Biased Resampling Strategies for Imbalanced Spatio-Temporal Forecasting

Mariana Oliveira, Nuno Moniz, Luís Torgo, Vítor Santos Costa

All code and data necessary to replicate results publicly available at
https://github.com/mrfoliveira/STResampling-DSAA2019.

Biased Resampling Strategies for Imbalanced Spatio-Temporal Forecasting

Mariana Oliveira[†*], Nuno Moniz[†*], Luís Torgo[†‡*], and Vítor Santos Costa[†*]
[*] INESC TEC, Porto, Portugal
[†] Department of Computer Science, Faculty of Sciences, University of Porto, Porto, Portugal
[‡] Dalhousie University, Nova Scotia, Canada
Email: {mariana.r.oliveira, nmmoniz}@inesctec.pt

Abstract—**Extreme and rare events, such as abnormal spikes in air pollution or weather conditions can have serious repercussions. Many of these sorts of events develop from spatio-temporal processes, and accurate predictions are a most valuable tool in addressing their impact, in a timely manner. In this paper, we propose a new set of resampling strategies for imbalanced spatio-temporal forecasting tasks, by introducing bias into formerly random processes. This spatio-temporal bias includes a hyper-parameter that regulates the relative importance of the temporal and spatial dimensions in the selection of observations during under- or over-sampling. We test and compare our proposals against standard versions of the strategies on 10 different geo-referenced numeric time series, using 3 distinct off-the-shelf learning algorithms. Experimental results show that our proposal provides an advantage over random resampling strategies in imbalanced spatio-temporal forecasting tasks. Additionally, we also find that valuing an observation's recency is more useful when over-sampling; while valuing its spatial distance to other cases with extreme values is more beneficial when under-sampling.**

I. Introduction

Abnormal weather conditions, pollution level spikes, and fire ignitions are examples of rare or extreme events. These are commonly associated with impactful situations, often developed as the effect of uncommon natural factors on spatio-temporal processes or through the impact of conditional changes in underlying factors, i.e. concept drift [1]. Due to characteristics of such events, they are difficult to predict, but an accurate anticipation is very important [2].

Our work focuses on this problem of forecasting extreme values in spatio-temporal settings. Standard learning methods often assume data to be independent and identically distributed (i.i.d.). However, in the context of spatio-temporal forecasting, this assumption is likely false due to the correlation that data points have with their neighbours, both spatially and temporally, possibly leading to poor predictive performance [3]. In fact, spatio-temporal modelling approaches are highly prone to problems when focusing on the accurate prediction of extreme values due to their assumption of uniform domain preferences by users (each point is equally important) and the use of standard evaluation metrics (both internally and in model optimization) which is known to specialize the forecasters towards the central tendency of the distribution (i.e. average target values) [4], instead of the extreme values which we are focused on accurately forecasting.

Our main motivation derives from the exciting results obtained by the application of resampling strategies in the prediction of rare events [4]. Resampling strategies are off-the-shelf methods that are applied in order to pre-process the training data, biasing it towards users' objectives, either by the removal or replication/generation of cases. Although the application of resampling strategies in the context of classification tasks has a long research record, having been applied to many domains (e.g. financial data analysis, intrusion detection in network forensics, oil spill detection and prognosis of machine failures), only recently has the problem of imbalanced domain learning been extended to numerical prediction tasks.

In this paper, we leverage recent work on imbalanced domain learning [4], [5], addressing the problem of forecasting extreme values in spatio-temporal settings. With this objective, we propose a novel set of resampling strategies that are tailored for the spatio-temporal context via 1) accounting for the spatial and temporal relevance of the data points and 2) by assuming that the spatial and temporal dimensions may have a different impact in the modelling process, depending on the domain. It should be noted that, as far as we know, this is the first proposal focused on solving imbalanced numerical forecasting in spatio-temporal contexts. The contributions of this paper are the following:

1) biased under-sampling and over-sampling strategies for spatio-temporal forecasting of extreme values;
2) an experimental evaluation including 10 data sets, a paired comparison of purely random and spatio-temporal biased resampling strategies for various learning algorithms, and sensitivity analysis of the strategies' main parameters.

The remainder of this paper is organized as follows. The following section provides the problem definition (Section II). Resampling strategies for spatio-temporal context proposed in this paper are described and formalized in Section III. An experimental evaluation is presented in Section IV, followed by results (Section V), and their discussion (Section VI). Finally, a review of previous work is detailed in Section VII and conclusions are presented (Section VIII).

II. Problem Definition

In spatio-temporal forecasting the aim is to predict the future values of a target variable at a given location. Consider a set of locations $L = \{l_1, \cdots, l_n\}$, a set of time-stamps $T = \{t_1, \cdots, t_m\}$, and a set of observations

$$\begin{aligned} \mathcal{D} \quad = \quad & \{\{y_{1,1}, <x^1_{1,1}, \cdots, x^k_{1,1}>\}, \cdots, \\ & \{y_{i,j}, <x^1_{i,j}, \cdots, x^k_{i,j}>\}\}^{j \in \{1,2,\cdots,n\}}_{i \in \{1,2,\cdots,m\}}, \end{aligned}$$

where $y_{i,j}$ and $x^k_{i,j}$ correspond, respectively, to the values of the target variable Y and predictors X^k, at time t_i and geographical location l_j. The goal is to predict the value of Y at a location of interest, l_s ($s \in \{1, 2, \cdots, n\}$), at a future time, t_f, given the observed values $y_{i,j}$ and $\mathbf{x_{i,j}}$, such that $t_i < t_f$.

In this work we focus on imbalanced spatio-temporal forecasting tasks, where certain ranges of values in the target variable Y are most important to the end-user, but severely under-represented in the training data[1], e.g. predicting extreme levels of pollution. This representation bias (imbalance) in the data commonly leads to modelling solutions that are optimized towards the prediction of values within the central tendencies of the distributions, and to the detriment of accurate predictions of extreme values. Given that events with extreme values often represent situations of high importance and interest to the end-users, our objective is improving the predictive ability of such cases.

In order to formalize our prediction task, we need to specify what is meant by "highly important" values of the target variable. To this end, we resort to the work of Ribeiro [6], proposing the use of a relevance function to map the domain of continuous variables into a $[0, 1]$ scale of relevance, i.e. $\phi(Y) : \mathcal{Y} \to [0, 1]$. Usually, 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, we do not assume expert knowledge concerning the domains. Instead, we employ an automated approach to define the relevance function. We use box plot statistics as detailed by Ribeiro [6], which automatically assigns more relevance/importance to the extreme low and high values of the target variable. This automatic approach uses a piecewise cubic Hermite interpolation polynomials [7] (*pchip*) algorithm to interpolate a set of points describing the distribution of the target variable. These points are given by box plot statistics. The outlier values according to box plot statistics (either extreme high or low) are given a maximum relevance of 1 and the median value of the distribution is given a relevance of 0. The relevance of the remaining values is then interpolated using the *pchip* algorithm.

