Resampling Strategies for Imbalanced Time Series

Nuno Moniz
LIAAD-INESC TEC
DCC-FCUP, University of Porto
Porto, Portugal
Email: nmmoniz@inescporto.pt

Paula Branco
LIAAD-INESC TEC
DCC-FCUP, University of Porto
Porto, Portugal
Email: paobranco@gmail.com

Luís Torgo
LIAAD-INESC TEC
DCC-FCUP, University of Porto
Porto, Portugal
Email: ltorgo@dcc.fc.up.pt

***Abstract*—Time series forecasting is a challenging task, where the non-stationary characteristics of the data portrays a hard setting for predictive tasks. A common issue is the imbalanced distribution of the target variable, where some intervals are very important to the user but severely underrepresented. Standard regression tools focus on the average behaviour of the data. However, the objective is the opposite in many forecasting tasks involving time series: predicting rare values. A common solution to forecasting tasks with imbalanced data is the use of resampling strategies, which operate on the learning data by changing its distribution in favor of a given bias. The objective of this paper is to provide solutions capable of significantly improving the predictive accuracy of rare cases in forecasting tasks using imbalanced time series data. We extend the application of resampling strategies to the time series context and introduce the concept of temporal and relevance bias in the case selection process of such strategies, presenting new proposals. We evaluate the results of standard regression tools and the use of resampling strategies, with and without bias over 24 time series data sets from 6 different sources. Results show a significant increase in predictive accuracy of rare cases associated with the use of resampling strategies, and the use of biased strategies further increases accuracy over the non-biased strategies.**

Keywords*—*Imbalanced Time Series, Resampling Strategies, Temporal Bias.

I. Introduction

Mining time series data is one of the most challenging problems in the field of data mining [1]. Time series forecasting holds a key importance in many application domains, where time series data is highly imbalanced. This occurs when certain ranges of values are over-represented in comparison to others and the user is particularly interested in the predictive performance on values that are the least represented. Such examples may be found in financial data analysis, intrusion detection in network forensics, oil spill detection and prognosis of machine failures. In these scenarios of imbalanced data sets, standard learning algorithms bias the models toward the more frequent situations, away from the user preference biases, proving to be an ineffective approach and a major source of performance degradation [2].

A common solution for the general problem of mining imbalanced data sets is to resort to resampling strategies. These strategies change the distribution of learning data in order to balance the number of rare and normal cases, attempting to reduce the skewness of the data. These strategies commonly achieve their goal by under or oversampling the data. In the former, the cases considered as normal (*i.e.* the majority of cases) are removed from the learning data; in the latter, cases considered to be rare (*i.e.* the minority) are generated and added to the data. For example, in fraud detection problems fraud cases are infrequent, and detecting them is the prime objective. Also, in intrusion detection problems, most of the behaviour in networks are normal and cases of intrusion, which one aims to detect, are scarce. This task of predicting rare occurrences has proven to be a difficult task to solve, but due to its importance in so many domains, it is a fundamental problem within predictive analytics [3].

Most existing work using resampling strategies for predictive tasks with an imbalanced target variable distribution involves classification problems ([4], [5], [6], [7]). Recently, efforts have been made to adapt existing strategies to numeric targets, *i.e.* regression problems ([8], [9]). To the best of our knowledge, no previous work addresses this question using resampling strategies in the context of time series forecasting. Although time series forecasting involves numeric predictions, there is a crucial difference compared to regression tasks due to the time dependency among the observed values. The main motivation of the current work is our claim that this order dependency should be taken into account when changing the distribution of the training set, *i.e.* when applying resampling. Our work is driven by the assumption that by biasing the sampling procedure with information on this order dependency, we are able to improve predictive performance.

In this paper, we study the use of resampling strategies in imbalanced time series. Our endeavour is based on two strategies: *i)* the first is based on undersampling (random undersampling [10]) and *ii)* the second combines undersampling and oversampling (**S**ynthetic **M**inority **O**ver-sampling **TE**chnique [11]). Both strategies were initially proposed for classification problems, and were posteriorly extended for regression tasks [8], [9]. We will refer to the extension of the SMOTE resampling strategy as SmoteR.

Time series often exhibit systematic changes in the distribution of observed values. These non-stationarities are often known as *concept drift* [12]. This concept describes the changes in the conditional distribution of the target variable in relation to the input features (*i.e.* predictors), whilst the distribution of the latter stays unchanged. This raises the question of how to devise learning approaches capable of coping with this issue. We introduce the concept of temporal bias in resampling strategies associated with forecasting tasks using imbalanced time series. Our motivation is the idea that in an imbalanced time series, where concept drift occurs, it is possible to improve forecasting accuracy by introducing a temporal bias in the case selection process of resampling strategies. This bias favours cases that are within the temporal vicinity of apparent regime changes. In this paper we propose two alternatives for the resampling strategies used in our work: undersampling and SmoteR with *1)* temporal bias, and *2)* with temporal and relevance bias.

An extensive experimental evaluation was carried out to evaluate our proposals comprising 24 time series data sets from 6 different sources. The objective is to verify if resampling strategies are capable of improving the predictive accuracy in comparison to standard forecasting tools, including those designed specifically for time series (*e.g.* ARIMA models [13]).

The contributions of this paper are:

- The extension of resampling strategies for time series;
- The proposal of novel resampling strategies that introduce the concept of temporal and relevance bias;
- An extensive evaluation including standard regression tools, time series specific models (ARIMA) and the use of resampling strategies.

The remainder of this paper is structured as follows. In Section II the problem tackled in our work is introduced and the assumptions in which our proposals are based are presented. Resampling strategies are described in Section III along with the adaptation of previous proposals, and new proposals. The data used to evaluate the proposals is introduced in Section IV, as well as the regression tools used and the evaluation methods. The evaluation process is described and results presented in Section V. Finally, previous work is discussed in Section VI and conclusions are presented in Section VII.

II. Problem Definition

The main objective of our proposals is to provide solutions that significantly improve the predictive accuracy of relevant (rare) cases in forecasting tasks using imbalanced time series.

The task of time series forecasting assumes the availability of a time-ordered set of observations of a given continuous variable $y_1, y_2, \ldots, y_t \in Y$, where y_t is the value measured at time t. The objective of this predictive task is to forecast the future value(s) of variable Y. The overall assumption is that an unknown function correlates the past and future values of Y, *i.e.* $Y_{t+h} = f(\langle Y_{t-k}, \ldots, Y_{t-1}, Y_t \rangle)$. The goal of the learning process is to provide an approximation of this unknown function. This is carried out using a data set with historic examples of the function mapping (*i.e.* training set).

Time series forecasting models usually assume the existence of a degree of correlation between successive values of the series. A form of modeling this correlation consists of using the previous values of the series as predictors of the future value(s), in a procedure known as time delay embedding [14]. This process allows the use of standard regression tools on time series forecasting tasks. However, specific time series modelling tools already exist, such as the ARIMA models [13].

