Sonja Pravilovic, Massimo Bilancia, Annalisa Appice, Donato Malerba, Using multiple time series analysis for geosensor data forecasting, Information Sciences, Volume 380, 2017, Pages 31-52, ISSN 0020-0255.

Using Multiple Time Series Analysis for Geosensor Data Forecasting

Sonja Pravilovic[a], Massimo Bilancia[b], Annalisa Appice[c,d,*], Donato Malerba[c,d]

[a] *Faculty of Information Technology, Mediterranean University, Vaka Djurovica - 81000 Podgorica - Montenegro*
[b] *Ionian Department of Law, Economics and Environment, Università degli Studi di Bari Aldo Moro, Via Lago Maggiore angolo Via Ancona - 74121 Taranto - Italy*
[c] *Department of Informatics, Università degli Studi di Bari Aldo Moro, via Orabona, 4 - 70125 Bari - Italy*
[d] *CINI - Consorzio Interuniversitario Nazionale per l'Informatica*

Abstract

Forecasting in geophysical time series is a challenging problem with numerous applications. The presence of correlation (i.e. spatial correlation across several sites and time correlation within each site) poses difficulties with respect to traditional modeling, computation and statistical theory. This paper presents a cluster-centric forecasting methodology that allows us to yield a characterization of correlation in geophysical time series through a spatio-temporal clustering step. The clustering phase is designed for partitioning time series of numeric data routinely sampled at specific space locations. A forecasting model is then computed by resorting to multivariate time series analysis, in order to predict the future values of a time series by utilizing not only its own historical values, but also information from other cluster-time series. Experimental results highlight the importance of dealing with both temporal and spatial correlation and validate the proposed cluster-centric strategy in the computation of a multivariate time series forecasting model.

Keywords: Time Series Forecasting, Spatio-Temporal Clustering, Multivariate Time Series Analysis,

1. Introduction

Natural processes and physical variables (e.g. rainfall, humidity and solar radiation) are being increasingly observed over time and across space. The ubiquity of this kind of spatio-temporal data, namely geophysical time series, has motivated us to investigate and develop appropriate models to analyze and forecast them. Forecasting can be useful in providing information to decision-makers. For example, accurate forecasts can be used, in order to anticipate actions (e.g. the prediction of solar radiation in a region allows us to define the best strategy to maximize profit in the energy market).

In the last two decades, the challenge of predicting the future by looking at the past has led to a variety of time series forecasting algorithms [12, 22, 28]. They determine a time series model by accounting

☆The authors have contributed equally to this paper.
*Corresponding author (Tel: +39 (0)805443262 Fax: +39(0)805443269)
Email addresses: sonja.pravilovic@unimediteran.net (Sonja Pravilovic), massimo.bilancia@uniba.it (Massimo Bilancia), annalisa.appice@uniba.it (Annalisa Appice), donato.malerba@uniba.it (Donato Malerba)

for temporal information, estimate the model parameters and provide accurate point estimates of future values of time series. Although the majority of these algorithms is robust to unusual time series patterns and applicable to large numbers of series without user intervention, they disregard, in general, the spatial dimension of data. On the other hand, the temporal information within a site and the spatial information across distinct sites are both informative in a geophysical forecasting context.

The scenario we consider here is that of data which are routinely sampled at fixed-to-ground locations for a physical numeric field. The analysis of both spatial and temporal correlations is more complicated than modeling purely spatial or purely temporal correlations. On the other hand, the classes of spatio-temporal dependence structures differ from each other in the way in which space and time are coupled. At one extreme, space and time are considered to be independent, giving rise to the separable covariance model that allows us to represent the spatio-temporal correlation function as the product of a spatial and temporal term. Otherwise, at the cost of a heavier computational burden, non-separable space-time models can be considered by including suitable parameters, that indicate the strength of the interaction between the spatial and temporal components [18]. In addition, as non-stationarity and anisotropy are usual characteristics of geophysical data, if they are present and unaccounted for in the geo-statistical model development, they can result in poorly specified models, as well as in inappropriate spatial-temporal inferences and predictions.

In view of the prohibitive costs of monitoring spatially and temporally dense networks, a spatio-temporal continuous model (based on observations at a limited number of monitoring stations) can be developed. Such a model provides spatial interpolation of the involved random fields and makes short-term daily predictions [32]. However, if spatial interpolation is not the primary objective and there is a target for forecast accuracy along the temporal dimension, it is often useless to have a model with a spatially continuous variable. On the other hand, there is no doubt that spatio-temporal interpolation, in whatever form it comes, can potentially provide more accurate predictions than temporal forecasting alone.

