Exploring Resampling with Neighborhood Bias on Imbalanced Regression Problems

Paula Branco[1,2], Luís Torgo[1,2], and Rita P. Ribeiro[1,2]

[1] LIAAD - INESC TEC
[2] DCC - Faculdade de Ciências - Universidade do Porto
{paula.branco,ltorgo,rpribeiro}@dcc.fc.up.pt

Abstract. Imbalanced domains are an important problem that arises in predictive tasks causing a loss in the performance of the most relevant cases for the user. This problem has been intensively studied for classification problems. Recently it was recognized that imbalanced domains occur in several other contexts and for a diversity of types of tasks. This paper focus on imbalanced regression tasks. Resampling strategies are among the most successful approaches to imbalanced domains. In this work we propose variants of existing resampling strategies that are able to take into account the information regarding the neighborhood of the examples. Instead of performing sampling uniformly, our proposals bias the strategies for reinforcing some regions of the data sets. In an extensive set of experiments we provide evidence of the advantage of introducing a neighborhood bias in the resampling strategies.

1 Introduction

The class imbalance problem is well known and has been thoroughly studied [7, 10]. This problem has important real world applications spanning from the medical to the meteorological or financial domains, among many others. In this type of predictive tasks, the goal of obtaining a model is hampered by the conjugation of: i) the non-uniform preferences of the user; and ii) the poor representation on the available data of the most important cases.

The study of the problem of imbalanced domains started with classification tasks, and in particular with two class problems. The majority of solutions for this problem is still concentrated in binary classification tasks. More recently, it was shown that the problem of imbalanced domains also arises in several other tasks, namely: regression, data streams or multi-target prediction tasks [3, 8].

In this paper, we address the problem of imbalanced domains in regression, to which we will refer as the imbalanced regression problem. In a regression context, the continuous nature of the target variable brings an extra level of difficulty to the problem. Moreover, the definition of the more and less important values of the target variable is not as straightforward as in a classification tasks. We will refer to the less important cases in a data set as the normal cases, while rare/interesting cases will be the most important. To address imbalanced regression problems some proposals for pre-processing the given data set have been made (e.g. [17]). Still, as far as we know, no attempt was made for biasing the new data set taking into consideration the neighborhood of the examples.

The main goal of this paper is to study the impact of introducing a bias both in the generation of new synthetic cases and in the removal of cases considering the type of nearest neighbors (normal or rare) of each case. This bias can be introduced to either favor the "safer" and easier to learn cases (i.e., cases surrounded by cases of the same type), or to reinforce the "frontier" or harder to learn cases (i.e., cases whose nearest neighbors are mainly from a different type).

This paper is organized as follows. In Section 2 the problem definition is presented. Section 3 provides an overview of the related work. Our proposals are described in Section 4 and the results of an extensive experimental evaluation are discussed in Section 5. Finally, Section 6 presents the main conclusions.

2 Problem Definition

The problem of imbalanced domains occurs in the context of predictive tasks, where the goal is to obtain a model that approximates an unknown function $Y = f(\mathbf{x})$. To achieve this goal a training set $\mathcal{D} = \{\langle \mathbf{x}_i, y_i \rangle\}_{i=1}^{N}$ with N examples is used. When the target variable Y is continuous we face a regression task and when it is nominal we have a classification task.

Imbalanced regression problems are a particular class of regression tasks. In imbalanced regression the user preferences are not uniform across the target variable domain, i.e., the user assigns more importance to the predictive performance in some ranges of the target variable. Moreover, there is a poor representation of the most relevant ranges in the available training set $\mathcal{D}$. The conjunction of these two factors is the key source of problems, because it causes a performance degradation on the most important cases for the user. The learning algorithms are not able to focus on the most important ranges of the target variable due to the lack of examples in those ranges.

This setting is similar to the class imbalance problem where the most important class is under-represented in the given training set leading to a poor performance in the important class. Typically, when dealing with a class imbalance problem, the user simply states which is the important class without specifically quantifying how much each class is important. This becomes more complicated when dealing with multiclass imbalanced problems. In this case, there can be several important and less important classes and their importance may not be easy to define. The simple consideration of multiclass leads to an increased difficulty when dealing with this problem. Therefore, tackling an imbalanced regression problem implies an increased level of difficulty because the target variable has a potentially infinite number of values.

