprocessing.

As we state previously, the algorithm selected for our study is C4.5, which was run using KEEL software [1] following the recommended parameter values given in this platform, that is, a confidence level of 0.25, 2 minimum

Table 1. Summary Description for Imbalanced Data-Sets

Data-set	#Ex.	#Atts.	Class (min., maj.)	%Class(min.; maj.)	IR
Glass1	214	9	(build-win-non_float-proc; remainder)	(35.51, 64.49)	1.82
Ecoli0vs1	220	7	(im; cp)	(35.00, 65.00)	1.86
Wisconsin	683	9	(malignant; benign)	(35.00, 65.00)	1.86
Pima	768	8	(tested-positive; tested-negative)	(34.84, 66.16)	1.90
Iris0	150	4	(Iris-Setosa; remainder)	(33.33, 66.67)	2.00
Glass0	214	9	(build-win-float-proc; remainder)	(32.71, 67.29)	2.06
Yeast1	1484	8	(nuc; remainder)	(28.91, 71.09)	2.46
Vehicle1	846	18	(Saab; remainder)	(28.37, 71.63)	2.52
Vehicle2	846	18	(Bus; remainder)	(28.37, 71.63)	2.52
Vehicle3	846	18	(Opel; remainder)	(28.37, 71.63)	2.52
Haberman	306	3	(Die; Survive)	(27.42, 73.58)	2.68
Glass0123vs456	214	9	(non-window glass; remainder)	(23.83, 76.17)	3.19
Vehicle0	846	18	(Van; remainder)	(23.64, 76.36)	3.23
Ecoli1	336	7	(im; remainder)	(22.92, 77.08)	3.36
New-thyroid2	215	5	(hypo; remainder)	(16.89, 83.11)	4.92
New-thyroid1	215	5	(hyper; remainder)	(16.28, 83.72)	5.14
Ecoli2	336	7	(pp; remainder)	(15.48, 84.52)	5.46
Segment0	2308	19	(brickface; remainder)	(14.26, 85.74)	6.01
Glass6	214	9	(headlamps; remainder)	(13.55, 86.45)	6.38
Yeast3	1484	8	(me3; remainder)	(10.98, 89.02)	8.11
Ecoli3	336	7	(imU; remainder)	(10.88, 89.12)	8.19
Page-blocks0	5472	10	(remainder; text)	(10.23, 89.77)	8.77
Yeast2vs4	514	8	(cyt; me2)	(9.92, 90.08)	9.08
Yeast05679vs4	528	8	(me2; mit,me3,exc,vac,erl)	(9.66, 90.34)	9.35
Vowel0	988	13	(hid; remainder)	(9.01, 90.99)	10.10
Glass016vs2	192	9	(ve-win-float-proc; build-win-float-proc, build-win-non_float-proc,headlamps)	(8.89, 91.11)	10.29
Glass2	214	9	(Ve-win-float-proc; remainder)	(8.78, 91.22)	10.39
Ecoli4	336	7	(om; remainder)	(6.74, 93.26)	13.84
Yeast1vs7	459	8	(nuc; vac)	(6.72, 93.28)	13.87
Shuttle0vs4	1829	9	(Rad Flow; Bypass)	(6.72, 93.28)	13.87
Glass4	214	9	(containers; remainder)	(6.07, 93.93)	15.47
Page-blocks13vs2	472	10	(graphic; horiz.line,picture)	(5.93, 94.07)	15.85
Abalone9vs18	731	8	(18; 9)	(5.65, 94.25)	16.68
Glass016vs5	184	9	(tableware; build-win-float-proc, build-win-non_float-proc,headlamps)	(4.89, 95.11)	19.44
Shuttle2vs4	129	9	(Fpv Open; Bypass)	(4.65, 95.35)	20.5
Yeast1458vs7	693	8	(vac; nuc,me2,me3,pox)	(4.33, 95.67)	22.10
Glass5	214	9	(tableware; remainder)	(4.20, 95.80)	22.81
Yeast2vs8	482	8	(pox; cyt)	(4.15, 95.85)	23.10
Yeast4	1484	8	(me2; remainder)	(3.43, 96.57)	28.41
Yeast1289vs7	947	8	(vac; nuc,cyt,pox,erl)	(3.17, 96.83)	30.56
Yeast5	1484	8	(me1; remainder)	(2.96, 97.04)	32.78
Ecoli0137vs26	281	7	(pp,imL; cp,im,imU,imS)	(2.49, 97.51)	39.15
Yeast6	1484	8	(exc; remainder)	(2.49, 97.51)	39.15
Abalone19	4174	8	(19; remainder)	(0.77, 99.23)	128.87

item-sets per leaf and the application of pruning for the final tree.