In this work we focus on imbalanced time series, where certain ranges of values of the target variable are more important to the end-user, but severely under-represented in the training data. As training data we assume a set of cases built using a time delay embedding strategy, *i.e.* where the target variable is the value of Y in the next time step (y_{t+1}) and the predictors are the k recent values of the time series, *i.e.* $y_t, y_{t-1}, \cdots, y_{t-k}$. To formalise our prediction task, namely in terms of criteria for evaluating the results of modeling approaches, we need to specify what we mean by "more important" values of the target variable. We resort to the work of Ribeiro [15], that proposes the use of a relevance function to map the domain of continuous variables into a $[0, 1]$ scale of relevance, *i.e.* $\phi(Y) : \mathcal{Y} \to [0, 1]$. Normally, this function is given by the users, attributing levels of importance to ranges of the target variable specific to their interest, taking into consideration the domain of the data. In our work, due to the lack of expert knowledge concerning the domains, we employ an automatic approach to define the relevance function using box plot statistics, detailed in Ribeiro [15]. Using this domain-dependent function the author has also proposed an evaluation framework that allows us to assert the quality of numeric predictions considering the user bias.

We use the above-mentioned evaluation framework to ascertain the predictive accuracy when using imbalanced time series data, by combining standard learning algorithms and resampling strategies. Additionally, we evaluate the most consistently accurate strategies resorting to paired comparisons between the proposals. The assumptions that we test in the evaluation process are:

Assumption 1 *The use of resampling strategies significantly improves the predictive accuracy of imbalanced time series forecasting models in comparison to the standard use of out of the box regression tools.*

Assumption 2 *The use of a temporal bias in resampling strategies significantly improves the predictive accuracy of imbalanced time series forecasting models in comparison to the baseline versions of each respective strategy.*

Assumption 3 *The use of resampling strategies significantly improves the predictive accuracy of imbalanced time series forecasting models in comparison to the use of ARIMA models.*

III. Resampling Strategies

Resampling strategies are pre-processing approaches that change the original data distribution in order to meet some user-given criteria. Among the advantages of pre-processing strategies is the ability of using any standard learning tool. However, to match a change in the data distribution with the user preferences is not a trivial task. The proposed resampling strategies aim at pre-processing the data for obtaining an increased predictive performance in cases that are scarce and simultaneously important to the user. As mentioned before, this importance is described by a relevance function $\phi(Y)$. Being domain-dependent information, it is the user responsibility to specify the relevance function. Nonetheless, when lacking expert knowledge, it is possible to automatically generate the relevance function. Being a continuous function on the scale $[0, 1]$, we require the user to specify a relevance threshold, t_R, that establishes the minimum relevance score for a certain value of the target variable to be considered relevant. This threshold is only required because the proposed resampling algorithms need to be able to decide which values are the most relevant when the distribution changes.

Figure 2 shows an example of an automatically generated relevance function, with a 0.9 relevance threshold, defined for the Temperature time series (Figure 1) obtained from the Bike Sharing data source [16] using observations between 22 March and 1 May 2011. In this example, we assign more importance to the highest and lowest values of Y.

Our resampling strategies proposals for imbalanced time series data are based on the concept of relevance bins. These are successive observations of the time series where the observed value is either relevant or irrelevant, for the user. The bins are created using time stamp information and the relevance of the values from the original time series, to cluster them into bins that have the following properties:

1) Each bin contains examples whose target variable value has a relevance score that is either all above or all below the relevance threshold t_R; and
2) Examples in a given bin are always consecutive cases in terms of the time stamp.

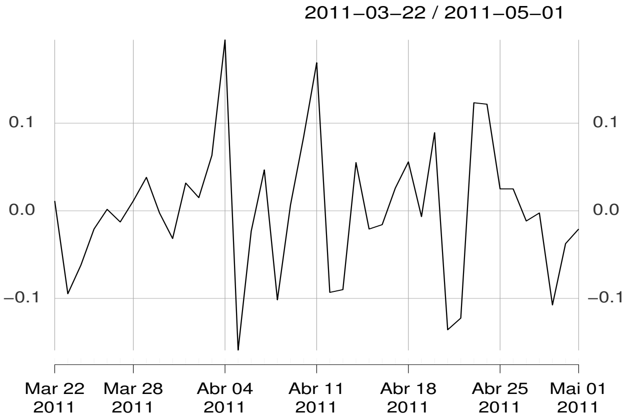

Fig. 1. Sample of Temperature time series from the Bike Sharing data source [16].

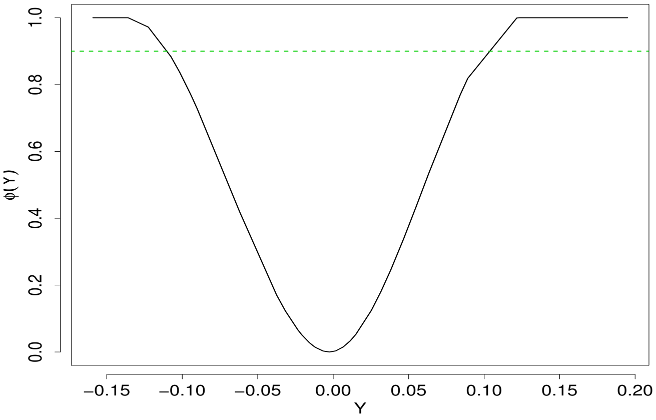

Fig. 2. Relevance function $\phi(Y)$ with a relevance threshold of 0.9 (dashed line) for the time series showed in Figure 1

Figure 3 shows the bins obtained in the Temperature time series displayed in Figure 1. The six dashed Y ranges represent the most relevant ranges, while the non-dashed ranges represent the common values which have a lower relevance to the user, based on the automatically generated relevance function (Figure 2). This means that, for the example under consideration, we have 13 bins: 3 bins with relevant high values and 3 bins with relevant low values (dashed ranges), and 7 bins with common values (non-dashed ranges). The relevance bins are successive periods where the target variable is either irrelevant or relevant to the user.

Our first proposals are an adaption to the time series context of the random undersampling and SmoteR strategies proposed by Torgo et al. [8] for tackling imbalanced regression tasks. The main change applied in both algorithms is the way the sampling is carried out. Instead of pure random selection as in the original algorithms, here we carry out sampling within each individual relevance bin.

The random undersampling (**U_B**) strategy balances the number of normal and rare values by randomly removing examples from the bins with normal cases, *i.e.*, bins with low relevance examples. The number of examples removed is automatically calculated to ensure that: 1) each undersampled bin gets the same number of normal cases; and 2) the total number of normal and rare cases are balanced.

The second strategy is an adaptation of the SmoteR algorithm to the time series context. The SmoteR algorithm combines random undersampling with oversampling through the generation of synthetic cases. The random undersampling is carried out as the previously described process. The oversampling strategy generates new synthetic cases by interpolating a seed example with one of its k-nearest neighbours from the respective bin of rare examples. The main changes introduced in comparison to the original SmoteR algorithm (**SM_B**) are: 1) the application of the oversampling and undersampling techniques in the bins constructed for the time series based on characteristics of the bins; 2) the number of cases to remove/add are determined automatically with the goal of balancing the distribution of rare and normal cases; and 3) the number of nearest neighbours set by the user and used for a given example is k. If an example does not have k nearest neighbours, it interpolates with the remaining examples of bin.

A. *Resampling with Temporal Bias*

As we have mentioned, concept drift is one of the main challenges in time series forecasting. This is particularly true for our target applications where the preference bias of the user is related with rare values of the series. In effect, this rarity makes it even more important to understand and anticipate when these shifts of regime occur.