To this end, we take a different approach that attempts to convey correlation information along both the temporal and a discrete spatial dimension of data. We illustrate a novel spatio-temporal forecasting methodology, called cVAR (spatio-temporal Cluster-based Vector AutoRegressive model), that learns a spatio-temporal model of historical geophysical time series data and uses this model to yield accurate forecasts of these time series, considering that the temporal distribution of a random field can undergo a slow spatial change. We propose a data-driven approach, including a clustering phase and a forecasting phase.

The *clustering phase* is data-driven by the spatial location of the time series, as well as by the time-stamped values of the time series. It constructs a spatio-temporal cluster model of the past data. Each cluster collects time series whose observations exhibit (almost approximately) stationary correlation over space in each time point of the series. Simultaneously, the behavior of a cluster is not necessarily stable over time, as observations can change over consecutive time points. However, intra-cluster series will exhibit similar temporal patterns (for example, strongly-interacting time series show potential movement in one series when the other time series move, as well as a strong concordance of the signs of increments).

The *forecasting phase* is inspired by the idea that temporal and spatial correlations govern data grouped

in a cluster. Spatially-coupled variables are constructed from the clustered time series. They are observed along the time series points defining a set of suitably defined time series, which are introduced into the information set of each geosensor site, in order to deal with temporal and spatial correlations simultaneously.

In this study, clustering is done using a partition-based machine learning technique with a newly defined spatio-temporal dissimilarity. The dissimilarity treats space and time independently, giving more similarity to closest pairs of geo-locations, with the additional constraint that similarity of one pair is strengthened if exponential smoothing of the corresponding time series leads to similar one-step-ahead forecasts. Regarding multivariate forecasting, we consider the stationary vector auto-regressive (VAR) model [49], in order to analyze the structure of this multivariate system of variables. In this way, forecasts for a given target time series are a linear combination of past target data, as well as of past spatially-coupled background data which summarize spatio-temporal information of the network. A stationary model VAR has some limitations, and without modification standard VARs miss nonlinearities, periodic components, as well as stochastic drifts. Given that cycles of a regular nature are a minor problem that can be easily incorporated in VARs, we deliberately disregard verifying whether other non-standard and non-stationary dynamic patterns are present or not (and eventually incorporate them into the model). Although the proposed algorithm should be considered a heuristic, we can demonstrate that our multivariate system (without any human intervention and manual parameter setting) is often more accurate than existing univariate time series models.

In short, the specific contributions of this paper are highlighted as follows: (1) We describe a spatio-temporal dissimilarity measure that gives more similarity to the closest pairs of geo-locations that exhibit the most similar one-step-ahead forecasts of the measured variable. (2) We define a system that couples an observation site with a multivariate system including the target time series observed at the site and a number of cluster-defined spatially-coupled time series. (3) We apply a multivariate time series solution to each site, in order to analyze the structure of this system of variables and construct an accurate predictor of the target time series, with the help of the spatially coupled variables. (4) We demonstrate the importance of dealing with both temporal and spatial correlation, in order to yield accurate forecasts of geophysical data and validate the accuracy of the proposed cluster-centric strategy in the computation of a multivariate time series forecasting model.

The paper is organized as follows. The related work will be briefly reviewed in Section 2. The mathematical notation is introduced in Section 3. The clustering step is described in Section 4, while the spatio-temporal multiple variable synthesis is presented in Section 5. A description of the multivariate forecasting step, based on VAR modeling, is reported in Section 6. The time-complexity analysis of the proposed algorithmic pipeline is analyzed in Section 7. Experiments with real data are presented in Section 8, where discovered cluster structures are inspected and forecasting results are discussed. Finally, Section 9 refocuses on the purpose of the research, draws conclusions and proposes future developments.

2. Background

We review the state-of-the-art in spatio-temporal clustering and time series forecasting, as they are the key ingredients of our proposal.

2.1. Spatio-temporal clustering

A few algorithms have been developed to cluster different types of spatio-temporal data (see [25] for a recent survey). Putting their differences aside and focusing on clustering geophysical time series, we consider clustering algorithms based on dissimilarity measures. Their measures include both temporal dissimilarity and spatial dissimilarity, as well as the interaction between temporal and spatial information.