4.2. Analysis of C4.5 behaviour based on the IR

In Table 2 we have summarized the global average Training and Test AUC and standard deviation obtained by C4.5 with and without SMOTE-ENN preprocessing.

Table 2. Global Average Training and Test AUC for C.45

	Global % AUC Training	Global % AUC Test
C4.5 without preprocessing	87.33% ± 13.89	79.29% ± 15.15
C4.5 with SMOTE-ENN preprocessing	94.38% ± 6.35	83.62% ± 13.09

We depict in Figure 1 the results for C4.5 in the case of preprocessing the 44 data-sets with SMOTE-ENN and not preprocessing them, sorting the data-sets by their IR value. We can observe that the good and bad results of C4.5 with and without SMOTE-ENN preprocessing is not related with the IR value, nor the improvements achieved with SMOTE-ENN. Therefore, the use of IR as a unique measure to characterise the improvement of the preprocessing is insufficient, and we need to use alternate measures to characterize such behaviour.

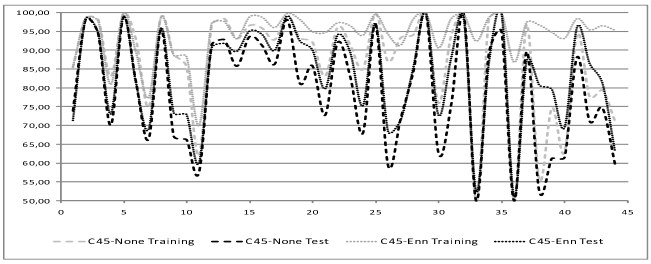

Figure 1. C4.5 without and with SMOTE-ENN AUC in Training/Test sorted by IR

4.3. Determination of Rules Based on C4.5 Behaviour with and without SMOTE-ENN

In Figures 2 and 3 the results for C4.5 with the original data-sets are depicted, whereas Figures 4 and 5 represent the results for C4.5 with SMOTE-ENN preprocessing.

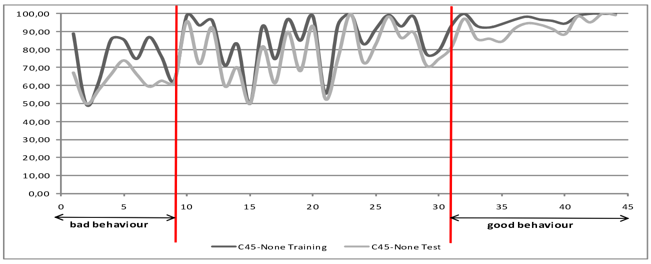

Figure 2. C4.5 without SMOTE-ENN AUC in Training/Test sorted by F1

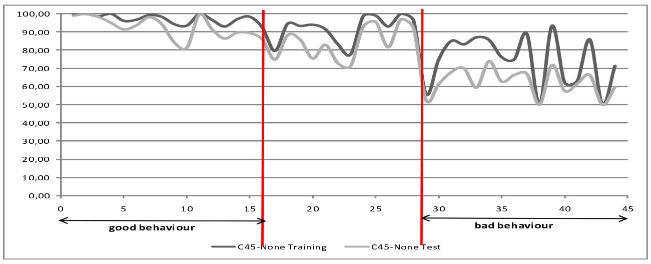

Figure 3. C4.5 without SMOTE-ENN AUC in Training/Test sorted by N4

In each figure the data-sets are sorted by the ascending value of the corresponding complexity measure. The X axis