Finally, in order to define the cases considered by the user as having extreme values, we are also required to establish a threshold value t_R. This leads to the formalization of two subsets of the data set, containing the cases with normal and extreme values, respectively: $\mathcal{D}_N = \{\langle \mathbf{x}, y \rangle \in \mathcal{D} : \phi(y) < t_R\}$ and $\mathcal{D}_R = \{\langle \mathbf{x}, y \rangle \in \mathcal{D} : \phi(y) \geq t_R\}$, where $|\mathcal{D}_R| \ll |\mathcal{D}_N|$. We should note that throughout this paper we assume a value of 0.9 as t_R. Furthermore, we should stress that this threshold does not serve the purpose of discretization, given that predictions are evaluated taking into account their numerical error and not merely their presence in either $\mathcal{D}_N$ or $\mathcal{D}_R$.

III. Spatio-Temporal Bias Resampling Strategies

Resampling strategies such as under-sampling and over-sampling which reduce/increase the number of normal/rare cases are well-known. In most contexts, observations are selected entirely randomly during this process, though there are proposals introducing bias, e.g., in a time series forecasting context [8]. In this section we detail our proposal of novel resampling strategies that include a spatio-temporal bias in case selection procedures.

A. Algorithms

We aim at taking advantage of the effect that spatial and temporal dimensions may have in determining which observations are more useful to keep or replicate during the resampling process. For this purpose, we make use of a sampling weight, $W_{i,j}$, that regulates the probability that a certain observation will be selected to stay/be replicated in the training set. The higher the weight, the higher the probability that an observation is found in the data set after resampling. This weight is based on the intuitions that the temporal recency of the observation, as well as the relevance of its spatial neighbours, may have an impact on the observation's importance to the learning process. It also takes into account that, depending on the domain, the temporal or spatial dimensions may have a different impact, so we introduce a parameter α to strike a balance between a spatial and a temporal component, $W^L_{i,j}$ and $W^T_{i,j}$, as in Eq. 1,

$$W_{i,j} = \alpha \times W^T_{i,j} + (1 - \alpha) \times W^L_{i,j} + \epsilon \qquad (1)$$

where i is a time-stamp index, j is a location index, and ϵ is a small value added so that no observation has zero probability of being kept/added to the training set during resampling.

The weights are calculated according to the algorithm depicted in Figure 1, and the intuitions behind them are as follows:

- Observations that are more recent should have higher probability of being selected given that they hold information that is more up-to-date and, thus, more relevant to the predictive task at hand (cf. line 12 in Figure 1);
- When selecting extreme observations to be over-sampled, observations that are spatially farther away from other extreme cases should take priority as they are otherwise isolated and could be more prone to be filtered out or ignored during the learning process (cf. lines 17–20);
- When selecting normal cases to keep in under-sampling, observations that are spatially farther away to extreme cases should have higher probability of being selected to stay in the final set so borders around extreme cases can be easier to learn (cf. lines 15–16, 19–20).

[1] Note that in this paper we assume that all events follow the same distribution, thus using a global definition of what constitutes a rare/extreme case, instead of a local one.

```
function SampleWeights($T, L, Y, \phi, t_R, \alpha, \epsilon$)
    ▷ $T = \{t_i\},\ i \in \{1 \cdots m\}$ - Time-stamps
    ▷ $L = \{l_j\},\ j \in \{1 \cdots n\}$ - Geolocations
    ▷ $Y = \{y_{i,j}\},\ i \in \{1 \cdots m\},\ j \in \{1 \cdots n\}$ - Target values
    ▷ $\phi(Y)$ - User specified relevance function
    ▷ $t_R$ - The relevance threshold for $y$ values
    ▷ $\alpha$ - Dimensional weighting factor
    ▷ $\epsilon$ - Minimum weight

    $d_{max} \leftarrow \max(\text{DIST}(l_j, l_k)),\ j \neq k$   ▷ DIST is a spatial distance function
    for $i \leftarrow 1, m$ do
        $W^T_{i,j} \leftarrow t_i$   ▷ Temporal weight is proportional to time-stamp (higher weight → more recent observation)
        $R_i \leftarrow \{(i,j) \mid \phi(y_{i,j}) \geq t_R\}$
        for $j \leftarrow 1, n$ do
            if $R_i = \emptyset$ then
                $W^L_{i,j} \leftarrow d_{max}$   ▷ No rare cases at time-stamp $t_i$ means spatial weights of normal cases are maximal
            else if $|R_i| = 1 \wedge (i,j) \in R_i$ then
                $W^L_{i,j} \leftarrow d_{max}$   ▷ Only one rare case at time-stamp $t_i$ means its spatial weight is maximal (isolated rare case)
            else
                $W^L_{i,j} \leftarrow \min(\text{DIST}(l_j, l_k)), (i,k) \in R_i, j \neq k$   ▷ Spatial weight is proportional to minimum distance to rare case at time-stamp $t_i$ (higher weight → more isolated rare case OR normal case farther away from rare cases)
            end if
        end for
    end for
    $W^T \leftarrow \text{NORM}(W^T, \phi)$   ▷ NORM normalizes weights separately for cases with normal and extreme values
    $W^L \leftarrow \text{NORM}(W^L, \phi)$
    $W \leftarrow \alpha \times W^T + (1-\alpha) \times W^L + \epsilon$
    return $W$
end function
```

Fig. 1. Spatio-temporal bias resampling weights

Since the under- and over-sampling algorithms focus, respectively, on randomly selecting normal/extreme cases to keep/add to the training set, the weights are normalized separately for each type of observation.

As an example, heatmaps of the relevance function, and the spatial, temporal and spatio-temporal weights for data set 31 (see Section IV-A1) are pictured in Figure 2. The relevance function, ϕ, is calculated automatically, and values are considered extreme when $\phi(y) \geq 0.9$. For each time-slice along the x-axis, the spatial weight of a case (a cell in the graph, W^L) is higher when the spatial distance to measured extreme values is larger. Note that while W^L is related to ϕ, this graph cannot show the spatial relationships between different locations along the y-axis. The temporal weights, W^T, increase smoothly along the x-axis, as observations become more recent. The spatio-temporal weight, W, is then calculated by combining the temporal and spatial dimensions in equal measure ($\alpha = 0.5$ in this example).

Next, we specify the proposed variants of under- and over-sampling algorithms which include the spatio-temporal weight to bias the random sampling process. The pseudocode for both variants can be found in Figure 3.

1) Biased Under-Sampling: The process of spatio-temporal biased random under-sampling first collects all observations with extreme values. Then, it samples a number of instances with normal values to be kept in the new data set, with a probability that is proportional to the weights calculated above. A case will have a higher probability of being selected if *i)* it is more recent, and *ii)* it is spatially farther away from any extreme cases at the time of observation. This approach uses random selection without replacement, and the parameter u, $0 < u < 1$, can be used to define the fraction of normal cases, w.r.t. to the number of normal cases in the data set, that is to be kept in the new data set.