A first step in the identification of these different regimes according to user preferences is implemented by the previously described creation of relevance bins (c.f. Figure 3). Still, within each bin the cases are not equally relevant. We claim that the most recent cases within each bin may potentially contain important information for understanding these changes in regime. In this context, we propose two new algorithms (Undersampling and SmoteR with Temporal Bias) that favour the selection of training cases that are in the vicinity of transitions between bins. This resembles the adaptive learning notion of gradual forgetting, where the older cases have a higher likelihood of being excluded from the learning data. However, this concept is applied to the extent of the data and

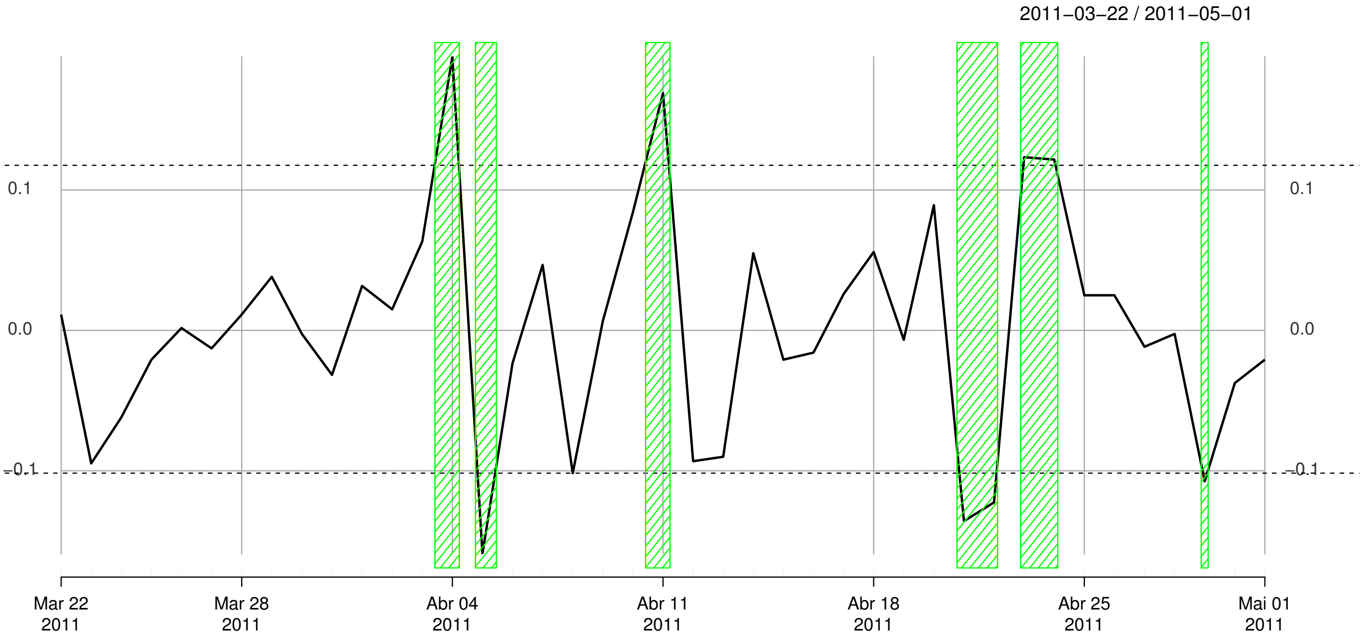

Fig. 3. Bins generated for time series of Figure 1 with relevance function ($\phi()$) provided in Figure 2 using a relevance threshold of 0.9 (dashed ranges represent bins with important cases).

in our proposal of temporal bias it is applied in each bin of normal cases.

Concerning the Undersampling with Temporal Bias (**U_T**) proposal, the main difference is the process of selecting examples to undersample within each bin of normal cases. Instead of randomly selecting cases, we use a biased undersampling procedure. In **U_T**, for each bin where undersampling is applied, the older the example is, the lower is the preference of it being selected for the new training set. This provides a modified distribution which is balanced in terms of normal and rare cases with a probabilistic preference towards the most recent cases, *i.e.* those in the vicinity of bin transitions. The integration of the temporal bias is performed as follows:

- order the cases in each bin B of normal cases by increasing time in a new bin $OrdB$;
- assign the preference of $i \times \frac{1}{|OrdB|}$ for selecting ex_i in $OrdB$, where $i \in (1, \ldots, |OrdB|)$;
- select a sample from $OrdB$ based on the former preferences.

Our second proposed strategy is SmoteR with Temporal Bias (**SM_T**). This approach combines undersampling with temporal bias in the bins containing normal cases, with an oversampling mechanism that also integrates a temporal component. The undersampling with temporal bias strategy is previously described. Regarding the oversampling strategy, we included in the SmoteR generation of synthetic examples a preference for the most recent examples. This means that when generating a new synthetic case, after evaluating the k-nearest neighbours of the seed example, the neighbour selected for the interpolation process is the most recent case. This includes, in the synthetic cases generation, a time bias towards the most recent examples instead of randomly selecting cases.

B. Resampling with Temporal and Relevance Bias

This section describes our final proposals of resampling strategies for imbalanced time series forecasting. The idea of the two algorithms described in this section is to also include the relevance scores in the sampling bias. The motivation is that while we assume that the most recent cases within each bin are important as they precede regime changes, we consider that older cases that are highly relevant should not be completely disregarded. To combine the temporal and relevance bias we propose two new algorithms: undersampling with temporal and relevance bias and SmoteR with temporal and relevance bias.

The integration of temporal and relevance bias in undersampling (**U_TPhi**) is performed as follows:

- order examples in each bin B of normal cases by increasing time in a new bin $OrdB$
- for each example ex_i in $OrdB$ use $\frac{i}{|OrdB|} \times \phi(ex_i[y])$ as the preference of selecting example ex_i;
- sample a number of examples from $OrdB$ assuming the previously determined preferences.

The same integration of time and relevance bias was also done in SmoteR algorithm. In this case, we have changed both the undersampling and oversampling steps of SmoteR algorithm. These changes correspond to biasing the undersampling process for considering the time and relevance of the examples in each bin, as previously described: the most recent examples with higher relevance are preferred to others for staying in the changed data set. Regarding the oversampling strategy, the generation of synthetic examples also assumes this tendency, *i.e.*, the new examples are built by prioritising the selection of highly relevant and recent examples. The bias towards more recent and high relevance examples is achieved in the selection of a nearest neighbour for the interpolation, as follows:

- calculate the relevance of the k-nearest neighbours;
- calculate the time position of k-nearest neighbours by ascending order and normalized to $[0, 1]$;
- select the nearest neighbour with the highest value of the product of relevance by time position.

In summary, for each of the two resampling strategies considered (random undersampling and SmoteR), we have proposed three new variants that try to incorporate some form of sampling bias that we hypothesize as being advantageous in terms of forecasting accuracy on imbalanced time series tasks where the user favours the performance on rare values of the series. The first variants (**U_B** and **SM_B**) carry out sampling within relevance bins that are obtained with the goal of including successive cases with similar relevance according to the user preference. The second variants (**U_T** and **SM_T**) add to the first variant a preference toward the most recent cases within each bin as these are the cases that precede regime transitions. Finally, the third variants (**U_TPhi** and **SM_TPhi**) add a third preference to the sampling procedures, to also include the relevance scores of the cases and avoid discarding cases that may not be the most recent, but are the most relevant for the user.