Qui et al. [40] have formulated a distance measure, including both spatial dissimilarity and temporal dissimilarity. They have incorporated the proposed dissimilarity into fuzzy C-means clustering. Birant and Kut [7] have extended the standard density-based clustering by introducing two different distance parameters: the former is used to measure the spatial closeness of two points; the latter is used to measure the similarity of non-spatial values. Liu et al. [27] have defined a spatio-temporal density-based algorithm, called STSNN (Spatio-Temporal Shared Nearest Neighbor). The algorithm resorts to a similarity measure that counts the number of neighbors shared on a data window. More recently, Appice et al. [1] have defined a clustering technique, called SUMATRA, that segments the geophysical time series into temporal windows. It computes clusters of spatially-close geo-referenced data, which vary according to a similar trend along the window time horizon. An incremental version of this clustering procedure is illustrated in [2]. Finally, Appice et al. [3] have defined an algorithm to detect spatial clusters by processing local indicators of spatial autocorrelation of data observed at a specific time point. Spatio-temporal clusters are constructed by grouping time series, whose sites are repeatedly assigned to the same spatial cluster over a time window.

All the algorithms described above may produce a model, in both space and time, of the correlation structure. Sometimes they deal with the spatial and temporal dimensions separately in progressive clustering steps. Sometimes they deal with both dimensions simultaneously in a single clustering step. The latter approach, that is faster, is closely followed in this paper. However, in all the algorithms described above, clustering is driven by the past data, without looking at the forecast evolution of the correlation structure.

2.2. Time series forecasting

Time series forecasting is often an empirical procedure that identifies the model (from a larger class of models) which is more suitable to the data at hand. This model can be used to extrapolate predictions of the future observations in the series. The model can be iteratively improved by human intervention, in terms of its empirical average forecasting error at several forecasting horizons. Several well-known models have been formulated in univariate time series analysis [29], when a single variable is observed over time, as well as in multivariate time series analysis [28], when a system of variables is monitored over time.

Multiple geophysical time series are traditionally dealt with separately, so that the forecasting model associated with a specific site is learned by neglecting the spatial-aware information enclosed in the excluded

sites. However, in recent years, research has started investigating specific techniques, which accommodate the spatial correlation in the forecasting model. Existing studies are mainly based on the univariate analysis, and a few studies have accounted for developments in multivariate analysis for this kind of data.

By focusing on univariate analysis, the first approach considering spatial correlation is investigated in [24]. This approach, called STARIMA, considers space-time auto-regressive integrated moving average models. It models a geosensor time series as a linear combination of past observations and disturbances at neighboring sites. Through the specification of a weighting matrix, STARIMA can reflect the standard idea that near sites exert more influence over each other than distant ones. More recently, Pravilovic et al. [38] have formulated a technique, called sARIMA, to identify the order of an ARIMA model, without any human intervention. sARIMA accounts for the property of spatial correlation to determine the number of ARIMA coefficients for a specific time series. It uses a spatial-aware formulation of the AICc automatic order-selection criterion, which is computed on data recorded on both a given geosensor and its neighboring stations (see Subsection 8.4.2 for additional details). Pokrajac and Obradovic [37] have produced spatio-temporal forecasts using a generalization of the standard spatial auto-regression and including a disturbance term modeled as a temporal auto-regression. A further approach has been investigated in [45], in which linear mixed models (LMMs) with spatial random effects are proposed in a Bayesian framework. The spatial correlation is taken into account with a Conditional Auto Regressive (CAR) prior distribution for spatial effects. Finally, Ohashi and Torgo [33] have described an approach sharing a few similarities with the one presented in this paper. They determine forecasts of each series based on the past values of the series up to a certain time window of fixed length, as well as a set of technical indicators (e.g. tendency, acceleration and momentum). These technical indicators, which are very frequently used in financial forecasting, are computed for each time stamp as summaries of certain properties of the time series in a neighborhood. They can also be regarded as additional descriptors of the dynamic behavior of the time series to forecast.

Considering the multivariate analysis, De Luna and Genton [10] propose a suitable VAR model identification strategy, taking advantage of the spatial location of the different time series, which is particularly useful when data is rich in the time dimension but sparse in the spatial dimension, and the main objective is to provide time-forward predictions (as in our case). A strategy based on multivariate VAR models is discussed in [5] as well.