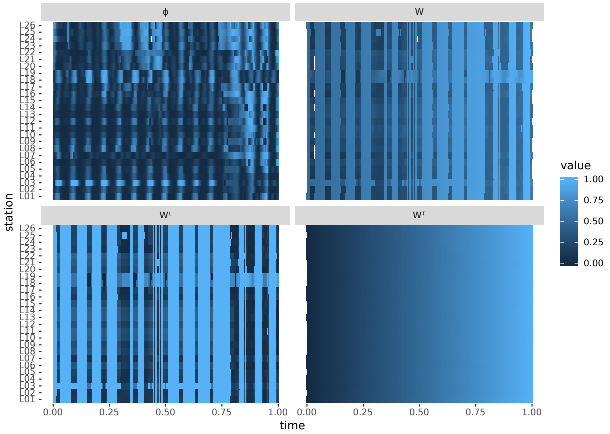


Fig. 2. Heatmaps for data set 31, showing relevance values, ϕ, spatial weight, W^L, temporal weight, W^T and their combination as the spatio-temporal weight, W. Each cell in the heatmap corresponds to one observation at a given point in time and location.

2) Biased Over-Sampling: The process of spatio-temporal biased random over-sampling starts with the initial data set. Then, it randomly selects a number of instances with extreme values to be replicated and added to the data set, with a probability that is proportional to the weights calculated above. The probability of being replicated is higher if the case is *i)* more recent, and *ii)* more distant to other extreme values measured at the time of observation. This random selection is done with replacement and the parameter $o > 0$ allows the user to select a specific percentage of highly relevant cases to replicate.

```
function STRandReSampling($D, Y, \phi(Y), t_R, B, u, o$)
    ▷ $D$ - A data set
    ▷ $Y$ - The target variable
    ▷ $\phi(Y)$ - User specified relevance function
    ▷ $t_R$ - The threshold for relevance on $y$ values
    ▷ $W$ - Spatio-temporal bias weight
    ▷ $B$ - Variant of biased resampling (STRUS for biased under-sampling; STROS for biased over-sampling)
    ▷ $u$ - Percentage of under-sampling (if $B = STRUS$)
    ▷ $o$ - Percentage of over-sampling (if $B = STROS$)

    $D_R \leftarrow \{D_i : \forall y_i \in Y, \phi(y_i) > t_R\}$   ▷ Cases considered as highly relevant
    $D_N \leftarrow \{D_i : \forall y_i \in Y, \phi(y_i) \leq t_R\}$   ▷ Cases considered as normal
    if $B = STRUS$ then   ▷ Biased random under-sampling
        $TgtNr \leftarrow |D_N| \times u$
        $newData \leftarrow D_R$   ▷ Highly relevant cases are kept in the new data set
        $selCases \leftarrow \text{SAMPLE}(tgtNr, D_N, W)$   ▷ Biased random selection of a number of normal cases from $D_N$
    else if $B = STROS$ then   ▷ Biased random over-sampling
        $tgtNr \leftarrow |D_R| \times o$
        $newData \leftarrow D$   ▷ All cases are kept in the new data set
        $selCases \leftarrow \text{SAMPLE}(tgtNr, D_R, W)$   ▷ Biased random selection of a number of rare cases from $D_R$
    end if
    $newData \leftarrow c(newData, selCases)$   ▷ Add selected cases to the new data set
    return $newData$
end function
```

Fig. 3. Spatio-temporal bias random under- and over-sampling

IV. Experimental Evaluation

In this section, we describe the experimental evaluation process used to test and compare the following resampling strategies: random under- and over-sampling (*RUS* and *ROS*), and spatio-temporal bias random under- and over-sampling (*STROS* and *STRUS*). Note that *ROS* and *RUS* are versions of *STROS* and *STRUS* where the probability of selection is not dependent on spatio-temporal weights, i.e., their sampling processes work as if all observations have the same non-zero weight. These strategies were also compared against a baseline where no resampling was applied.

A. Data and Methods

Next, we describe the data sets, pre-processing methods, and learning algorithms used.

1) Data: Ten variables from five different environmental monitoring data sources were used as target variables, as if each was an independent and univariate data set. A summary of the characteristics of each data set can be found in Table I. The size of sensor networks varies from 20 to 72 geolocations irregularly distributed in space, with measurements being taken from below 200 times to more than 11k times at different frequencies. Percentages of cases with extreme values range from 2.4% to 8.6% of instances. These percentages were calculated by inferring the relevance function of each data set, ϕ, using a relevance threshold t_R of 0.9.

2) Methods: In this section, we explain how features were generated, how we handle missing data, and the used regression algorithms.

a) Feature engineering: In order to use standard machine learning algorithms and compare the effect of resampling approaches, the pre-processing strategy proposed in [12] was used to transform the data sets. For each observation, the predictors used to predict the target value are *i)* a temporal embed of values previously measured at that location, *ii)* a set of spatio-temporal indicators built by calculating summary statistics of previous measurements at neighbouring locations within three dataset-specific boundaries of spatio-temporal distance, and *iii)* ratios between the indicators of spatio-temporal neighbourhoods of increasing radius. This data transformation was performed on the whole data set before any train/test data divisions, and resulted in a total of 20 predictors.

b) Handling missing data: Three of the data sources were measured at every point in time and space, with no missing values. However, for others, only a percentage of location and time-stamp pairs (from 39% to 49%) have available values, due to failures in data acquisition, or sensor stations being set up at later times. Before applying a resampling strategy and/or training a model within the evaluation framework, all columns that have 20% or more of training data missing are discarded as they should not be very useful predictors. The remaining missing data is dealt with as follows: first, any rows that have too many predictors missing (set at 20% of columns) are discarded from the training set; then, missing values for both the training and test sets are imputed as the median of that column in the set.

c) Regression algorithms: Three standard regression algorithms were selected to test whether the results are consistent across different tools. Implementations available in free and open source **R** packages were used: multivariate adaptive regression splines algorithm (*MARS*) from package **earth** [13], random forest (*RF*) from **ranger** [14], and regression trees (*RPART*) from **rpart** [15]. Results were obtained using default parametrizations, with the exception of number of trees in *RF* which was set to 250.

B. Evaluation Methodology

The evaluation methodology involves two choices: which performance metrics are more appropriate, and what procedure should be used to estimate them. In this section, we describe our evaluation framework.

1) Performance Metrics: The focus of this experimental evaluation is to assess the predictive ability of models in forecasting highly relevant cases, corresponding to extreme target values. As previously mentioned, given the considerable representation bias between cases with values around the central tendency of the distribution and those with extreme values, the use of standard (average-based) numerical evaluation metrics is not appropriate: these will lead to an optimization process focusing on reducing the average error of the models. In order to provide a thorough analysis of the models' ability in predicting extreme values, this experimental evaluation is focused on the use of the utility-based F-Score [6] metric.