To address the problem of defining the target variable important ranges the notion of a **relevance function** was proposed by Torgo and Ribeiro [16] and Ribeiro [12]. The **relevance function**, $\phi : \mathcal{Y} \rightarrow [0, 1]$, maps the target variable domain into a scale of relevance, where 1 corresponds to the maximal relevance and 0 to the minimum relevance. The task of defining this relevance can be hard in regression problems. Ideally, a domain expert should provide this information. Although being the user responsibility to provide the relevance function, Ribeiro [12] proposed an automatic way for obtaining this information. Function $\phi(y)$ is estimated from the target variable domain distribution assuming that the rare and most extreme cases are the most relevant to the user, which is typically the case.

The relevance function values can be used to determine the sets of normal and rare values. To achieve this the user is required to set a threshold t_R on the relevance values. Given this threshold we can formally define the set of rare and relevant cases, $\mathcal{D}_R$, and the set of normal and uninteresting cases, $\mathcal{D}_N$, as follows: $\mathcal{D}_R = \{\langle \mathbf{x}, y \rangle \in \mathcal{D} : \phi(y) \geq t_R\}$ and $\mathcal{D}_N = \{\langle \mathbf{x}, y \rangle \in \mathcal{D} : \phi(y) < t_R\}$.

Let us consider, for instance, a regression problem where the target variable values represent the values of a sensor in a given machine. When the sensor indicates the most common value, typically there is not problem. However, when the sensor indicates extremely high or low values, then this can represent, for instance, a malfunction in the machine. These extreme values will be the most relevant for the user. Still, these values are usually under-represented in the data set because a normal functioning machine is expected for most of the time.

To handle imbalanced domains, it is required to take into account: i) the performance evaluation issue; and ii) the problem of biasing the learning algorithms towards the user preferences. Regarding the first issue, it has been shown that standard metrics are not suitable for this type of problems [12, 3]. Therefore, new metrics were proposed for dealing with imbalanced domains in classification and also in regression, although fewer exist for the latter. A framework for obtaining precision and recall for imbalanced regression tasks was proposed in [14] and [12]. This framework is able to capture the key features of precision and recall measures defined for classification as well as the notion of numeric error needed in regression. In this paper we use the F_1-measure (F_1^ϕ) proposed in [1] that is based on the mentioned framework [14, 12]. The contributions in this paper concern the second issue of biasing the learners towards the important rare cases. We explore the introduction of a bias linked with the cases neighborhood on existing pre-processing strategies for regression. This bias can be put forward in several different ways that we also study.

3 Related Work

As we have mentioned, most of the existing work regarding imbalanced domains is concentrated on binary classification tasks. More recently, solutions for this problem in other tasks began to appear. Pre-processing solutions are among the most commonly used because they act by changing the original data distribution, and therefore allow the use of any standard learning algorithm. Pre-processing methods for dealing with imbalanced domains mainly range between removing normal cases, including replicas of the rare cases, or generating new synthetic examples. These methods are efficient because they change the target variable distribution so that the learners focus on the rare and important examples. However, how this change should be carried out, is still an open research question.

The number of pre-processing strategies that has been proposed for classification tasks is extensive [3]. Still, for dealing with imbalanced regression only two pre-processing methods were proposed: random under-sampling [17, 15] and smoteR [17]. These methods were initially proposed for dealing with class imbalance and were later adapted to a regression context. Random under-sampling is a simple method that changes the target variable distribution by randomly removing normal cases, i.e., cases with the less important target variable values. This way a better balance is achieved between rare (important) and normal (uninteresting) cases. A relevance function and a threshold are used to determine D_R and D_N. The method also requires setting a parameter that represents the reduction to be carried out in the normal cases. An adaption of the well-known Smote [4] algorithm was proposed for regression with the name of SmoteR [17]. This proposal combines the application of random under-sampling in the normal cases with the generation of new synthetic "smoted" examples from the rare cases. This method also depends on the definition of a relevance function for setting the rare and normal cases. The synthetic examples are generated through an interpolation strategy. The key idea is to build a new synthetic example by interpolating the features of two rare cases. The target variable value of the new case is determined as a weighted average of the target variable values of the two rare cases used. All rare cases are used in turn as seed examples. The user is also required to define the percentage of over and under-sampling to be carried out[3].