IV. Materials and Methods

A. Data

The experiments described in this paper use data from 6 different sources, totaling 24 time series from diverse real-world domains. For the purposes of evaluation we assumed that each time series is independent from others of the same source (*i.e.* we did not use the temperature time series data in the Bike Sharing source to predict the count of bike rentals). All proposed resampling strategies, in combination with each of the regression tools, are tested on these 24 time series which are detailed in Table I. All of the time series were pre-processed to overcome some well-known issues with this type of data, as is non-available (*NA*) observations. To resolve issues of this type, we resorted to the imputation of values using the **R** function **knnImputation** of the package **DMwR** [17]. For each of these time series data sets we applied the previously described approach of the time delay coordinate embedding. It requires an essential parameter: how many values to include as recent values, *i.e.* the size of the embed, k. This is not a trivial task as it requires to try different values of embed size in order to decide on an acceptable value. In our experiments we have used $k = 10$. Experiments with a few other values have not shown significant differences in results. The outcome of the application of this embedding approach produces the data sets we use as learning data.

For each of these data sets we need to decide which are the relevant ranges of the time series variable. To this purpose, we use a relevance function. As previously mentioned, due to the lack of expert knowledge concerning the used domains, we resort to an automatic approach to define the relevance function, detailed in Ribeiro [15]. This approach uses box plot statistics to derive a relevance function that assigns higher relevance scores to values that are unusually high or low, *i.e.* extreme and rare values. We use this process to obtain the relevance functions for all our time series. An example of the application of this approach, where only high extreme values exist, is depicted in Figure 4, and in Figure 1 a case with two extremes is shown. Having defined the relevance functions we still need to set a threshold on the relevance scores above which a value is considered important, *i.e.*, the relevance threshold t_R. The definition of this parameter is domain dependent. Still, we have used a relevance threshold t_R of 0.9, which generally leads to a small percentage of the values to be considered important. In Table I we added an indication concerning the proportion of rare cases (both very high and low values).

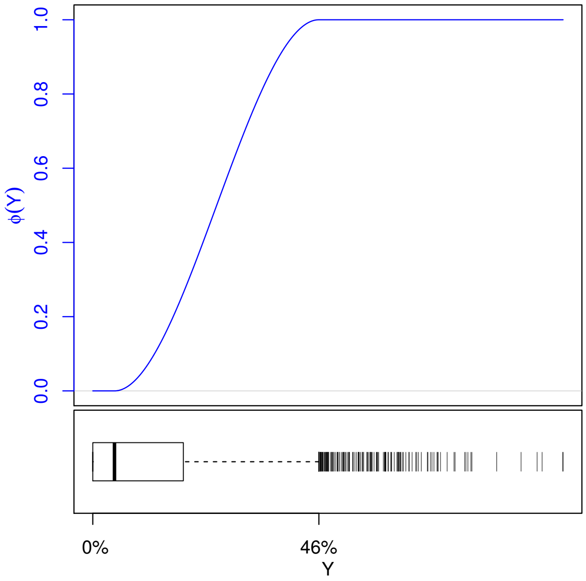

Fig. 4. Relevance function $\phi()$ with high extreme values and box plot of Y distribution.

B. Regression Algorithms

In order to test our assumptions we selected a diverse set of standard regression tools. Our goal is to verify that our conclusions are not biased by the choice of a particular tool.

Table II shows the regression methods used in our experiments. To ensure that our work is easily replicable we used the implementations of these tools available in the free and open source **R** environment. All tools were applied using their default parameter values. In addition to these standard regression tools, we also include the ARIMA model [13], which is considered to be a standard time series forecasting approach. Since ARIMA models also require a significant tuning effort in terms of parameters, we used the **auto.arima** function available in the R package **forecast** [18], which implements an automatic search method for the optimal parameter settings.

TABLE I. DESCRIPTION OF THE DATA SETS USED.

ID	Time Series	Data Source	Granularity	Characteristics	% Rare
DS1	Temperature	Bike Sharing [16]	Daily	From 01/01/2011 to 31/12/2012 (731 values)	9.9%
DS2	Humidity				9.3%
DS3	Windspeed				7.8%
DS4	Count of Bike Rentals				13.3%
DS5	Temperature		Hourly	From 01/01/2011 to 31/12/2012 (7379 values)	3.5%
DS6	Humidity				4.8%
DS7	Windspeed				12.5%
DS8	Count of Bike Rentals				17.6%
DS9	Flow of Vatnsdalsa River	Icelandic River [19]	Daily	From 01/01/1972 to 31/12/1974 (1095 values)	21.1%
DS10	Minimum Temperature	Porto weather[1]	Daily	From 01/01/2010 to 28/12/2013 (1457 values)	4.8%
DS11	Maximum Temperature				13.3%
DS12	Maximum Steady Wind				11%
DS13	Maximum Wind Gust				11.1%
DS14	SP	Istanbul Stock Exchange [20]	Daily	From 05/01/2009 to 22/02/2011 (536 values)	16.3%
DS15	DAX				11.4%
DS16	FTSE				9.7%
DS17	NIKKEI				11.6%
DS18	BOVESPA				10.1%
DS19	EU				8.2%
DS20	Emerging Markets				6.8%
DS21	Total Demand	Australian electricity load [21]	Half-Hourly	From 01/01/1999 to 01/09/2012 (239602 values)	1.8%
DS22	Recommended Retail Price				10.2%
DS23	Pedrouços	Water Consumption of Oporto[2]	Half-Hourly	From 06/02/2013 to 11/01/2016 (51208 values)	0.08%
DS24	Rotunda AEP				3.4%

[1] Source: Freemeteo http://freemeteo.com.pt/

[2] Source: Águas do Douro e Paiva http://addp.pt/

TABLE II. REGRESSION ALGORITHMS AND RESPECTIVE R PACKAGES

ID	Method	R package
LM	Multiple linear regression	stats [22]
SVM	Support vector machines	e1071 [23]
MARS	Multivariate adaptive regression splines	earth [24]
RF	Random forests	randomForest [25]

C. Evaluation Metrics

It is known that when the interest of the user is predictive performance at a small proportion of cases (*i.e.* rare cases), the use of standard performance metrics will lead to biased conclusions [15]. In effect, standard metrics focus on the "average" behaviour of the prediction models and for the tasks addressed in this paper, the user goal is a small proportion of cases. Although most of the previous studies on this type of issues are focused on classification tasks, Torgo and Ribeiro [26], [15] have shown that the same problems arise on regression tasks when using standard metrics, such as Mean Squared Error. Moreover, these authors have shown that discretizing the target numeric variable into a nominal variable followed by the application of classification algorithms is also prone to problems and leads to sub-optimal results.

In this context, we will base our evaluation on the utility-based regression framework proposed in the work by Torgo and Ribeiro [26], [15] which assumes the existence of a relevance function ϕ, as previously described. Using this approach and the user-provided relevance threshold, the authors defined a series of metrics that focus the evaluation of models on the cases that the user is interested. In our experiments we used the value 0.9 as relevance threshold.

The evaluation process of the prediction models will mainly rely on one utility-based regression metric: F-Score. This is a composite measure that integrates the values of precision and recall according to their adaptation for regression described in the above mentioned utility-based evaluation framework.