The utility-based F-Score is motivated by the well-known precision/recall evaluation framework [16] used in classification tasks. Based on the concepts of relevance (see Section II) and utility, Ribeiro [6] presents a formulation of precision and recall for regression tasks with imbalanced domains, of which the following is an alternate definition for simplification purposes:

$$prec_{\phi}^{u} = \frac{\sum\limits_{\phi(\hat{y}_i)\geq t_R, \phi(y_i)\geq t_R} (1 + u(\hat{y}_i, y_i))}{\sum\limits_{\phi(\hat{y}_i)\geq t_R} (1 + \phi(\hat{y}_i))} \tag{2}$$

$$rec_{\phi}^{u} = \frac{\sum\limits_{\phi(\hat{y}_i)\geq t_R, \phi(y)\geq t_R} (1 + u(\hat{y}_i, y_i))}{\sum\limits_{\phi(y_i)\geq t_R} (1 + \phi(y_i))} \tag{3}$$

where $\phi(y_i)$ and $\phi(\hat{y}_i)$ is the relevance associated with the true value y_i and predicted value $\hat{y}_i$, respectively; t_R is a user-defined relevance threshold, above which cases are signalled as highly relevant for the user, and $u(\hat{y}_i, y_i)$ is the utility of making the prediction $\hat{y}_i$ for the true value y_i, normalized to $[-1, 1]$. In this paper t_R is set to 0.9, and relevance functions are automatically calculated.

Utility is commonly referred to as being a function combining positive benefits and negative benefits (costs). In this paper, we use the approach for utility surfaces proposed by Ribeiro [6]. Unlike in classification tasks, utility is interpreted as a continuous version of the benefit matrix proposed by Elkan [17], where utility U is defined as the difference between

TABLE I
REAL-WORLD DATA SETS

Data set	ID	Variables	Frequency	#timeIDs	#locIDs	#inst.	%avail.	%extr.	Source
MESA Air Pollution	10	NO$_X$ conc.	bi-weekly	280	20	5.6k	100	7.3	[9] [1]
NCDC Air Climate	20	precipitation	monthly	105	72	7.6k	100	6.0	[9] [1]
TCE Air Climate	30	ozone	hourly	360	26	8.6k	100	6.3	[9] [1]
	31	temperature				9.4k		3.8	
	32	wind speed				9.4k		2.4	
RURAL airBase	40	PM10 conc.	daily	4382	70	149k	49	7.5	[10] [2]
Beijing UrbanAir	50	NO$_X$ conc.	hourly	11235	36	404k	39	3.5	[11] [3]
	51	PM10 conc.		11312		407k	39	5.5	
	52	wind speed		11319		407k	41	8.6	
	53	PM25 conc.		11350		409k	41	3.8	

[a] Downloaded at: http://www.di.uniba.it/\~appice/software/COSTK/index.htm

[b] Loaded from *R* packages *GSIF* (0.5-4) and *spacetime* (1.2-1).

[c] Downloaded at: https://www.microsoft.com/en-us/research/publication/u-air-when-urban-air-quality-inference-meets-big-data/

benefits B and costs C, $U = B - C$. To calculate utility, we have to consider two factors: *i)* if the true and predicted values and their respective relevance belong to similar relevance bins (e.g. both values are extreme and thus highly relevant); and *ii)* whether the prediction is reasonably accurate, given a factor of maximum admissible loss, defined by the author. Accurate predictions are attributed non-negative utility, with higher utility values being attributed to correct predictions of the (highly relevant) extremes of the target variable. When predictions are not entirely accurate, utility also takes into account the magnitude of predictive errors: predictions that are reasonably close to the true values have non-negative utility; but, as the distance between predicted and true values increases, utility becomes negative, tending to -1.

Finally, the utility-based F-Score metric F^u_β combines both precision ($prec^u_\phi$) and recall (rec^u_ϕ) with an harmonic mean, including a β factor denoting the importance attributed to the components. In this paper, it is set as 1, equally weighting precision and recall. We should stress that, unlike the traditional F-Score metric used in classification tasks, this formulation of the utility-based F-Score is based on the analysis of numerical prediction errors.

2) Estimation Procedures: Estimating performance metrics using cross-validation in settings where temporal and spatial dependence structures are present raises issues [18]–[20]. In this context, we opt for a *prequential temporal block evaluation* procedure (as described in the work of Oliveira et al. [20]), where the data set is divided into 10 blocks respecting temporal order, using a growing window for training. That is, a model is trained using the first block and tested in the second; then, it is trained using the first two blocks, and tested in the third; until all (except the first) blocks are used for testing. The relevance function is automatically calculated based on the training target values at each step. Evaluation metrics are averaged over the 9 testing blocks.

For each resampling strategy, there are parameters that need to be set: the over- and under-sampling percentages, o and u, and α for biased resampling approaches. These parameters can be set *a priori*, or they can be tuned internally. The set of parameters tested include all combinations of $u \in \{0.2, 0.4, 0.6, 0.8, 0.95\}$, $o \in \{0.5, 1, 2, 3, 4\}$, and $\alpha \in \{0, 0.25, 0.5, 0.75, 1\}$. Once again, note that the percentage of under-sampling determines that the set of non-extreme observations will be reduced to u of its original size. The percentage of over-sampling establishes that o of the original number of extreme observations will be added to the training set, i.e., a o of 1 means the frequency of rare cases will be doubled after resampling.

When internally tuning them, we deploy time-blocked cross-validation (also described in [20]) within each growing window of blocks in the prequential evaluation procedure described above. The parameters obtaining the best results in that training window are then used to re-sample that training set. While time-blocked cross-validation does not completely respect temporal order, it does acknowledge temporal dependence and it uses the whole training set to search for parameters. This is why it was employed instead of a prequential evaluation in this internal setting, where the amount of working data is smaller.

The proposed methods are implemented in an **R** package, *STResampling*, available at https://github.com/mrfoliveira/STResampling-DSAA2019. All the code and data necessary to replicate our results is also included.

V. RESULTS

In this section, we present the results obtained using the framework detailed in the previous section and three different parametrization methodologies.

Specifically, we show the results of following two strategies that are often required in real-world applications: *a)* tuning the parameters internally, and *b) a priori* fixing the parameters at arbitrary values. The first strategy requires more computational resources, while the second gives us a glimpse of what may happen if "default" parameters are used regardless of data set characteristics. We also show optimal results obtained by *c)* using the best parameters for each data set, chosen *a posteriori*. While reporting on optimal results does not guarantee that it would be possible to replicate them, it can help establish the full potential of the resampling strategies.

Next, we present a summary of the overall results, which are further detailed below. We dedicate the remainder of the section to parameter sensitivity analysis and the precision-recall trade-off of our resampling strategies.