The most closely related proposal regarding the introduction of a bias in the generation of synthetic examples that takes into account each example neighborhood is the Adaptive Synthetic (ADASYN) method [6]. ADASYN was proposed for dealing with imbalanced classification. The key idea of this method is to use a density distribution for deciding the number of synthetic examples to generate for each original rare class case. A bias is introduced on the generation of new synthetic examples that favors the examples from the minority and important class cases that are closer to the decision border. With ADASYN, more synthetic examples are generated for the rare class cases that are harder to learn (i.e. with a larger number of neighbors from the normal class), while fewer new cases are generated for the easier examples.

4 Biasing Pre-processing Strategies

In this section we will describe our proposals regarding the introduction of a bias in resampling strategies for regression by considering the examples neighborhood. We propose two methods: one for adapting under-sampling and another for adapting an over-sampling strategy. These methods consider the neighborhood of each example for allowing the introduction of a bias on the resampling strategies. The methods proposed were tested on adaptations of two previously

[3] Further details regarding SmoteR algorithm can be obtained in [17].

proposed strategies: random under-sampling and SmoteR. As mentioned before, random under-sampling approach simply removes normal and uninteresting cases while SmoteR strategy combines under-sampling of normal cases with the generation of new synthetic rare cases. These strategies either uniformly select normal cases to be removed or generate new cases using uniformly each rare case. Our proposals for biasing under- and over-sampling allow to bias these strategies using the information of each case neighborhood.

The key idea of resampling with neighborhood bias is to inspect the examples nearest neighbors distribution in order to decide which normal cases should be removed with higher probability or which rare cases should be used more frequently as seed examples in the generation of new cases. We highlight that when applying an under-sampling strategy we are only interested in removing **normal cases** (i.e., examples in $\mathcal{D}_N$), while for over-sampling we are only concerned with increasing the **rare cases** (i.e., examples that belong to $\mathcal{D}_R$). Our proposals will bias the resampling strategies to achieve a non-uniform sampling that takes into consideration the distribution of the examples neighbors.

Let us begin with the definition of **frontier** and **safe** cases. An example $ex_i = \langle \mathbf{x}_i, y_i \rangle \in D_R$ (D_N) is as closer to the **frontier** as higher is the number of its k-nearest neighbors (kNN) that belong to D_N (D_R). An example $ex_i = \langle \mathbf{x}_i, y_i \rangle \in D_R$ (D_N) is as **safe** as higher is the number of its kNN that belong to D_R (D_N). This means that a rare case (belonging to D_R) having all its kNN belonging to D_N is as close as possible to the frontier. In this situation, the rare case is completely surrounded by cases from a different type (normal) and can be thought as an harder to learn case. On the other hand, a rare case is as safe as possible when all its kNN are also rare. This case can also be thought as an easy to learn case. We highlight that these notions apply in a similar way to both rare and normal cases. For introducing a bias in either under-sampling or over-sampling strategies the following two main variants may be considered: i) **reinforce the frontier**, or harder to learn cases; and/or ii) **reinforce the safe** or easier to learn cases. Both variants can be applied on the normal and on the rare cases, that is, we can reinforce the frontier cases either on the normal or the rare cases, and the same applies for reinforcing the safe cases.

When performing over-sampling, the variant that reinforces the frontier generates more synthetic examples for the cases having a larger number of normal nearest neighbors. On the other hand, when applying under-sampling to the normal cases, the frontier is reinforced when the examples with more rare neighbors are more likely to be kept. In both situations, the bias will favor the cases closer to the frontier. The key idea for reinforcing the safe cases is to bias the resampling in favor of these cases. When applying over-sampling, more synthetic cases should be generated for the examples with higher number of rare nearest neighbors. On the other hand, the application of under-sampling for reinforcing the safe cases assumes that normal cases having more rare neighbors should be more likely to be removed. Table 1 summarizes the application of variants described on the two unbiased resampling strategies used in this paper: random under-sampling and SmoteR.