V. EXPERIMENTAL EVALUATION

This section presents the results of our experimental evaluation on three sets of experiments concerning forecasting tasks with imbalanced time series data sets. Each of these experiments were designed with the objective of testing the assumptions set forth in Section II. In the first set we evaluate the predictive accuracy of standard regression tools in combination with the proposed resampling strategies. In the second set of experiments, the evaluation is focused on the task of inferring the possibility of the biased resampling strategies over-performing the non-biased strategies. Finally, in the third set, we evaluate the assumption of enabling a better predictive performance of models using standard regression tools with resampling strategies over ARIMA models, which are considered a standard time series forecasting approach. The ARIMA models and all of the proposed resampling strategies combined with each of the standard regression tools were tested on 24 real-world time series data sets, obtained from six different data sources described in Table I. As previously stated, we will base our evaluation process in the evaluation metric F1-Score as described by the referred utility-based regression framework, detailed in Section IV-C. Concerning the testing of our assumptions, we resort to paired comparisons using Wilcoxon signed rank tests in order to infer the statistical significance (with $p-$value < 0.05) of the paired differences in the outcome of the approaches.

Concerning evaluation algorithms, caution is required in the decision on how to obtain reliable estimates of the evaluation metrics. Since time series data are temporally ordered, we must ensure that the original order of the cases is maintained as to guarantee that prediction models are trained with past data and tested with future data, thus avoiding over-fitting and over-estimated scores. As such, we rely on Monte Carlo estimates as the chosen experimental methodology for our evaluation. This methodology selects a set of random points in the data. For each of these points a past window is selected as training data (*Tr*) and a subsequent window as test data (*Ts*). This methodology guarantees that each method used in our forecasting task is evaluated using the same training and test sets, thus ensuring a fair pairwise comparison of the estimates obtained. In our evaluation 50 repetitions of the Monte Carlo estimation process are carried out for each data set with 50% of the cases used as training set and the subsequent 25% used as test set. Exceptionally, due to their size, in the case of the data sets *DS21* and *DS22* we used 10% of the cases as training set and the following 5% as test set, and 20% of the cases as training set and the following 10% as test set for data sets *DS23* and *DS24*. This process is carried out using the infrastructure provided by the R package **performanceEstimation** [27].

In order to clarify the nomenclature associated with the standard regression tools used in this evaluation process, the experiments include results given by multiple linear regression (**LM**), support vector machine (**SVM**), multivariate adaptive regression splines (**MARS**) and random forest (**RF**) models. As for the resampling strategies, we use random undersampling (**U_B**), SmoteR (**SM_B**), undersampling (**U_T**) and SmoteR (**SM_T**) with temporal bias, and undersampling (**U_TPhi**) and SmoteR (**SM_TPhi**) with temporal and relevance bias. The overall results given the F1-Score evaluation metric proposed by Ribeiro [15], obtained by Monte Carlo estimates, concerning all 24 time series data sets are presented in Table III. Results in bold show the best approach within each group of resampling strategies. The non-resampled approaches are denoted in bold when their result is as good or better then one of the best results obtained by approaches employing resampling strategies (also denoted in bold), with respect to a given standard regression tool. The ARIMA models are denoted in bold when their result is as good or better than one of the best results obtained by approaches employing resampling strategies overall, for a given data set.

From the obtained results, we observe that the application of resampling strategies shows great potential in terms of boosting the performance of forecasting tasks using imbalanced time series data. This is observed within each of the standard regression tools used (vertical analysis), but also regarding the data sets used (horizontal analysis), where it is clear that the approaches employing resampling strategies obtain the best results according to the averaged F1-Score evaluation metric.

A. Assumption 1

The first assumption brought forth in our work proposes that the use of resampling strategies significantly improves the predictive accuracy of imbalanced time series forecasting tasks in comparison to the use of standard regression tools. Although results presented in Table III point to the confirmation of this assumption, it still remains unclear the degree of statistical significance concerning the difference in evaluation between the use or non-use of resampling strategies combined with standard regression tools.

Table IV presents the paired comparisons of the application of random undersampling (**U_B**) and SmoteR (**SM_B**), and the standard regression tools without any applied resampling strategy. The information in the columns represents the number of wins and losses for each approach against the baseline. In this case, the baseline represents the regression tools without the application of resampling strategies.

It is shown that the use of resampling strategies adds a significant boost in terms of forecasting relevant cases in imbalanced time series data, when compared to its non-use, in all standard regression tools employed in the experiment. Therefore, these experiments provide strong empirical evidence to confirm our first assumption.

B. Assumption 2

The second assumption states that the use of a temporal and/or relevance bias in resampling strategies significantly improves the predictive accuracy of time series forecasting tasks in comparison to the baseline versions of each respective strategy. In order to prove this assumption, results in Table V present the paired comparisons of the application of the resampling strategies **U_T**, **U_TPhi**, **SM_T** and **SM_TPhi**, against the respective resampling strategies **U_B** and **SM_B**, for each standard regression tool. For this experiment set, the baseline is defined as being the application of random undersampling and SmoteR in their initial adaptation to imbalanced time series.

Results show an overall advantage of the use of temporal and/or relevance bias in the case selection process of the resampling strategies used in our experiments. In the case of **U_T** and **SM_T**, we observe similar results when combined with the **SVM** models in comparison with the baseline, with an advantage to the latter. Nonetheless, the application of the temporal and relevance bias approach in the resampling strategies shows a clearly superior performance independently of the regression tool employed, thus showing strong empirical evidence to confirm our second assumption.

C. Assumption 3

The third assumption proposed in our work is that the use of resampling strategies significantly improves the predictive accuracy of time series forecasting tasks in comparison to the use of ARIMA models. As previously referred, the ARIMA models are commonly pointed as a standard approach to time series forecasting. In this context, we want to check if our proposals based on resampling are able to significantly improve the predictive performance of these models. We stress again that in this evaluation we employed a version of the ARIMA models which automatically searches for the optimal number of past values to build the embed, while the standard regression tools are used with their default parameter settings and only enhanced through our resampling strategies.

TABLE III. EVALUATION RESULTS OF BASELINE REGRESSION ALGORITHMS AND THE APPLICATION OF THE RESAMPLING STRATEGIES OVER 24 DATA SETS, GIVEN BY THE AVERAGE OF THE UTILITY-BASED REGRESSION METRIC F1-SCORE.