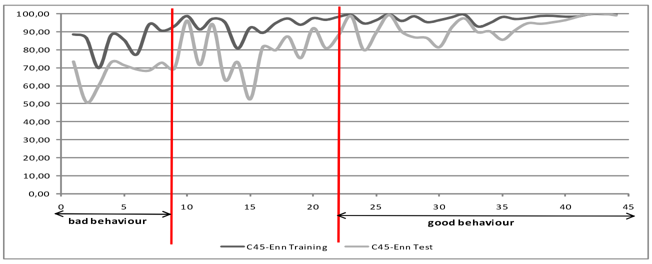

Figure 4. C4.5 with SMOTE-ENN AUC in Training/Test sorted by F1

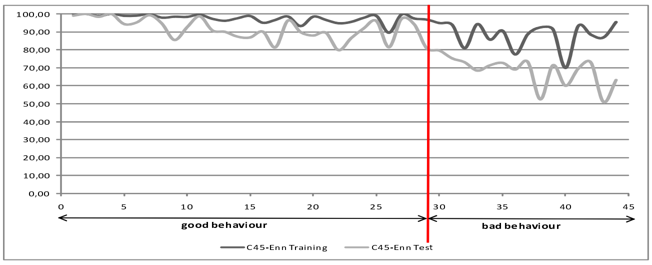

Figure 5. C4.5 with SMOTE-ENN AUC in Training/Test sorted by N4

depicts the data-sets so each one has the same space in the graphic representation. The Y axis depicts the AUC obtained both in training and test. We can find different *ad-hoc* intervals which present *good* or *bad behaviour* of C4.5 and we use a vertical line to delimit the interval of the region of interest.

- We understand as *good behaviour* an average high test AUC in the interval (at least 80%), as well as the absence of over-fitting.
- By *bad behaviour* we refer to the presence of over-fitting and/or average low test AUC in the interval.

Table 3. Significant intervals

C4.5 Interval	C4.5 with SMOTE-ENN Interval	Behaviour
$F1 \geq 2.391$	$F1 \geq 1.124$	*good behaviour*
$N4 \leq 0.1122$	$N4 \leq 0.2069$	*good behaviour*
$F1 \leq 0.366$	$F1 \leq 0.366$	*bad behaviour*
$N4 \geq 0.2261$	$N4 \geq 0.2261$	*bad behaviour*

In Table 3 we have summarized the intervals found ad-hoc from Figures 2 to 5. From these ad-hoc intervals we construct several rules that model the performance of C4.5. In Table 4 we have summarized the rules derived from Table 3 for C4.5 without preprocessing and the rules derived from the use of SMOTE-ENN. Given a particular data-set X, we get the complexity measure of X with the notation $CM[X]$. Table 4 is organised with the following columns.

- The first column corresponds to the identifier of the rule for further references.
- The "Rule'" column presents the rule itself.
- The third column "Support" presents the percentage of data-sets which verifies the antecedent of the rule.
- The column "% Training" shows the average AUC in training of all the data-sets covered by the rule.
- The column "Training Diff." contains the difference between the training AUC of the rule and the global training AUC.
- The column "% Test" shows the average AUC in test of all the data-sets covered by the rule.
- The column "Test Diff." contains the difference between the test AUC of the rule and the global test AUC.

The positive rules (denoted with a "+" symbol in their identifier) always show a positive difference with the global average AUC, both in training and test. The negative ones (with a "-" symbol) verify the opposite case. The support of the rules shows us that we can characterize a wide range of data-sets and obtain significant differences in AUC. We have also added the string "-W" for the obtained without preprocessing, and the string "-S-ENN" for the rules based on the use of SMOTE-ENN.

From this set of rules we can state that a high F1 value or a low N4 value results in a good behaviour of C4.5 in both cases (with and without preprocessing). On the other hand, a low value in the F1 metric or a high N4 value produces a bad behaviour of C4.5 in both scenarios. If we compare the use of SMOTE-ENN preprocessing with not preprocessing, we can observe two interesting facts:

- The positive rules increment their support significantly with the use of SMOTE-ENN preprocessing. The average AUC in training and test are similar to not preprocessing. Thus the positive rules for SMOTE-ENN characterize the regions in which the use of SMOTE-ENN preprocessing will allow C4.5 to increase its performance with respect to not preprocessing.
- The negative rules maintain their support. The differences in training AUC are lower, but the differences in test are similar. Therefore the negative regions have characterised the data-sets which do not improve with the use of SMOTE-ENN very accurately.

4.4. Collective Evaluation of the Set of Rules

The objective of this section is to analyse the good rules jointly, and the bad rules together as well, considering the application of the SMOTE-ENN preprocessing separately.

Table 4. Rules with one metric obtained from the intervals for C4.5

Id.	Rule	Support	%Training	Training Diff.	% Test	Test Diff.
C4.5 without preprocessing						
R1-W+	If $F1[X] \geq 2.391$ then *good behaviour*	31.82%	96.76%	9.43%	91.89%	12.6%
R2-W+	If $0 \leq N4[X] \leq 0.1122$ then *good behaviour*	36.36%	97.15%	9.82%	92.43%	13.14%
R1-W-	If $0.1691 \leq F1[X] \leq 0.366$ then *bad behaviour*	20.45%	74.84%	-12.49%	62.80%	-16.49%
R2-W-	If $N4[X] \geq 0.2261$ then *bad behaviour*	36.36%	74.19%	-13.14%	62.44%	-16.85%
C4.5 with SMOTE-ENN preprocessing						
R1-S-ENN+	If $F1[X] \geq 1.124$ then *good behaviour*	50.00%	97.85%	3.47%	92.34%	8.72%
R2-S-ENN+	If $0 \leq N4[X] \leq 0.2069$ then *good behaviour*	63.63%	97.56%	3.18%	91.96%	8.34%
R1-S-ENN-	If $0.1691 \leq F1[X] \leq 0.366$ then *bad behaviour*	20.45%	86.13%	-8.25%	67.62%	-16%
R2-S-ENN-	If $N4[X] \geq 0.2261$ then *bad behaviour*	36.36%	88.81%	-5.57%	69.03%	-14.59%