A. *Summary of Results*

Table II summarizes our results. The rank according to F_1^u is calculated per learning model and data set pair, and then the overall average is calculated. The best results are in bold. The results show that whether the parametrization is internally tuned according to the results of each training window, or it is chosen *a posteriori* to be the one getting optimal results for each data set, or it is fixed arbitrarily to be the same for all data sets, biased under-sampling always achieves the best overall average rank. Moreover, if considering optimal or internally tuned parametrization, the second best result is obtained by the biased form of over-sampling. However, when fixing parameters arbitrarily, both types of under-sampling are better than over-sampling. Note that applying any form of resampling always improves against the baseline and that whether under- or over-sampling is used, the results are always improved, on average, by including a spatio-temporal bias to the approach.

TABLE II
AVERAGE RANKS OF F_1^u RESULTS

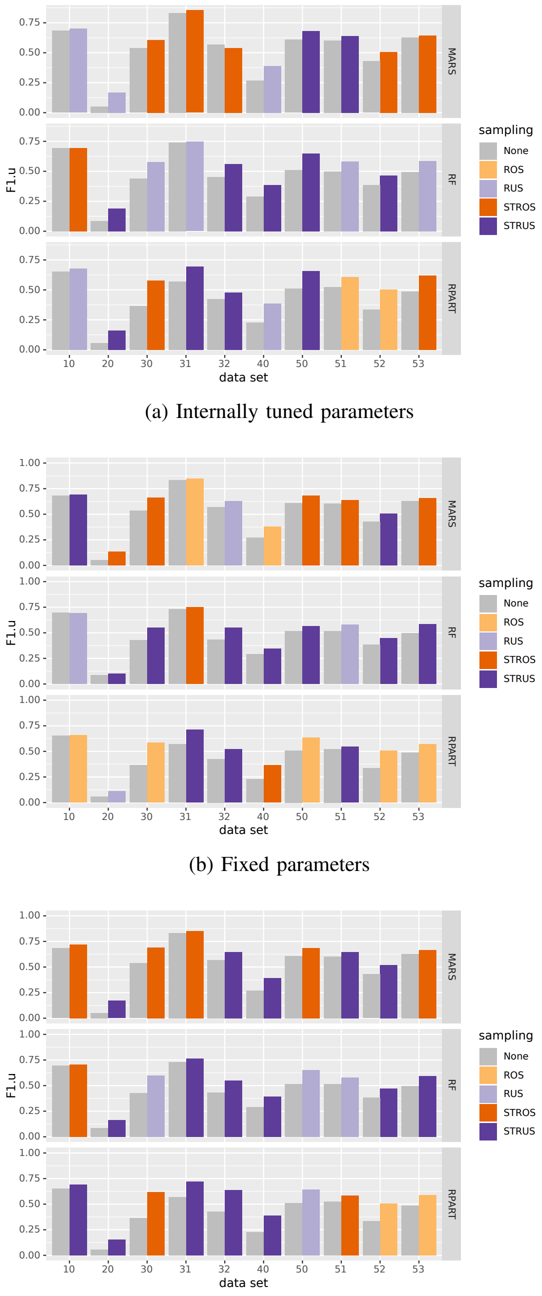
parametrization	None	ROS	STROS	RUS	STRUS
internally tuned	4.60	3.07	2.37	2.67	**2.30**
fixed arbitrarily *a priori*	4.53	2.77	2.73	2.57	**2.40**
optimal *a posteriori*	5.00	3.07	2.27	2.93	**1.73**

B. *Internally Tuning Parameters*

This subsection describes the results of prequential time-block evaluation with internal parameter tuning using time-block cross-validation, as detailed in Section IV-B.

Figure 4a shows the best resampling approach w.r.t. the baseline. It is clear that some sort of resampling benefits all data sets and learning model pairs, except data set 32 when using MARS. When using any of the three learning models, the best results on the majority of data sets are obtained when using a form of biased resampling. When using RF, there is a very clear preference for under-sampling, while for the other two models the best performers are more balanced between under- and over-sampling alternatives.

For each learning model and data set pair, we rank the different strategies according to F_1^u. Table III shows the average ranks of each sampling method aggregated by model. We can see that our biased under-sampling proposal is the best performer in the case of RF and RPART (tied with non-biased over-sampling in the case of RPART), while biased over-sampling is best when using MARS. Table IV shows the breakdown by data set. Notice that the baseline is consistently ranked bottom for 8 of the data sets. The best results are always obtained by using resampling. Biased resampling strategies achieve the best rank in 6 of the 10 data sets; random resampling works best for 3; and in the case of data set 31, there is a tie.



(a) Internally tuned parameters

(b) Fixed parameters

(c) Optimal parameters

Fig. 4. Baseline and best F_1^u result achieved for each data set. Top layers present MARS results, followed by RF and RPART. Two bars correspond to one data set: the baseline, in gray, and the best result. Under-sampling is shown in shades of purple; over-sampling, in orange shades; darker colours indicate bias

C. *Fixing Parameters A Priori*

Next, we show the results obtained by the prequential evaluation procedure when setting parameters to the values in

TABLE III
AVERAGE RANKS OF F_1^u PER MODEL USING TUNED PARAMETERS

model	None	ROS	STROS	RUS	STRUS
MARS	4.40	3.20	**2.20**	2.50	2.70
RF	4.70	3.60	2.90	2.00	**1.80**
RPART	4.70	**2.40**	2.90	2.60	**2.40**

TABLE IV
AVERAGE RANKS OF F_1^u PER DATA SET USING TUNED PARAMETERS

data	None	ROS	STROS	RUS	STRUS
10	2.67	2.67	3.33	**2.33**	4.00
20	5.00	3.67	3.33	1.67	**1.33**
30	5.00	3.00	**2.00**	2.33	2.67
31	5.00	**2.33**	**2.33**	**2.33**	3.00
32	3.33	4.00	**2.00**	3.67	**2.00**
40	5.00	3.33	3.33	**1.33**	2.00
50	5.00	3.33	3.00	2.67	**1.00**
51	5.00	2.67	3.00	**2.00**	2.33
52	5.00	3.00	2.67	2.67	**1.67**
53	5.00	2.67	**1.67**	2.67	3.00

the middle of the grid search, regardless of data set or learning algorithm. That is, α is set to 0.5, o to 2, and u to 0.6.

In Figure 4b, we can see the results obtained by the best resampling approach against the baseline. A strong preference for under-sampling when using RF is still noticeable, while MARS and RPART favour over-sampling in more cases (cf. 4a). Using these fixed parameters, biased resampling approaches have the advantage in almost all cases when using MARS and RF. However, in the case of RPART (the least complex of learning models), biased resampling only achieves the best result for 4 out of 10 data sets.

For another perspective on the results, Tables V and VI show the average ranks of F_1^u, of each sampling method using the parametrization outlined above, aggregated by learning model and data set, respectively.