Table 1. Summary of the resampling variants with neighborhood bias.

Acronym	Strat	Normal	Rare	Acronym	Strat	Normal	Rare
S._._	SmoteR	-	-	U._._	Undersamp.	-	-
S.F.F	SmoteR	frontier	frontier	U.F._	Undersamp.	frontier	-
S.F.S	SmoteR	frontier	safe	U.S._	Undersamp.	safe	-
S.S.F	SmoteR	safe	frontier				
S.S.S	SmoteR	safe	safe				

Algorithm 1: Under-sampling with neighborhood bias.

```
Input: $\mathcal{D}$ - original regression data set
       $Bin_N$ - subset of $\mathcal{D}$ with normal cases
       $tgtNr$ - target number of examples to obtain in the new data set
       $k$ - nr of k nearest neighbors
       $Fr$ - logical value indicating if the reinforcement is applied to the
           frontier (TRUE) or safe (FALSE) cases
Output: $newD$ - a new under-sampled data set
$KNNs \leftarrow kNN(Bin_N, D, k)$ // k-NN in set $\mathcal{D}$ of examples in $Bin_N$
$\boldsymbol{r} \leftarrow$ vector of dimension $|Bin_N|$
foreach $x_i \in Bin_N$ do
    if $Fr = TRUE$ then
        $\Delta_i \leftarrow$ nr of KNNs of $x_i$ that belong to $Bin_N$
    else
        $\Delta_i \leftarrow$ nr of KNNs of $x_i$ that do not belong to $Bin_N$
    $r_i \leftarrow \Delta_i / k$
$\hat{\boldsymbol{r}} \leftarrow \boldsymbol{r} / \sum_{i=1}^{|Bin_N|} r_i$
$newD \leftarrow$ sample $tgtNr$ examples from $Bin_N$ with probability $\hat{\boldsymbol{r}}$
return $newD$
```

Algorithms 1 and 2 show with more detail how our proposed variants for biasing the resampling strategies are obtained. We highlight that Algorithm 1 uses as input Bin_N: a subset of $\mathcal{D}_\mathcal{N}$ with a given range of normal cases. This happens because it only makes sense to apply an under-sampling strategy to the normal cases. The same reasoning applies to the over-sampling strategy which we expect to be applied on rare cases. Therefore, in Algorithm 2 a subset $Bin_R \subseteq \mathcal{D}_\mathcal{R}$ with a given range of rare cases is considered. Both Algorithms may be applied to one or more subsets of normal (Bin_N) or rare (Bin_R) cases. This can occur for instance on the rare cases when the user defines two relevant and distinct regions of the target variable values. In this case, the rare cases in $\mathcal{D}_R$ belong to two distinct bins, for instance, the cases with extreme low and high target variable values. When this occurs, the under-/over-sampling strategies should be applied in each bin separately. Algorithm 1 returns a data set $newD$ with an under-sampled with neighborhood bias Bin_N. Algorithm 2 returns a data set $newD$ with a new set of examples obtained from Bin_R with a neighborhood bias through a user provided over-sampling function $GenEx$.