Table 5. Disjunction Rules from all simple rules

Id.	Rule	Support	%Training	Training Diff.	% Test	Test Diff.
C4.5 without preprocessing						
PRD-W	If R1+ or R2+ then *good behaviour*	43.18%	96.96%	9.63%	92.11%	12.82%
NRD-W	If R1- or R2- then *bad behaviour*	36.36%	74.19%	-13.14%	62.44%	-16.85%
not characterised	If not PRD and not NRD then *good behaviour*	20.45%	90.34%	3.01%	82.19%	2.9%
C4.5 with SMOTE-ENN preprocessing						
PRD-S-ENN	If R1+ or R2+ then *good behaviour*	63.63%	97.56%	3.18%	91.56%	7.94%
NRD-S-ENN	If R1- or R2- then *bad behaviour*	36.36%	88.81%	-5.57%	69.03%	-14.59%
not characterised	If not PRD and not NRD	0%	-%	-%	-%	-%

We perform the disjunctive combination of all the positive rules to obtain a single rule (and all the negative ones for another one) which will be activated if any of the component rules' antecedents are verified. In Table 5 we summarize both disjunctions, and a third rule representing those data-sets which are not charaterised by either disjunction rules.

From the collective rules we can observe that the support has been increased from the single rules for the Positive Rule Disjunction (PRD), while the Negative Rule Disjunction (NRD) maintains it. On the other hand, the training and test AUC differences are similar to the single rules from Table 4 in both with and without preprocessing situations. Since there are no data-sets in PRD and NRD simultaneously in both with and without preprocessing cases, we can consider three blocks of data-sets with their respective support, as depicted in Figure 6 and Figure 7 respectively. The data-sets have the same order in both figures.

- The first block (the left-side one) represents the data-sets covered by the PRD rule, which are recognized as being those in which C4.5 has good AUC.
- The second block (the middle one) contains the unclassified data-sets by the previous two rules.
- The third and last block (the right-side one) plots the data-sets for the rule NRD, which are bad for C4.5.

The 80% of the analysed data-sets in the case of not preprocessing are covered by PRD-W and NRD-W rules, and hence the *good behaviour* and *bad behaviour* consequents represent well the AUC of C4.5. In the case of SMOTE-ENN preprocessing, we can afford a full characteristation by both PRD-S-ENN or NRD-S-ENN rule.

The NRD-W rule and NRD-S-ENN rules cover the same data-sets, with little difference in AUC between both cases. The PRD-S-ENN rule has a larger support than the PRD-W rule, that is, the data-sets in the not characterized region are included in the PRD-S-ENN rule.

5. Conclusions

We have carried out a study in the framework of imbalanced data-sets to analyse the behaviour of preprocessing using C4.5. We have computed two data complexity measures over the imbalanced data-sets in order to obtain intervals of such metrics in which C4.5's performance is significantly good both using the original data and applying the

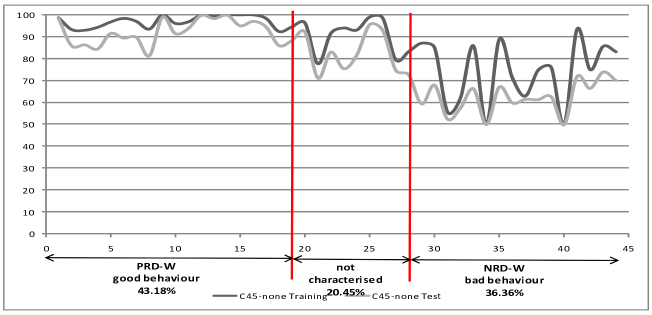

Figure 6. Three blocks representation for PRD, NRD and not covered data-sets for C4.5 without SMOTE-ENN