In Table V, we see that when using MARS and RF, the best performance is achieved by using resampling methods with spatio-temporal bias when over- or under-sampling, respectively. Though in the case of RPART the best performance is obtained by random over-sampling, there is still an advantage to applying spatio-temporal bias when under-sampling. When using RF, it is advantageous to apply spatio-temporal bias regardless of the resampling method being used.

TABLE V
AVERAGE RANKS OF F_1^u PER MODEL WITH FIXED PARAMETERS

model	BASELINE	ROS	STROS	RUS	STRUS
MARS	4.60	2.70	**2.00**	2.80	2.90
RF	4.60	3.60	3.10	2.00	**1.70**
RPART	4.40	**2.00**	3.10	2.90	2.60

In Table VI, we can see that, even when using parameters that were not specifically selected for each data set, applying a spatio-temporal bias to a form of random resampling still achieves a result that is best (or tied for best) for half the data sets. It is also noticeable in Table VI that, as the size of the data set increases (the table is ordered accordingly), the advantage of biased approaches becomes more clear, indicating that, with these "default" parameters, the bias works better as it is applied to larger data sets that include missing data.

TABLE VI
AVERAGE RANKS OF F_1^u PER DATA SET WITH FIXED PARAMETERS

data	None	ROS	STROS	RUS	STRUS
10	2.33	3.33	4.33	**2.00**	3.00
20	5.00	3.00	2.67	**1.67**	2.67
30	4.67	**2.00**	2.33	3.67	2.33
31	5.00	**1.67**	2.67	2.67	3.00
32	3.67	4.33	3.33	**1.67**	2.00
40	5.00	2.67	**2.33**	**2.33**	2.67
50	5.00	2.33	**2.00**	3.00	2.67
51	4.67	3.00	3.00	2.67	**1.67**
52	5.00	3.00	2.67	2.67	**1.67**
53	5.00	2.33	**2.00**	3.33	2.33

D. *Optimal Parametrization*

In this section, we present the optimal results achieved by each resampling strategy. These results were obtained by running prequential time-block evaluation with all combinations of parameters and, *a posteriori*, selecting the best performers for each data set and learning algorithm pair, in order to assert the potential of each strategy.

In Figure 4c, a baseline and the optimal F_{1_u} achieved overall are presented for each data set and learning model. When using MARS or RPART, a biased approach is always (or almost always) preferred, with a near-perfect balance between under- and over-sampling. When using RF, however, a strong preference for under-sampling is noticeable and non-biased under-sampling is responsible for the best results for 3 out of 10 data sets.

Table VII shows the average rank of each resampling approach aggregated by model. The baseline, where no step to address the imbalance problem was taken, is consistently outperformed by all resampling approaches. Including a spatio-temporal bias improves the rank of both over- and under-sampling in all cases. Biased under-sampling achieves the best results when using RPART or RF, and biased over-sampling when using MARS.

TABLE VII
AVERAGE RANKS OF OPTIMAL F_1^u RESULTS PER MODEL

model	None	ROS	STROS	RUS	STRUS
MARS	5.00	2.80	**1.70**	3.70	1.80
RF	5.00	3.70	2.90	2.00	**1.40**
RPART	5.00	2.70	2.20	3.10	**2.00**

In Table VIII, a complementary view is given, aggregating the ranks by data set instead of by model. It is shown that for all data sets, it is beneficial to apply some kind of random resampling, with the advantage being greater, in general, if a spatio-temporal bias is observed.

1) Statistical Significance: The statistical significance of our findings was tested with the Friedman test, as suggested in [21]. The resulting critical difference diagrams can be

TABLE VIII
AVERAGE RANKS OF OPTIMAL F_1^u RESULTS PER DATA SET

data	None	ROS	STROS	RUS	STRUS
10	5.00	3.00	**1.33**	3.67	2.00
20	5.00	4.00	2.67	2.33	**1.00**
30	5.00	2.67	**1.67**	3.00	2.67
31	5.00	2.33	2.00	4.00	**1.67**
32	5.00	3.67	2.67	2.67	**1.00**
40	5.00	3.00	3.00	3.00	**1.00**
50	5.00	3.00	2.67	**2.00**	2.33
51	5.00	3.67	2.00	2.67	**1.67**
52	5.00	2.67	2.33	3.33	**1.67**
53	5.00	2.67	**2.33**	2.67	**2.33**

seen in Figure 5. While differences between biased and non-biased variants of the resampling approaches are not generally significant, both spatio-temporally biased resampling strategies stand out by achieving significantly better results than the baseline regardless of the learning model being used.

E. Parameter Sensitivity Analysis

To study parameter sensitivity, we investigate how each resampling percentage and α pair ranks, on average, against all other pairs (including the baseline where no resampling is performed), for all data sets. Results are presented in Figure 6. In general, a colour gradient is more noticeable along the X-axis, indicating that resampling percentage has a larger impact than α (or its absence, which corresponds to resampling without bias). However, a diagonal gradient is discernible with the worst performers being accumulated in the two corners where values for both parameters are either very high (when under-sampling) or very low (when over-sampling).

1) Parameter α: In Figure 7, results are aggregated to better evidence the impact of weighting factor α. It is apparent that, when under-sampling, low values of α are preferred. This indicates that it is more useful to keep normal cases that are distant spatial neighbours to extreme cases, weighing temporal recency less (or not at all). When over-sampling, it is more advantageous to favour more recent extreme cases for replication, weighting spatial isolation from other rare cases to a lesser degree.

F. Precision and Recall Trade-Off

While F_1^u is a useful metric for evaluation/optimization processes when an algorithm capable of accurately predicting extreme values is required, the trade-offs between precision and recall that these approaches allow should be considered and analysed.

In Figure 8, the average ranks of both $prec_\phi^u$ and rec_ϕ^u can be found. The darker region in the centre of the X-axis in Figure 8a was to be expected in this type of problem. Those resampling percentages (higher o, lower u) result in training sets that have a comparatively much higher number of extreme cases, causing the learning algorithms to focus more on these cases. This creates a tendency to increase the rate of normal cases being predicted as highly relevant (with extreme values), lowering precision. A similar explanation can be extended to recall, although in the opposite direction (see Figure 8b). However, in the case of under-sampling, there seems to be a more noticeable additional layer of dependence, with the appearance of a more diagonal gradient. This shows an interesting interplay between α and u, and indicates that the diagonal pattern found previously in Figure 6 is mostly impacted by variations in recall.

VI. DISCUSSION

In this section we examine some aspects of our work in further detail, and motivate future work in this context. We will address two issues: *i)* the impact of data characteristics and *ii)* a local/global definition of extreme values.