Algorithm 2: Over-sampling with neighborhood bias.

```
Input: $\mathcal{D}$ - original regression data set
       $Bin_R$ - subset of $\mathcal{D}$ with rare cases
       $tgtNr$ - number of new examples to generate in the new data set
       $k$ - nr of k nearest neighbors
       $Fr$ - logical value indicating if the reinforcement is applied to the
           frontier (TRUE) or safe (FALSE) cases
       $GenEx$ - function for obtaining the new examples
Output: $newD$ - a data set containing the new examples
$KNNs \leftarrow kNN(Bin_R, D, k)$ // k-NN in set $\mathcal{D}$ of examples in $Bin_R$
$\boldsymbol{r} \leftarrow$ vector of dimension $|Bin_R|$
foreach $x_i \in Bin_R$ do
    if $Fr = TRUE$ then
        $\Delta_i \leftarrow$ nr of KNNs of $x_i$ that do not belong to $Bin_R$
    else
        $\Delta_i \leftarrow$ nr of KNNs of $x_i$ that belong to $Bin_R$
    $r_i \leftarrow \Delta_i / k$
$\hat{\boldsymbol{r}} \leftarrow \boldsymbol{r} / \sum_{i=1}^{|Bin_R|} r_i$
for $i = 1$ to $|Bin_R|$ do
    $g_i \leftarrow \hat{r}_i \times tgtNr$
$newD \leftarrow$ use $GenEx$ function to generate $g_i$ new examples for each $x_i$.
return $newD$
```

We highlight that the decision of either reinforcing the frontier or the safe cases may not be trivial. In fact, the better option can be data dependent and several reasons may be pointed out for and against the two options. For instance, if we consider a data set having high levels of noisy examples, then, it is probably better to generate new rare examples based on the existing safe rare cases. However, if we have a data set with few noisy examples, then, the use of the frontier cases for obtaining new cases can be beneficial.

5 Experimental Evaluation

The main goal of our experiments is to assess the effectiveness of introducing a bias on the pre-processing strategies. We have selected 18 regression data sets from different domains whose main characteristics are described in Table 2. For each of these data sets we have obtained a relevance function through the automatic method proposed in [12]. This method assigns higher relevance to high and low extreme values of the target variable using the quartiles and the interquartile range of the target variable distribution[4]. We considered a threshold of 0.8 on the relevance values in all data sets. To ensure the reproducibility of our results, all code, data sets used and main results are available in `https://github.com/paobranco/NeighborhoodBiasResamplingRegression`. All experiments were carried out in the R environment and we selected the three

[4] Further details available in [12].

Table 2. Data sets information by descending order of rare cases percentage. (N: nr of cases; *tpred*: nr predictors; *p.nom*: nr nominal predictors; *p.num*: nr numeric predictors; *nRare*: nr. cases with $\phi(y) > 0.8$; %*Rare*: *nRare*/N).

Data Set	N	tpred	p.nom	p.num	nRare	% Rare	Data Set	N	tpred	p.nom	p.num	nRare	% Rare
servo	167	4	2	2	34	0.204	a2	198	11	3	8	22	0.111
a6	198	11	3	8	33	0.167	fuelCons	1764	38	12	26	164	0.093
Abalone	4177	8	1	7	679	0.163	availPwr	1802	16	7	9	157	0.087
machCpu	209	6	0	6	34	0.163	cpuSm	8192	13	0	13	713	0.087
a3	198	11	3	8	32	0.162	maxTorq	1802	33	13	20	129	0.072
a4	198	11	3	8	31	0.157	bank8FM	4499	9	0	9	288	0.064
a1	198	11	3	8	28	0.141	ConcrStr	1030	8	0	8	55	0.053
a7	198	11	3	8	27	0.136	Accel	1732	15	3	12	89	0.051
boston	506	13	0	13	65	0.128	airfoild	1503	5	0	5	62	0.041

Table 3. Regression algorithms, parameter variants, and respective R packages used.

Learner	Parameter Variants	R package
MARS	$nk = \{10, 17\}, degree = \{1, 2\}, thresh = \{0.01, 0.001\}$	**earth** [11]
SVM	$cost = \{10, 150, 300\}, gamma = \{0.01, 0.001\}$	**e1071** [5]
RF	$mtry = \{5, 7\}, ntree = \{500, 750, 1500\}$	**randomForest** [9]

following types of learning algorithms: Multivariate Adaptive Regression Splines (MARS), Support Vector Machines (SVM) and Random Forests (RF). The learning algorithms, respective R packages and the used parameter variants are displayed in Table 3. We applied each of the 20 learning approaches (8 MARS + 6 SVM + 6 RF) to each of the 18 regression data sets using 9 resampling strategies. Thus 3240 ($20 \times 18 \times 9$) combinations were tested. All the resampling strategies were applied with the goal of balancing the rare and normal cases in the data sets. The 9 resampling strategies applied were as follows: i) use the original data set without any pre-processing ("none"); ii) apply the original SmoteR method without any bias; iii) apply the original random under-sampling method; iv) apply 4 variants of neighborhood bias with SmoteR; v) apply two variants of neighborhood bias with under-sampling. Table 1 describes the resampling variants and acronyms used.