All the techniques described above deal with the spatial information in the time series analysis. In any case, all assume that spatial correlation is stationary over a circular/conic neighborhood with fixed size. This static analysis of spatial correlation may suffer from serious limitations, as spatial correlation can often be manifested with different underlying latent structures of the space, which can vary in shape and size over data. This limit is overcome in [39], where a meaningful model of the data structure is discovered by resorting to a spatio-temporal clustering step. The cluster information is integrated in the ARIMA model, yielding accurate forecasts (see Subsection 8.4.2). In the following, we further elaborate on this idea.

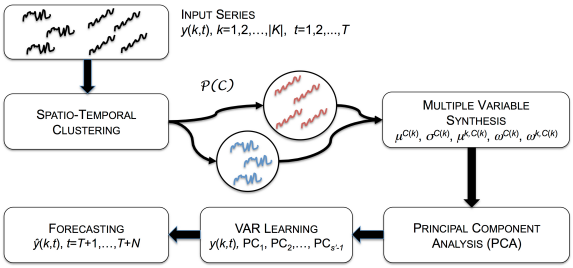

Figure 1: Block diagram of cVAR (spatio-temporal Cluster-based Vector AutoRegressive model).

3. Data and methodology definition

Let $\mathcal{D}(K, Y, T)$ be a geophysical time series dataset, so that: K is a set of geo-locations over a given spatial domain, Y is a numeric geophysical variable and T is the size of a temporal window that is discretized in equally-spaced time points denoted as $t = 1, 2, \ldots, T$. In this data setting, $y(k, t)$ denotes the sequence of geo-referenced measures of Y collected at a certain projected geolocation $k \in K$ for each time point $t = 1, 2, \ldots, T$. We present a novel spatio-temporal forecasting methodology, called cVAR (see Figure 1), which inputs dataset $\mathcal{D}(K, Y, T)$ and consists of a pipeline of three algorithmic steps:

1. A clustering pattern $\mathcal{P}(\mathcal{C})$ partitioning K into distinct clusters on the ground of the spatial and temporal similarity of the time series of $\mathcal{D}$ (see Section 4 for details). A peculiarity of our approach is the definition of a novel spatio-temporal dissimilarity (see Subsection 4.1), which includes both time series dissimilarity and spatial distance as separate contributions. The objective is that the time series of $\mathcal{D}$ which manifest a high degree of spatio-temporal correlation are grouped into the same cluster, while those series which manifest abrupt variation with each other are grouped into different clusters.
2. A feature expansion mechanism synthesizing the additional time series, borrowing from the spatial and temporal-aware information that is enclosed in the clustering pattern $\mathcal{P}(\mathcal{C})$ (see Section 5). The Principal Component Analysis (PCA) transforms the multiple variables constructed per geosensor into a set of latent orthogonal factors, which are useful to improve forecasting of the target series.
3. A stationary model VAR $\hat{y}(k, t)$ (see Section 6) for each target time series $y(k, t)$, constructed by exploiting the additional time series defined in the previous step. This model can be used to forecast new data points $\hat{y}(k, T + 1), \ldots, \hat{y}(k, T + N)$ for a suitable forecasting horizon N.

4. Spatio-temporal clustering

The clustering step is performed by resorting to a partition-based algorithm that accepts dissimilarity data. In particular, time series are partitioned around representative medoids (i.e. representative time series of the dataset), which are selected so that total dissimilarity of all time series to their nearest medoid is minimal. The dissimilarity is evaluated over space and time simultaneously. The number of final clusters is determined by looking for a local maximum of quality of clustering.

4.1. Spatio-temporal dissimilarity

Let k_i and k_j be two geo-locations ($k_i, k_j \in K$), so that $(x_1^{(i)}, x_2^{(i)})$ and $(x_1^{(j)}, x_2^{(j)})$ denote the spatial coordinates of k_i and k_j, respectively. The spatio-temporal dissimilarity between the time series $y(k_i, t)$ and $y(k_j, t)$ is determined by computing the spatial distance between spatial coordinates ($\mathsf{Sdiss}(\cdot,\cdot)$ of k_i and k_j), as well as a suitable dissimilarity between the time series values ($\mathsf{Tdiss}(\cdot,\cdot)$ of k_i and k_j). Formally,

$$\mathsf{diss}(k_i, k_j) = \mathsf{Sdiss}(k_i, k_j) + \mathsf{Tdiss}(k_i, k_j). \tag{1}$$