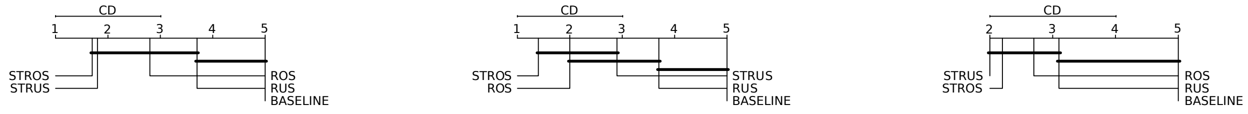
Examining how data characteristics impact the results obtained in our evaluation, we observed that the size of the data sets may be related to the efficiency of our proposals (see Section V-C). However, in order to extract deeper insights regarding the interplay of this and other characteristics and the failure/success of the methods, a multidimensional analysis of the results would be required. Such analysis should be paired with the learning algorithm used, and its own parametrization, in order to correctly assert the degree of influence that the characteristics of the data have in the outcome of applying spatio-temporal biased resampling strategies.

In this paper, we used a general concept of relevance w.r.t. the target values instead of a local approach (i.e. per location and/or time window). We recognize that there may be issues with either configuration. Our choice is based on the assumption that all events follow the same distribution, thus resulting in a general notion of relevance, as we do not consider spatial or temporal locality. Depending on the application, this may be the best decision, e.g. pollution levels should not necessarily be adjusted locally – there are health risks associated with them that are independent of local variables. In other situations, this might not be the ideal approach, and extreme values might need to be determined locally to be considered useful, e.g. an extremely high influx of costumers at a specific bike rental station should be considered in relation to normal consumer behaviour in the neighbourhood at similar periods of the year, not against a station with a much higher or lower average number of customers. In future work, we will address this issue, by comparing our results with a methodology that includes the derivation of relevance functions specific to each location.

VII. RELATED WORK

There are several ways of approaching a spatio-temporal forecasting problem by leveraging the contextual information of both spatial and temporal dimensions. Some proposals focus on adapting or combining context-aware learning models [11], [22]–[25]. Others use feature engineering to encode spatio-temporal contextual information, while taking advantage of off-the-shelf regression algorithms [12], [26]–[28]. In our work, we make use of the features proposed in [12], before applying our proposed resampling strategies.

In the context of spatio-temporal forecasting, and although the relevance of solving tasks related to the prediction of



(a) MARS (b) RF (c) RPART

Fig. 5. Critical difference diagrams for different algorithms

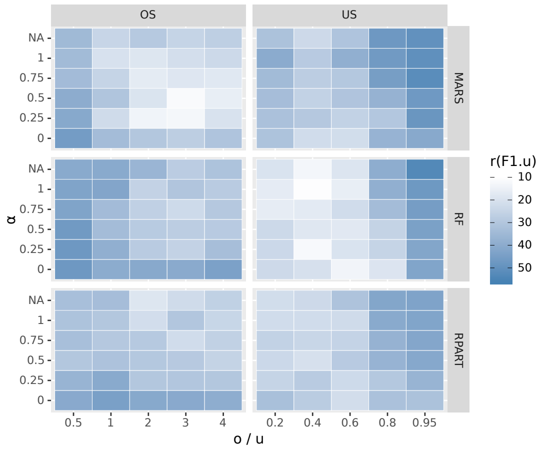


Fig. 6. Average F_1^u rank for 60(+1) different parametrizations. Ranks were calculated separately for each learning model and data set before averaging. The baseline was included in rank calculation, but excluded from the graph. Non-biased resampling is denoted by $\alpha =$ NA. Lower ranks correspond to better results

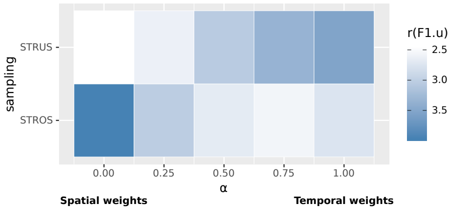
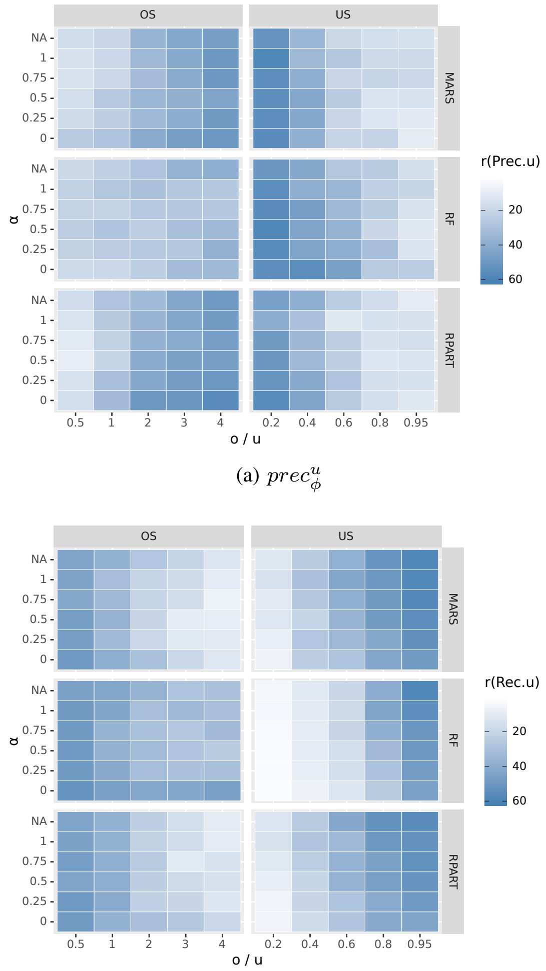


Fig. 7. Average F_1^u rank for 5 different values of α. Ranks were calculated separately for each learning model and data set before averaging. Lower ranks correspond to better results



(a) $prec_\phi^u$

(b) rec_ϕ^u

Fig. 8. Average precision and recall rank for 60(+1) different parametrizations. Ranks were calculated separately for each learning model and data set before averaging. The baseline was included in rank calculation, but excluded from the graph. Non-biased resampling is denoted by $\alpha =$ NA. Lower ranks correspond to better results

rare cases or extreme values is considerable [29], only a small fraction of this body of work specifically tackles such problems [11], [25], [27]. Furthermore, we also note that the great majority of related work concerns the prediction of rare cases, i.e. classification tasks [11], [25]. We should highlight the work of Oliveira et al. [27] and Moniz et al. [8] tackling numerical prediction of extreme values.

In the work of Oliveira et al. [27], a similar framework to the one we use is applied to the prediction of areas burnt by wildfires: features encoding spatio-temporal information are extracted from the data set, and random resampling is then applied to improve predictions of cases with extreme values. While the feature engineering step takes into account the spatio-temporal context of the problem, using random resampling ignores it. In our paper, we have proposed the first set of resampling strategies that are specific to the task of imbalanced spatio-temporal forecasting. Additionally, we have shown that taking into account the possibly different impact of the temporal and spatial dimensions is capable of boosting the predictive performance of the forecasters. In the work of Moniz et al. [8], the authors propose seminal approaches for solving imbalanced time series forecasting tasks, using a similar approach as that used in our work, i.e. incorporation of a vector weighting the relevance of each case w.r.t. the sampling process. Notwithstanding, their work is based on the analysis and pre-processing of data solely based on the temporal dimension of the data, which is not the objective in the present paper.