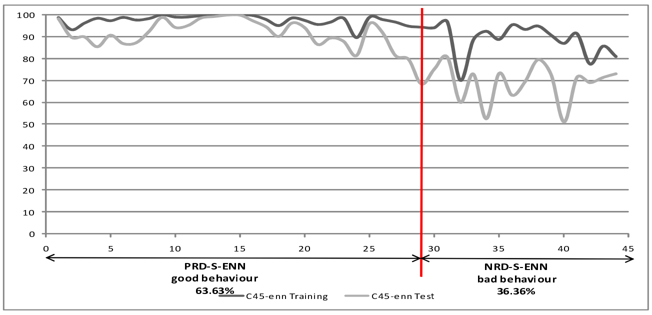

Figure 7. Three blocks representation for PRD, NRD and not covered data-sets for C4.5 with SMOTE-ENN

SMOTE-ENN preprocessing technique as an external solution to deal with this type of data.

We have constructed descriptive rules, and we have observed that the IR itself is not enough to predict when C4.5 obtains a good or bad performance, or when the use of SMOTE-ENN will enhance the results significatively.

We have obtained two rules from the initial ones, which are simple and precise to describe both good and bad performance of C4.5. The use of the SMOTE-ENN preprocessing technique has a positive influence on the rules obtained, since it produces an increment in the amount of data-sets characterised by the good rules, as well as improving the quality of all the rules in terms of average AUC. Therefore, we present the possibility of determining which data-sets C4.5 would increase its performance prior to the preprocessing execution, using the data complexity measures.

As future work our aim is to extend our study with a large collection of imbalance data-sets and to analyse in depth other data-complexity metrics that determines the behaviour of different types of computational intelligence techniques on the framework of imbalanced data-sets.

Acknowledgements This work had been supported by the Spanish Ministry of Science and Technology under Project TIN2008-06681-C06-01. J. Luengo holds a FPU scholarship from Spanish Ministry of Education and Science.

References

[1] J. Alcalá-Fdez, L. Sánchez, S. García, M. J. del Jesus, S. Ventura, J. M. Garrell, J. Otero, C. Romero, J. Bacardit, V. M. Rivas, J. C. Fernández, and F. Herrera. KEEL: A software tool to assess evolutionary algorithms to data mining problems. *Soft Computing*, 13(3):307–318, 2009.

[2] A. Asuncion and D. Newman. UCI machine learning repository, 2007.

[3] M. Basu and T. K. Ho. *Data Complexity in Pattern Recognition (Advanced Information and Knowledge Processing)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[4] G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard. A study of the behaviour of several methods for balancing machine learning training data. *SIGKDD Explorations*, 6(1):20–29, 2004.

[5] R. Baumgartner and R. L. Somorjai. Data complexity assessment in undersampled classification of high-dimensional biomedical data. *Pattern Recognition Letters*, 12:1383–1389, 2006.

[6] E. Bernadó-Mansilla and T. K. Ho. Domain of competence of xcs classifier system in complexity measurement space. *IEEE Trans. Evolutionary Computation*, 9(1):82–104, 2005.

[7] N. V. Chawla, N. Japkowicz, and A. Kolcz. Editorial: special issue on learning from imbalanced data sets. *SIGKDD Explorations*, 6(1):1–6, 2004.

[8] A. Fernández, S. García, M. J. del Jesus, and F. Herrera. A study of the behaviour of linguistic fuzzy rule based classification systems in the framework of imbalanced data–sets. *Fuzzy Sets and Systems*, 159(18):2378–2398, 2008.

[9] T. K. Ho and M. Basu. Complexity measures of supervised classification problems. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(3):289–300, 2002.

[10] A. Hoekstra and R. P. Duin. On the nonlinearity of pattern classifiers. In *ICPR '96: Proceedings of the International Conference on Pattern Recognition (ICPR '96) Volume IV-Volume 7472*, pages 271–275, Washington, DC, USA, 1996. IEEE Computer Society.

[11] J. Huang and C. X. Ling. Using AUC and accuracy in evaluating learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 17(3):299–310, 2005.

[12] A. Orriols-Puig and E. Bernadó-Mansilla. Evolutionary rule–based systems for imbalanced datasets. *Soft Computing*, 13(3):213–225, 2009.

[13] J. R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo–California, 1993.

[14] J. Sánchez, R. Mollineda, and J. Sotoca. An analysis of how training data complexity affects the nearest neighbor classifiers. *Pattern Anal. Appl.*, 10(3):189–201, 2007.

[15] Q. Yang and X. Wu. 10 challenging problems in data mining research. *International Journal of Information Technology and Decision Making*, 5(4):597–604, 2006.