The spatial dissimilarity $\mathsf{Sdiss}(k_i, k_j)$ is computed, with squared Euclidean distance, from the normalized spatial coordinates of k_i and k_j, that is,

$$\mathsf{Sdiss}(k_i, k_j) = (\tilde{x}_1^{(i)} - \tilde{x}_1^{(j)})^2 + (\tilde{x}_2^{(i)} - \tilde{x}_2^{(j)})^2. \tag{2}$$

To compute Formula 2, spatial coordinates are constrained to $[-1, 1] \times [-1, 1]$ as follows:

$$\tilde{x}_h = \frac{x_h - \langle x_h; K\rangle}{\max(|x_h - \langle x_h; K\rangle|)}, \text{ with } h = 1, 2, \tag{3}$$

where $\langle x_h; K\rangle$ is the average value of the corresponding component of the spatial coordinates falling in K.

The temporal dissimilarity $\mathsf{Tdiss}(k_1, k_2)$ is computed with simple exponential smoothing (SES) from the time series of the squared element-wise Euclidean distances, calculated between the suitably transformed time series values of k_i and k_j, that is:

$$\begin{aligned}
\mathsf{Tdiss}(k_i, k_j) &\stackrel{\mathsf{def}}{=} \mathsf{Tdiss}(k_i, k_j, T), \text{ with} && (4)\\
\mathsf{Tdiss}(k_i, k_j, T) &= \alpha\left[\tilde{y}(k_i, T) - \tilde{y}(k_j, T)\right]^2 + (1-\alpha)\mathsf{Tdiss}(k_i, k_j, T-1) && (5)\\
&\vdots \\
\mathsf{Tdiss}(k_i, k_j, 2) &= \alpha\left[\tilde{y}(k_i, 2) - \tilde{y}(k_j, 2)\right]^2 + (1-\alpha)\mathsf{Tdiss}(k_i, k_j, 1),\\
\mathsf{Tdiss}(k_i, k_j, 1) &= \alpha\left[\tilde{y}(k_i, 1) - \tilde{y}(k_j, 1)\right]^2 + (1-\alpha)\ell_0, && (6)
\end{aligned}$$

where $0 \leq \alpha \leq 1$ is a smoothing parameter and $\ell_0 = \mathsf{Tdiss}(k_i, k_j, 0)$. Formula 5, that represents the one-step-ahead forecast at time $T+1$, can be written as the following weighted average of all past observations:

$$\mathsf{Tdiss}(k_i, k_j, T) = \sum_{u=0}^{T-1} \alpha(1-\alpha)^u \left[\tilde{y}(k_i, T-u) - \tilde{y}(k_j, T-u)\right]^2 + (1-\alpha)^T \ell_0. \tag{7}$$

Measuring the dissimilarity between the values, which have been forecasted for the compared time series, is a way to quantify the strength of the correlation between data that we expect to observe in the near future. In particular, the more the one-step-ahead forecasts (see Formula 4) are close to zero, the more the corresponding time series are considered to be similar. The rate at which the weights in Formula 7 decrease is controlled by the parameter α (see [21] for details). If α is close to zero then more weight is given to observations from the distant past. If α is close to 1 then more weight is given to the more recent observations. If $\alpha = 1$, then the one-step-ahead forecast coincides with the naïve forecast

$\mathsf{Tdiss}(k_i, k_j, T) = [\tilde{y}(k_i, T) - \tilde{y}(k_j, T)]^2$. The set-up of α is discussed in Subsection 4.2. In Formulae 5-6, $\tilde{y}(k, t)$ is computed by applying a normalization function to $[-1, +1]$, using a transformation mathematically equivalent to Formula 3 and defined as follows:

$$\tilde{y}(k, t) = \frac{y(k, t) - \langle y(k, t) \rangle_t}{\max(|y(k, t) - \langle y(k, t) \rangle_t|)}, \tag{8}$$

where $\langle y(k, t) \rangle_t$ is the time average of the time series collected at geo-location $k \in K$.