All the alternatives described were evaluated using the F_1^ϕ measure for regression referred in Section 2. We used $\beta = 1$, which means that the same importance is given to both precision and recall scores. The F_1^ϕ values were estimated through a 2×10 - fold cross validation process and the statistical significance of the observed paired differences was measured using the non-parametric Wilcoxon paired test. The experiments were carried out using the following R packages: *performanceEstimation* [13] for the experimental infra-structure; *uba*[5] for the relevance function and F_1^ϕ metric; and *UBL* [2] for the implementation of the random under-sampling and SmoteR resampling strategies.

We summarize the main results in Figures 1 and 2. We provide the detailed results, as well as all the code and data sets used in `https://github.com/paobranco/NeighborhoodBiasResamplingRegression`. Figure 1 shows the number of best median F_1^ϕ scores across all strategies by learner type in all data sets tested, i.e., it counts the number of times each strategy (aggregated by none,

[5] Available at `http://www.dcc.fc.up.pt/~rpribeiro/uba/`.

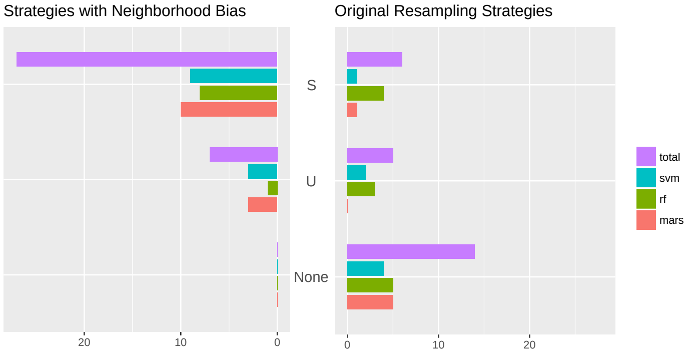

Fig. 1. Number of data sets with best median F_1^{ϕ} scores by learner and strategy type (S: SmoteR-based; U: undersampling-based).

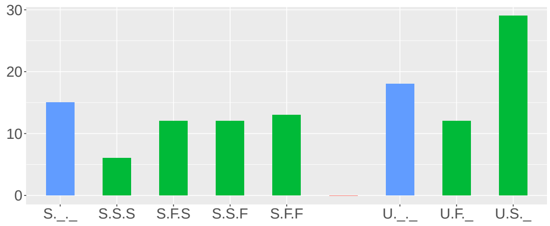

Fig. 2. Number of best median F_1^{ϕ} scores inside each type of resampling strategy.

original or with neighborhood bias) has the best overall F_1^{ϕ} score in each data set. Figure 2 shows the number of best median F_1^{ϕ} scores observed when taking into account only the scores inside each type of resampling strategy, i.e., for each strategy type we counted the number of times that each variant displayed the best score, considering only the scores obtained on those strategies.

The results presented show that there is an advantage on considering the new biased resampling strategies using the examples neighborhood. However, it is not straightforward which variant should be selected for each data set. This means that the resampling strategies with a neighborhood bias show an improved gain in F_1^{ϕ} when compared to the original resampling strategies. Still, we are not able to identify which is the biased strategy that has the best overall results on the 18 considered data sets. The scores obtained, although generally better with the introduction of bias in the resampling strategies, seem to be domain dependent in what concerns the reinforcement of the frontier or the safe cases.