Model	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9	DS10	DS11	DS12
lm	0.027	0.027	0.154	0.219	0.000	0.000	0.027	0.244	0.100	0.161	0.033	0.147
U_B	0.208	0.434	0.307	0.383	0.161	0.069	0.204	0.352	0.099	0.418	0.141	0.382
U_T	**0.210**	0.424	0.308	0.380	0.163	0.070	0.205	0.343	**0.116**	0.416	0.152	0.391
U_TPhi	0.207	**0.439**	**0.318**	**0.389**	**0.170**	**0.082**	**0.225**	**0.434**	0.113	**0.467**	**0.159**	**0.402**
SM_B	0.231	**0.443**	0.330	0.412	0.203	0.146	0.276	0.389	0.118	0.416	**0.191**	0.437
SM_T	0.241	0.416	**0.340**	0.419	0.200	0.170	0.306	0.388	0.166	0.417	0.183	0.439
SM_TPhi	**0.266**	0.423	0.336	**0.421**	**0.206**	**0.177**	**0.319**	**0.436**	**0.167**	**0.461**	0.186	**0.452**
svm	0.107	0.000	0.063	0.082	0.021	0.000	0.027	0.496	0.083	0.196	0.053	0.051
U_B	0.162	**0.256**	0.179	0.230	**0.221**	0.181	0.238	**0.526**	**0.278**	0.394	0.168	**0.261**
U_T	0.175	0.254	0.170	0.246	0.218	0.187	0.244	0.524	0.277	0.393	0.165	0.252
U_TPhi	**0.179**	0.243	**0.198**	**0.260**	0.217	**0.234**	**0.294**	0.512	0.268	**0.464**	**0.186**	0.260
SM_B	0.171	0.283	**0.221**	0.274	**0.235**	0.269	0.313	**0.539**	**0.292**	0.279	0.225	0.315
SM_T	0.214	0.294	0.211	**0.294**	0.225	0.259	0.324	0.535	0.226	0.290	0.230	0.305
SM_TPhi	**0.229**	**0.297**	0.211	0.292	0.223	**0.279**	**0.349**	0.527	0.216	**0.375**	**0.253**	**0.318**
mars	0.044	0.089	0.192	0.213	0.005	0.000	0.044	0.406	0.116	0.172	0.067	0.162
U_B	0.204	**0.299**	0.236	0.341	0.191	0.097	0.228	0.457	0.142	0.393	0.111	0.349
U_T	0.228	0.294	0.239	**0.367**	0.193	0.096	0.232	0.458	0.138	0.396	0.120	0.341
U_TPhi	**0.243**	0.291	**0.282**	0.355	**0.200**	**0.115**	**0.250**	**0.461**	**0.150**	**0.466**	**0.140**	**0.352**
SM_B	0.251	0.369	0.325	0.400	**0.236**	0.184	0.296	0.479	0.144	0.387	0.211	**0.410**
SM_T	0.293	0.361	0.315	0.402	0.219	0.193	0.323	0.494	**0.175**	0.372	0.224	0.397
SM_TPhi	**0.307**	**0.393**	**0.349**	**0.405**	0.223	**0.201**	**0.333**	**0.507**	0.173	**0.430**	**0.229**	0.396
rf	0.010	0.010	0.000	0.198	0.032	0.000	0.060	0.476	**0.150**	0.112	0.043	0.041
U_B	0.142	0.122	0.079	0.260	0.201	0.119	0.222	**0.520**	**0.150**	0.381	**0.102**	**0.164**
U_T	0.133	0.124	0.080	**0.270**	**0.207**	0.118	0.225	0.517	0.143	0.378	0.094	0.159
U_TPhi	**0.148**	**0.131**	**0.088**	0.267	0.203	**0.142**	**0.245**	0.499	0.145	**0.472**	0.096	**0.164**
SM_B	0.151	0.181	**0.096**	0.309	0.129	0.095	0.206	**0.521**	0.156	0.234	0.119	0.203
SM_T	0.169	0.213	0.083	0.323	0.115	0.087	0.220	**0.521**	0.163	0.225	0.133	0.206
SM_TPhi	**0.180**	**0.224**	0.084	**0.329**	**0.138**	**0.125**	**0.257**	0.508	**0.170**	**0.418**	**0.136**	**0.233**
ARIMA	0.015	0.000	**0.158**	0.231	0.000	0.000	0.037	0.184	**0.147**	0.179	0.039	0.137

	DS13	DS14	DS15	DS16	DS17	DS18	DS19	DS20	DS21	DS22	DS23	DS24
lm	0.146	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.218	0.394	0.123	0.417
U_B	0.419	0.059	0.026	0.152	0.057	**0.035**	**0.158**	0.012	**0.219**	0.508	0.168	0.453
U_T	0.426	**0.085**	0.049	**0.187**	0.053	0.030	0.154	**0.032**	**0.219**	**0.509**	0.170	0.455
U_TPhi	**0.445**	0.068	**0.051**	0.161	**0.079**	0.031	0.148	0.024	0.218	0.505	**0.174**	**0.458**
SM_B	0.466	0.145	0.062	**0.204**	**0.183**	0.067	0.197	0.020	**0.219**	**0.519**	0.164	0.456
SM_T	0.462	0.157	**0.119**	0.170	0.135	0.107	0.215	**0.063**	**0.219**	0.351	0.163	0.462
SM_TPhi	**0.476**	**0.164**	0.112	0.183	0.150	**0.117**	**0.218**	0.053	0.218	0.344	**0.168**	**0.472**
svm	0.109	0.006	0.000	0.000	0.000	0.000	0.000	0.000	0.216	**0.484**	0.176	0.427
U_B	0.316	**0.109**	0.006	0.099	0.059	0.024	0.061	0.002	**0.218**	0.405	0.270	0.470
U_T	0.302	0.080	0.002	**0.103**	0.060	0.016	0.051	0.004	**0.218**	0.406	0.256	**0.472**
U_TPhi	**0.344**	0.107	**0.031**	0.093	**0.127**	**0.033**	**0.084**	**0.014**	0.217	**0.412**	**0.282**	0.453
SM_B	0.371	**0.205**	0.033	0.144	0.142	0.073	**0.080**	0.007	**0.217**	**0.410**	0.180	0.469
SM_T	0.348	0.161	0.064	0.113	0.160	**0.080**	0.062	0.006	**0.217**	0.305	0.165	**0.477**
SM_TPhi	**0.372**	0.191	**0.074**	**0.153**	**0.187**	0.078	**0.080**	**0.012**	**0.217**	0.324	**0.198**	0.453
mars	0.132	0.018	0.000	0.008	0.000	0.000	0.004	**0.020**	**0.218**	**0.362**	0.155	0.423
U_B	0.372	0.117	0.022	0.080	0.067	0.034	0.026	0.000	**0.218**	0.350	**0.224**	0.474
U_T	0.386	**0.166**	0.029	0.080	0.070	**0.045**	0.014	0.010	**0.218**	0.354	0.221	**0.475**
U_TPhi	**0.391**	0.126	**0.038**	**0.084**	**0.085**	0.034	**0.041**	**0.015**	**0.218**	**0.368**	0.221	0.454
SM_B	0.423	**0.242**	0.098	**0.226**	0.205	0.144	0.136	0.007	**0.218**	**0.345**	0.178	0.473
SM_T	0.414	0.232	0.094	0.196	0.193	0.145	0.179	0.059	**0.218**	0.303	0.164	**0.482**
SM_TPhi	**0.429**	0.240	**0.113**	0.195	**0.217**	**0.179**	**0.199**	**0.063**	0.217	0.322	**0.197**	0.464
rf	0.098	0.000	0.000	0.004	0.000	0.000	0.000	0.000	0.215	0.398	0.179	0.429
U_B	0.193	0.036	**0.004**	0.058	0.002	**0.037**	0.023	**0.003**	**0.225**	**0.428**	0.161	0.476
U_T	0.205	**0.048**	0.000	**0.063**	**0.008**	0.028	0.028	0.000	**0.225**	**0.428**	0.170	**0.480**
U_TPhi	**0.208**	0.039	0.003	0.058	0.007	0.033	**0.057**	0.000	0.218	0.409	**0.223**	0.452
SM_B	0.230	**0.084**	0.006	**0.097**	0.060	0.071	**0.050**	0.000	0.217	**0.423**	0.239	0.464
SM_T	0.240	0.075	0.004	0.049	0.061	0.074	0.035	**0.016**	**0.218**	0.388	0.228	**0.479**
SM_TPhi	**0.261**	**0.084**	**0.014**	0.045	**0.089**	**0.100**	0.028	**0.016**	0.217	0.377	**0.242**	0.462
ARIMA	0.146	0.000	0.000	0.000	0.000	0.000	0.000	0.000	**0.218**	**0.387**	0.148	0.427

TABLE IV. PAIRED COMPARISONS RESULTS OF EACH REGRESSION ALGORITHM BASELINE WITH THE APPLICATION OF RESAMPLING STRATEGIES, IN THE FORMAT NUMBER OF WINS (STATISTICALLY SIGNIFICANT WINS) / NUMBER OF LOSSES (STATISTICALLY SIGNIFICANT LOSSES).