4.2. Setting the smoothing parameter α and the initial value ℓ_0

A standard, computationally fast set-up is $\alpha = 0.5$, $\ell_0 = \mathsf{Tdiss}(k_i, k_j, 0) \stackrel{\mathsf{def}}{=} [\tilde{y}(k_i, 1) - \tilde{y}(k_j, 1)]^2$. An alternative is to estimate α along the initial value ℓ_0 from the observed data [21]. Let us consider the one-step-ahead in-sample forecasting errors, which are defined as follows for $t = 2, \ldots, T$:

$$\begin{aligned} \hat{\tilde{e}}(k_i, k_j, t) &= [\tilde{y}(k_i, t) - \tilde{y}(k_j, t)]^2 - \mathsf{Tdiss}(k_i, k_j, t-1) = \\ &= [\tilde{y}(k_i, t) - \tilde{y}(k_j, t)]^2 - \sum_{u=0}^{t-2} \alpha(1-\alpha)^u \left[\tilde{y}(k_i, t-u-1) - \tilde{y}(k_j, t-u-1)\right]^2 + (1-\alpha)^{t-1}\ell_0, \end{aligned}$$

while, for $t = 1$, we simply have $\hat{\tilde{e}}(k_i, k_j, 1) = [\tilde{y}(k_i, 1) - \tilde{y}(k_j, 1)]^2 - \ell_0$. For each pair, k_i and k_j, we find the smoothing parameter α and the initial value ℓ_0 which minimize the in-sample sum-of-squares error:

$$\mathsf{SSE}(\alpha, \ell_0) = \sum_{t=1}^{T} \hat{\tilde{e}}(k_i, k_j, t)^2. \tag{9}$$

The range of summation in Formula 9 is meaningful, given that ℓ_0 is the one-step-ahead forecast for $t = 1$. The minimization of the quadratic form in Formula 9 is a non-linear minimization problem that does not admit closed-form solutions. It is solved numerically using the Levenberg-Marquardt algorithm [22].

The automatic procedure performs a global choice to set-up α. In fact, a global value α is determined, independently of time t, in order to optimize the performance of the smoothing parameter with t varying between 1 and T [8]. As an alternative, a local choice of the smoothing parameter can be considered [8, 52]. A local choice would aim at choosing a local value $\alpha(t)$, that depends on time t, in order to optimize the performance of the smoothing parameter at each time point t. In this way, every $\alpha(t)$ (with t varying between 1 and T) would be selected from data collected up to t and then used in the computation of Formulae 4-6. As observed in [8], a local choice of any smoothing parameter is more flexible than the global one and the resulting smoothing is more capable of adapting to the dynamic change of the underlying time series. At the same time, the local choice is harder and more variable, since only the local data are involved in choosing the estimates. Based upon this analysis, we have favored simplicity and stability by opting for the global choice. In any case, the viability of this decision is also investigated in the empirical study, where we compare the performance of the proposed forecasting approach with the choice of a global value α to that of the same approach with the choice of local values $\alpha(t)$ (see details in Section 8.4.3).

4.3. Clustering algorithm

As a partition-based algorithm, we consider PAM (Partitioning Around Medioids) [43, 48]. This algorithm is selected as the existing distance-based clustering approaches can be extended to the considered data setting only when the distance function can be defined via a suitable inner product between two data points. In fact, in this case, we are able to kernelize the algorithm (kernel k-means being the prototypical example). However, it is not immediately apparent how to express the spatio-temporal dissimilarity (see Formula 1) in terms of a suitable inner product. On the other hand, PAM, that requires only an arbitrary matrix of dissimilarities as input, seems to be a mathematically simpler alternative than kernel k-means in our setting.

More specifically, PAM inputs: (1) an integer g (with $g_{\min} \leq g \leq |K|$) (by default $g_{\min} = 2$), that is, the number of clusters to discover and (2) a $|K| \times |K|$ dissimilarity matrix D, where $\mathsf{diss}(k_i, k_j) = \mathsf{diss}(k_j, k_i)$ measures the 'difference' between k_i and k_j computed according to Formula 1. It outputs a clustering pattern $\mathcal{P}(\mathcal{C})$, that is, a set of g clusters $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_g$, such that: (1) $\emptyset \notin \mathcal{P}(\mathcal{C})$, $\bigcup_{\mathcal{C}_i \in \mathcal{P}(\mathcal{C})} \mathcal{C}_i = K$ and (2) $\forall\, \mathcal{C}_i, \mathcal{C}_j \in \mathcal{P}(\mathcal{C}), \text{if } i \neq j \text{ then } \mathcal{C}_i \cap \mathcal{C}_j = \emptyset$. This pattern is determined by partitioning the geo-location set K around a subset of medoid geo-locations $\{k_{m_1}, \ldots, k_{m_g}\} \subset \{k_1, k_2, \ldots, k_{|K|}\}$, which minimize the objective function $\sum_{i=1}^{|K|} \min_{\ell=1,2,\ldots,g} \mathsf{diss}(k_i, k_{m_\ell})$.