Figure 3 show the total number of F_1^{ϕ} wins and losses (and significant wins/losses) for each resampling strategy against the baseline of using the original data set. The results were obtained with the Wilcoxon Signed Rank test for each data set with a significance level of 95%. Darker bars indicate significant wins/losses while lighter bars represent wins/losses without significance. These figures confirm that there is an advantage on considering the biased resampling strategies on imbalanced regression tasks. However, the results obtained are clearly dependent of the used learning algorithm. This is evident when comparing the results of Random Forest learner against the remaining learners.

We also compared the wins/losses of our proposed biased resampling strategies against the original resampling strategies. Figures 4 and 5 show the wins and losses of F_1^ϕ score of the neighborhood biased alternatives against respectively the SmoteR and random under-sampling strategies as baseline. The results of the comparison of the biased strategies against the SmoteR as baseline are not conclusive. Still, the S.F.S strategy that reinforces the safe rare cases and the frontier of the normal cases stands out. However, for the RF and MARS learners the original SmoteR strategy presents globally more wins. In the comparison against random under-sampling as baseline, the strategy that reinforces the frontier cases has more wins and significantly for all learners. This confirms that this biased strategy is preferable to the random under-sampling strategy.

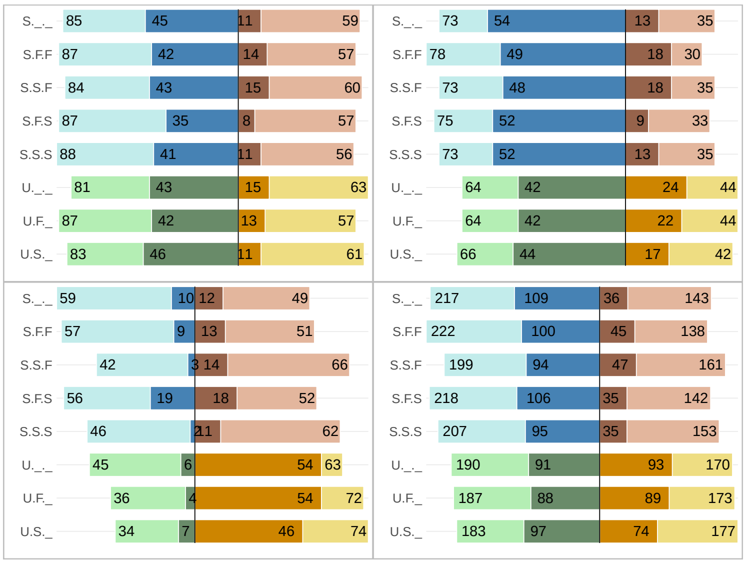

Fig. 3. Wins (left) and losses (right) of each learner (top left: MARS, top right: SVM, bottom left: RF and bottom right: Total) against using the original data set.

6 Conclusions

In this paper we studied the introduction of a neighborhood bias on resampling strategies for dealing with the problem of imbalanced domains in regression

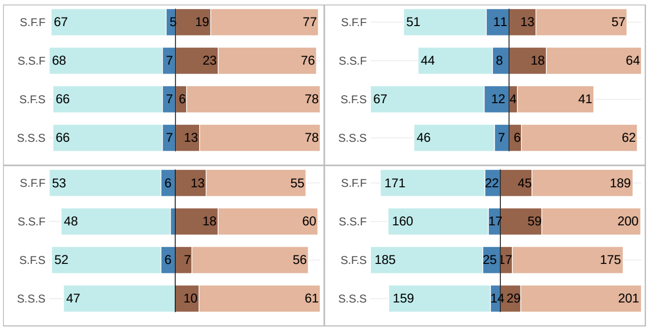

Fig. 4. Wins (left) and losses (right) of each learner (top left: MARS, top right: SVM, bottom left: RF and bottom right: Total) against the SmoteR strategy.

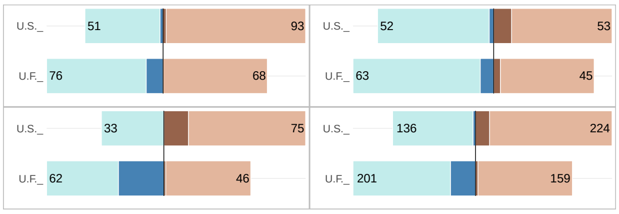

Fig. 5. Wins (left) and losses (right) of each learner (top left: MARS, top right: SVM, bottom left: RF and bottom right: Total) against the random under-sampling strategy.

tasks. The goal of introducing a bias in the pre-processing strategies is to avoid an uniform under-/over-sampling reinforcing some regions of the data sets at the expense of other regions. We use the information on the examples neighborhood to bias the resampling strategies towards the safe and/or frontier regions.