	LM	SVM	MARS	RF
U_B	22 (21) / 2 (1)	24 (21) / 0 (0)	23 (19) / 1 (0)	23 (19) / 1 (0)
SM_B	23 (23) / 1 (1)	23 (21) / 1 (0)	23 (21) / 1 (1)	22 (21) / 2 (0)

TABLE V. PAIRED COMPARISONS RESULTS OF EACH REGRESSION ALGORITHM WITH BASELINE RESAMPLING STRATEGIES AND THE APPLICATION OF BIASED RESAMPLING STRATEGIES, IN THE FORMAT NUMBER OF WINS (STATISTICALLY SIGNIFICANT WINS) / NUMBER OF LOSSES (STATISTICALLY SIGNIFICANT LOSSES).

	LM.U_B	SVM.U_B	MARS.U_B	RF.U_B
U_T	16 (7) / 8 (1)	11 (1) / 13 (1)	19 (3) / 5 (0)	14 (1) / 10 (2)
U_TPhi	19 (10) / 5 (2)	15 (9) / 9 (6)	19 (8) / 5 (1)	17 (5) / 7 (3)

	LM.SM_B	SVM.SM_B	MARS.SM_B	RF.SM_B
SM_T	14 (7) / 10 (2)	11 (5) / 13 (8)	13 (8) / 11 (4)	15 (8) / 9 (5)
SM_TPhi	18 (12) / 6 (3)	12 (8) / 12 (6)	16 (11) / 8 (2)	17 (13) / 7 (4)

The results from the paired comparisons of all the approaches employing resampling strategies and the ARIMA models (considered the baseline) are presented in Table VI.

Results show that independently of the regression tool used, the application of resampling strategies provides a highly significant improvement over the results obtained by the ARIMA models. This goes to show the validity of our third and final assumption. Additionally, we also observe that results show a consistent advantage for the temporal and relevance bias

TABLE VI. Paired comparisons results of ARIMA and the application of Resampling Strategies in each Regression algorithm, in the format Number of Wins (Statistically Significant Wins) / Number of Losses (Statistically Significant Losses).

Algorithm	Strategy	ARIMA
LM	U_B	22 (21) / 2 (2)
	U_T	23 (23) / 1 (1)
	U_TPhi	22 (22) / 2 (2)
	SM_B	22 (22) / 2 (2)
	SM_T	23 (22) / 1 (1)
	SM_TPhi	23 (22) / 1 (1)
SVM	U_B	23 (18) / 1 (0)
	U_T	24 (18) / 0 (0)
	U_TPhi	23 (21) / 1 (0)
	SM_B	24 (20) / 0 (0)
	SM_T	22 (20) / 2 (0)
	SM_TPhi	22 (22) / 2 (1)
MARS	U_B	21 (19) / 3 (1)
	U_T	21 (19) / 3 (1)
	U_TPhi	21 (20) / 3 (0)
	SM_B	21 (20) / 2 (2)
	SM_T	22 (20) / 2 (2)
	SM_TPhi	22 (21) / 2 (2)
RF	U_B	22 (18) / 2 (2)
	U_T	20 (18) / 4 (1)
	U_TPhi	20 (18) / 4 (1)
	SM_B	20 (19) / 4 (2)
	SM_T	21 (19) / 3 (1)
	SM_TPhi	21 (19) / 3 (2)

approach in combination with the SmoteR resampling strategy, in all regression tools.

VI. Related Work

Despite an extensive research process, we did not find any previous work that proposes the use of resampling strategies for forecasting tasks with imbalanced time series data. However, we found different approaches related to the scope of our endeavour, in the problems of rare event forecasting and anomaly detection, which we describe below.

A genetic-base machine learning system, *timeweaver*, was proposed by Weiss and Hirsh [28], designed to address rare event prediction problems with categorical features, by identifying predictive temporal and sequential patterns. The genetic algorithm used is responsible for updating a set of prediction patterns, where each individual should perform well at classifying a subset of the target events and which collectively should cover most of those events.

Vilalta and Ma [29] proposed an algorithm to address prediction of rare events in imbalanced time-series. The authors proposed to resolve the class-imbalance by transforming the event prediction problem into a search for all frequent event sets (patterns) preceding target events, focused solely on the minority class. These patterns are then combined into a rule-based model for prediction. Both the work of Weiss and Hirsh [28] and of Vilalta and Ma [29] assume that events are characterized by categorical features and display uneven inter-arrival times. However, this is not assumed in classical time-series analysis.

Temporal sequence associations are used by Chen et al. [30] for predicting rare events. The authors propose a heuristic for searching interesting patterns associated with rare events in large temporal event sequences. The authors combine association and sequential pattern discovery with a epidemiology-based measure of risk in order to assess the relevance of the discovered patterns.

In anomaly detection [31] problems, applications for several domains have been proposed using diverse techniques. In the Medical and Public Health Domain, Lin et al. [32] use nearest neighbor based techniques to detect these rare cases. These same techniques are used by Basu and Mackenshimer [33] and parametric statistical modelling is used by Keogh et al. [34] in the domain of mechanical units fault detection. Finally, Scott [35] and Ihler et al. [36] propose Poisson-based analysis techniques for the respective domains of intrusion detection in telephone networks and Web Click data.

VII. Conclusions

In this work we study the application of resampling strategies with imbalanced time series data. Our overall objective is to enhance the predictive accuracy on rare and relevant cases as this is the objective in several application domains. This fact increases the interest in finding ways to significantly improve the predictive accuracy of prediction models in these tasks.

In this context, we have proposed the extension of existing resampling methods to time series forecasting tasks. Resampling methods can be used to change the distribution of the available learning sets with the goal of biasing learning algorithms to the cases that are more relevant to the users. Our proposals build upon prior work on resampling methods for numeric prediction tasks. Besides the extension of known resampling strategies, we propose new resampling strategies with the goal of adapting them to the specific characteristics of time series data. Specifically, we have proposed sampling strategies that introduce a temporal bias that we claim to be useful when facing non-stationary time series that are frequently subjected to concept drift. We also propose a relevance bias that makes more relevant cases have a higher probability of being selected for the final training sets.

An extensive set of experiments was carried out to ascertain the advantages of applying resampling strategies to such problems. Results from the experimental evaluation show that we were able to significantly improve the predictive accuracy of the models, focusing on rare and relevant cases of imbalanced time series data. Results show that *1)* the application of resampling strategies in combination with standard regression tools significantly improves the ability to predict rare and relevant cases in comparison to not applying these strategies; *2)* the use of a temporal and/or relevance bias improves the results in relation to the non-biased resampling approaches; and *3)* the combination of resampling approaches with standard regression tools provide a significant advantage in comparison to the time series focused ARIMA models.

Concerning future work, we plan to further evaluate these proposals concerning the effect of parameters values such as the relevance threshold or the k number of nearest neighbours in SmoteR, and study ways of automatically adapting these parameters to the distribution. We also plan to generalize the concept of bias in resampling strategies as to study the possibility of its use not only in time series problems, but also in classification and regression tasks using various types of dependency-oriented data, such as discrete sequences, spatial and spatiotemporal data.