VIII. Conclusions

We address the problem of imbalanced spatio-temporal numerical forecasting, by proposing the first set of resampling strategies that can take advantage of the interplay between the temporal and spatial dimensions of the data. By incorporating a bias in the selection procedure of the resampling strategies, we have shown that *i)* biased resampling improves performance; *ii)* the contributions (i.e. weight) of each dimension should be optimized according to the domain; and *iii)* biased under-sampling and over-sampling approaches display opposite tendencies. Namely, in the case of spatio-temporal biased under-sampling, the strategy works best when it prefers the selection of normal cases that are distant from instances with extreme values, and gives less weight to the temporal aspect of the data. Biased over-sampling works best when it favours the selection of recent extreme instances, and attributes less weight to the spatial dimension of the data.

A thorough experimental evaluation provides extensive empirical evidence supporting the ability of the spatio-temporal biased resampling strategies to boost predictive performance towards cases with extreme values, when compared to state-of-the-art resampling strategies, in several parametrization scenarios: using internally tuned, fixed, and optimal parameters.

Acknowledgments

This work is financed by National Funds through the Portuguese funding agency, FCT - Fundação para a Ciência e a Tecnologia within the project: UID/EEA/50014/2019. Mariana Oliveira is supported by FCT/MAP-i PhD research grant (PD/BD/128166/2016). The work of L. Torgo was undertaken, in part, thanks to funding from the Canada Research Chairs program. V. Santos Costa gratefully acknowledges the project POCI-01-0145-FEDER-031356 - (PTDC/CCI-BIO/31356/2017).

References

[1] 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.

[2] A. Baxevani and R. Wilson, "Prediction of catastrophes in space over time," *Extremes*, Mar 2018.

[3] A. Eklund, T. E. Nichols, and H. Knutsson, "Cluster failure: Why fmri inferences for spatial extent have inflated false-positive rates," *Proc. of the National Academy of Sciences*, 2016.

[4] P. Branco, L. Torgo, and R. P. Ribeiro, "A survey of predictive modeling on imbalanced domains," *ACM Comput. Surv.*, vol. 49, no. 2, pp. 31:1–31:50, Aug. 2016.

[5] V. López, A. Fernández, S. García, V. Palade, and F. Herrera, "An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics," *Inform. Sciences*, vol. 250, pp. 113–141, 2013.

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

[7] R. L. Dougherty, A. Edelman, and J. M. Hyman, "Nonnegativity-, monotonicity-, or convexity-preserving cubic and quintic Hermite interpolation," *Math. of Comput.*, vol. 52, no. 186, pp. 471–494, 1989.

[8] N. Moniz, P. Branco, and L. Torgo, "Resampling strategies for imbalanced time series forecasting," *Int. J. of Data Science and Analytics*, vol. 3, no. 3, pp. 161–181, 2017.

[9] S. Pravilovic, A. Appice, and D. Malerba, "Leveraging correlation across space and time to interpolate geophysical data via CoKriging," *Int. J. Geogr. Inf. Sci.*, vol. 32, no. 1, pp. 191–212, 2018.

[10] E. Pebesma, "spacetime: Spatio-temporal data in R," *J. Stat. Softw.*, vol. 51, no. 7, pp. 1–30, 2012.

[11] Y. Zheng, F. Liu, and H.-P. Hsieh, "U-Air: When Urban Air Quality Inference Meets Big Data," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.* ACM, 2013, pp. 1436–1444.

[12] O. Ohashi and L. Torgo, "Wind speed forecasting using spatio-temporal indicators," in *Proc. Eur. Conf. Artif. Intell.*, 2012, pp. 975–980.

[13] S. M. D. from mda:mars by Trevor Hastie and R. T. U. A. M. F. utilities with Thomas Lumley's leaps wrapper., *earth: Multivariate Adaptive Regression Splines*, 2018, R package version 4.6.3.

[14] M. N. Wright and A. Ziegler, "ranger: A fast implementation of random forests for high dimensional data in C++ and R," *J. Stat. Softw.*, vol. 77, no. 1, pp. 1–17, 2017.

[15] T. Therneau and B. Atkinson, *rpart: Recursive Partitioning and Regression Trees*, 2018, R package version 4.1-13.

[16] J. Davis and M. Goadrich, "The relationship between precision-recall and roc curves," in *Proc. of ICML*. ACM, 2006, pp. 233–240.

[17] C. Elkan, "The foundations of cost-sensitive learning," in *Proc. of the Int. Joint Conf. on Artificial Intelligence*, 2001, pp. 973–978.

[18] D. R. Roberts et al., "Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure," *Ecography (Cop.)*, no. December 2016, pp. 1–17, 2017.

[19] V. Cerqueira, L. Torgo, J. Smailović, and I. Mozetič, "A comparative study of performance estimation methods for time series forecasting," in *IEEE Int. Conf. on DSAA*, 2017, pp. 529–538.

[20] M. Oliveira, L. Torgo, and V. S. Costa, "Evaluation procedures for forecasting with spatio-temporal data," in *Proc. of ECML-PKDD*, 2018, pp. 703–718.

[21] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, no. Jan, pp. 1–30, 2006.

[22] C. Barber, J. Bockhorst, and P. Roebber, "Auto-regressive HMM inference with incomplete data for short-horizon wind forecasting," in *NIPS.*, 2010, pp. 136–144.

[23] A. Appice, M. Ceci, D. Malerba, and A. Lanza, "Learning and transferring geographically weighted regression trees across time," in *MSM/MUSE*, 2011, pp. 97–117.

[24] S. Pravilovic, A. Appice, and D. Malerba, "An intelligent technique for forecasting spatially correlated time series," in *AIIA*, 2013, pp. 457–468.

[25] A. McGovern, D. J. G. II, J. K. Williams, R. A. Brown, and J. B. Basara, "Enhancing understanding and improving prediction of severe weather through spatiotemporal relational learning," *Mach. Learn.*, vol. 95, no. 1, pp. 27–50, 2014.

[26] A. Appice, S. Pravilovic, D. Malerba, and A. Lanza, "Enhancing regression models with spatio-temporal indicator additions," in *AIIA*, 2013, pp. 433–444.

[27] M. Oliveira, L. Torgo, and V. S. Costa, "Predicting wildfires - propositional and relational spatio-temporal pre-processing approaches," in *Proc. of Int. Conf, on Discovery Science*, 2016, pp. 183–197.

[28] M. Ceci, R. Corizzo, F. Fumarola, D. Malerba, and A. Rashkovska, "Predictive modeling of pv energy production: How to set up the learning task for a better prediction?" *IEEE Trans. on Ind. Informat.*, vol. 13, no. 3, pp. 956–966, 2017.

[29] 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.