We show that there is a clear advantage when considering resampling strategies with a neighborhood bias. Moreover, the new strategies can easily be extended to other resampling strategies. The key contributions of this paper are: i) the proposal of new resampling strategies that take into account the information on the examples neighborhood; ii) test and compare our proposals against the baseline of not applying resampling and the original unbiased strategy.

As future work we plan to extend these approaches to imbalanced classification tasks, comparing the impact of reinforcing the safe/frontier cases in different data sets.

Acknowledgments. This work is financed by the ERDF European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation - COMPETE 2020 Programme within project POCI-01-0145-FEDER-006961, and by National Funds through the FCT Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) as part of project UID/EEA/50014/2013. P. Branco is supported by a

PhD scholarship of FCT (PD/BD/105788/2014). Prof. L. Torgo would also like to thank the support of Projects NORTE-01-0145-FEDER-000036 and UTAP-ICDT/CTM-NAN/0025/2014.

References

1. Branco, P.: Re-sampling Approaches for Regression Tasks under Imbalanced Domains. Master's thesis, Dep. Computer Science, Faculty of Sciences - University of Porto (2014)
2. Branco, P., Ribeiro, R.P., Torgo, L.: UBL: an R package for utility-based learning. arXiv preprint arXiv:1604.08079 (2016)
3. Branco, P., Torgo, L., Ribeiro, R.P.: A survey of predictive modeling on imbalanced domains. ACM Computing Surveys (CSUR) 49(2), 31 (2016)
4. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: Synthetic minority over-sampling technique. JAIR 16, 321–357 (2002)
5. Dimitriadou, E., Hornik, K., Leisch, F., Meyer, D., Weingessel, A.: e1071: Misc Functions of the Department of Statistics (e1071), TU Wien (2011)
6. He, H., Bai, Y., Garcia, E.A., Li, S.: Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In: Neural Networks, 2008. IEEE International Joint Conference on. pp. 1322–1328. IEEE (2008)
7. He, H., Garcia, E.A.: Learning from imbalanced data. IEEE Transactions on knowledge and data engineering 21(9), 1263–1284 (2009)
8. Krawczyk, B.: Learning from imbalanced data: open challenges and future directions. Progress in Artificial Intelligence pp. 1–12 (2016)
9. Liaw, A., Wiener, M.: Classification and regression by randomforest. R News 2(3), 18–22 (2002)
10. López, V., Fernández, A., García, S., Palade, V., Herrera, F.: An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. Information Sciences 250, 113–141 (2013)
11. Milborrow, S.: earth: Multivariate Adaptive Regression Spline Models. Derived from mda:mars by Trevor Hastie and Rob Tibshirani. (2012)
12. Ribeiro, R.P.: Utility-based Regression. Ph.D. thesis, Dep. Computer Science, Faculty of Sciences - University of Porto (2011)
13. Torgo, L.: An infra-structure for performance estimation and experimental comparison of predictive models in r. CoRR abs/1412.0436 (2014)
14. Torgo, L., Ribeiro, R.P.: Precision and recall in regression. In: DS'09: 12th Int. Conf. on Discovery Science. pp. 332–346. Springer (2009)
15. Torgo, L., Branco, P., Ribeiro, R.P., Pfahringer, B.: Resampling strategies for regression. Expert Systems 32(3), 465–476 (2015)
16. Torgo, L., Ribeiro, R.P.: Utility-based regression. In: PKDD'07. pp. 597–604. Springer (2007)
17. Torgo, L., Ribeiro, R.P., Pfahringer, B., Branco, P.: Smote for regression. In: Progress in Artificial Intelligence, pp. 378–389. Springer (2013)