For the sake of reproducible science, all code and data necessary to replicate the results shown in this paper are available in the Web page http://tinyurl.com/hbtquqw. All code is written in the free and open source R software environment.

Acknowledgment

This work is financed by the ERDF – European Regional Development Fund through the 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. The work of N. Moniz is supported by a PhD scholarship of FCT (SFRH/BD/90180/2012). The work of P. Branco is supported by a PhD scholarship of FCT (PD/BD/105788/2014). The authors would like to thank Rita Ribeiro for her comments and inputs.

References

[1] Q. Yang and X. Wu, “10 challenging problems in data mining research,” *Int. Jour. of Inf. Tech. & Dec. Mak.*, vol. 5, no. 4, pp. 597–604, 2006.

[2] N. V. Chawla, N. Japkowicz, and A. Kotcz, “Editorial: Special issue on learning from imbalanced data sets,” *SIGKDD Explor. Newsl.*, vol. 6, no. 1, pp. 1–6, Jun. 2004.

[3] T. R. Hoens, Q. Qian, N. V. Chawla, and Z.-H. Zhou, “Building decision trees for the multi-class imbalance problem,” in *Proc. of the 16th PAKDD*. Springer Berlin Heidelberg, 2012, pp. 122–134.

[4] C. Seiffert, T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano, “Rusboost: A hybrid approach to alleviating class imbalance,” *IEEE Trans. SMC*, vol. 40, no. 1, pp. 185–197, 2010.

[5] B. C. Wallace, K. Small, C. E. Brodley, and T. A. Trikalinos, “Class imbalance, redux,” in *Proc. of 11th ICDM*. IEEE, 2011, pp. 754–763.

[6] C. Bunkhumpornpat, K. Sinapiromsaran, and C. Lursinsap, “Dbsmote: Density-based synthetic minority over-sampling technique,” *Applied Intelligence*, vol. 36, no. 3, pp. 664–684, 2012.

[7] K. Li, W. Zhang, Q. Lu, and X. Fang, “An improved smote imbalanced data classification method based on support degree,” in *Proc. of 2014 International Conference IIKI*. IEEE, 2014, pp. 34–38.

[8] L. Torgo, R. P. Ribeiro, B. Pfahringer, and P. Branco, “Smote for regression,” in *Prog. in Art. Int.* Springer, 2013, pp. 378–389.

[9] L. Torgo, P. Branco, R. P. Ribeiro, and B. Pfahringer, “Resampling strategies for regression,” *Exp. Sys.*, vol. 32, no. 3, pp. 465–476, 2015.

[10] M. Kubat and S. Matwin, “Addressing the curse of imbalanced training sets: One-sided selection,” in *Proc. of the 14th ICML*. Nashville, TN, USA: Morgan Kaufmann, 1997, pp. 179–186.

[11] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “Smote: Synthetic minority over-sampling technique,” *JAIR*, vol. 16, pp. 321–357, 2002.

[12] G. Widmer and M. Kubat, “Learning in the presence of concept drift and hidden contexts,” *Mach. Lear.*, vol. 23, no. 1, pp. 69–101, 1996.

[13] C. Chatfield, *The analysis of time series: an introduction*, 6th ed. CRC Press, 2004.

[14] F. Takens, *Dynamical Systems and Turbulence*. Springer Berlin Heidelberg, 1981, ch. Detecting strange attractors in turbulence, pp. 366–381.

[15] R. Ribeiro, “Utility-based regression,” Ph.D. dissertation, Dep. Computer Science, Faculty of Sciences - University of Porto, 2011.

[16] H. Fanaee-T and J. Gama, “Event labeling combining ensemble detectors and background knowledge,” *Prog. in Art. Int.*, pp. 1–15, 2013.

[17] L. Torgo, *Data Mining with R, learning with case studies*. Chapman and Hall/CRC, 2010.

[18] R. Hyndman and Y. Khandakar, “Automatic time series forecasting: The forecast package for r,” *Jour. of Stat. Soft.*, vol. 27, no. 1, pp. 1–22, 2008.

[19] H. Tong, B. Thanoon, and G. Gudmundsson, “Threshold time series modeling of two icelandic riverflow systems1,” *JAWRA*, vol. 21, no. 4, pp. 651–662, 1985.

[20] O. Akbilgic, H. Bozdogan, and M. E. Balaban, “A novel hybrid RBF neural networks model as a forecaster,” *Statistics and Computing*, vol. 24, no. 3, pp. 365–375, 2014.

[21] I. Koprinska, M. Rana, and V. Agelidis, “Yearly and seasonal models for electricity load forecasting,” in *Proc. of 2011 IJCNN*, July 2011, pp. 1474–1481.

[22] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2014.

[23] D. Meyer, E. Dimitriadou, K. Hornik, A. Weingessel, and F. Leisch, *e1071: Misc Functions of the Department of Statistics (e1071), TU Wien*, 2012, r package version 1.6-1.

[24] S. Milborrow, *earth: Multivariate Adaptive Regression Spline Models*, 2013.

[25] A. Liaw and M. Wiener, “Classification and regression by randomforest,” *R News*, vol. 2, no. 3, pp. 18–22, 2002.

[26] L. Torgo and R. Ribeiro, “Utility-based regression,” in *Proc. of 11th PKDD*, Springer, Ed., 2007, pp. 597–604.

[27] L. Torgo, “An infra-structure for performance estimation and experimental comparison of predictive models in r,” *CoRR*, vol. abs/1412.0436, 2014.

[28] G. M. Weiss and H. Hirsh, “Learning to predict rare events in event sequences,” in *Proc. of the 4th KDD*. AAAI Press, 1998, pp. 359–363.

[29] R. Vilalta and S. Ma, “Predicting rare events in temporal domains,” in *Proc. of the 2002 IEEE ICDM*, 2002, pp. 474–481.

[30] J. Chen, H. He, G. J. Williams, and H. Jin, “Temporal sequence associations for rare events,” in *Proc. of the 8th PAKDD*. Springer, 2004, pp. 235–239.

[31] T. Fawcett and F. Provost, “Activity monitoring: Noticing interesting changes in behavior,” in *Proc. of the 5th ACM SIGKDD*, 1999, pp. 53–62.

[32] J. Lin, E. J. Keogh, A. W.-C. Fu, and H. V. Herle, “Approximations to magic: Finding unusual medical time series.” in *CBMS*. IEEE Computer Society, 2005, pp. 329–334.

[33] S. Basu and M. Meckesheimer, “Automatic outlier detection for time series: An application to sensor data,” *Knowl. Inf. Syst.*, vol. 11, no. 2, pp. 137–154, Feb. 2007.

[34] E. Keogh, S. Lonardi, and B. Y.-c. Chiu, “Finding surprising patterns in a time series database in linear time and space,” in *Proc. of the 8th ACM SIGKDD*, New York, NY, USA, 2002, pp. 550–556.

[35] S. L. Scott, “Detecting network intrusion using a markov modulated nonhomogeneous poisson process,” *Subm. to Jour. ASA*, 2000.

[36] A. Ihler, J. Hutchins, and P. Smyth, “Adaptive event detection with time-varying poisson processes,” in *Proc. of the 12th ACM SIGKDD*, New York, NY, USA, 2006, pp. 207–216.