Procedurally, the medoid set $M = \{k_{m_1}, \ldots, k_{m_g}\}$ is determined by resorting to the steepest ascent hill climber. Unlike other partition-based clustering algorithms (such as k-means and k-medoids), PAM does not need initial random guesses for the cluster centers. In fact, it constructs a 'reasonable' initial medoid set, with a deterministic building phase, at the cost of a mild additional complexity. Then, in each iteration of the swap phase, it attempts to improve the medoid set and increase the quality of the clustering. This is done by selecting pairs $(k_i, k_{m_j}) \in (K - M) \times M$ that produce the best decrease in the objective function when their roles are switched. Each $k_i \in K$ is assigned to the cluster corresponding to the nearest medoid.

4.4. Choosing the number of clusters

To automate the choice of g, we use the Silhouette index to measure how well each time series lies within its cluster [44]. The Silhouette index for geosensor k_i, given a cluster set $\mathcal{P}(\mathcal{C})$, is computed as $\mathsf{Sil}(k_i, \mathcal{P}(\mathcal{C})) = \frac{b(k_i) - a(k_i)}{\max\{b(k_i), a(k_i)\}}$, where $a(k_i)$ is the average spatio-temporal dissimilarity (computed according to Formula 1) of k_i from every other k_j in the same cluster, while $b(k_i)$ is the lowest average spatio-temporal dissimilarity of k_i to any other cluster of which k_i is not a member. $\mathsf{Sil}(k_i, \mathcal{P}(\mathcal{C}))$ ranges in $[-1, 1]$. As $b(k_i)$ captures the degree to which k_i is spatio-temporally separated from other clusters, a value of $\mathsf{Sil}(k_i, \mathcal{P}(\mathcal{C}))$ closer to 1 means that the cluster containing k_i is compact and that k_i is 'far away' from other clusters.

The average index over all geo-locations $k_i \in K$ is a clustering quality index, measuring how tightly all the geophysical time series are grouped. It is formulated as follows:

$$\mathsf{Sil}(K, \mathcal{P}(\mathcal{C})) = \frac{1}{|K|} \sum_{k_i \in K} \mathsf{Sil}(k_i, \mathcal{P}(\mathcal{C})). \tag{10}$$

The average index is used to determine the number of clusters as follows. We start with the number of clusters set equal to an initial $g_{\min}$ guess, apply PAM with the specified number of clusters and use the discovered clustering model, in order to compute $\mathsf{Sil}(K, \mathcal{P}(\mathcal{C}))$. We iterate the execution of PAM by varying the number of cluster g between $g_{\min}$ and $g_{\max}$, so that $2 \leq g_{\min} \leq g_{\max} \leq |K| - 1$. Finally, we select the number of cluster g_{est}, for which $\mathsf{Sil}(K, \mathcal{P}(\mathcal{C}))$ achieves a global maximum peak on $[g_{\min}, g_{\max}]$. According to the subjective interpretation of $\mathsf{Sil}(K, \mathcal{P}(\mathcal{C}))$ reported in [48], a significant clustering structure can be found whenever $\mathsf{Sil}(K, \mathcal{P}(\mathcal{C})) > 0.5$ in the final partition $\mathcal{P}(\mathcal{C})$.

We note that the average Silhouette index is an internal measure of clustering quality based on a given spatio-temporal dissimilarity and, as such, is not suitable for detecting irregularly shaped clusters. In particular, it will tend to agree with distance or dissimilarity-based algorithms, such as PAM, which cannot identify clusters of irregular shapes. This phenomenon can be seen as a subtle form of overfitting, which may create a tendency to overestimate the true number of clusters. However, this problem could be alleviated by the the peculiar structure of the proposed spatio-temporal dissimilarity, as the compactness of the obtained clusters may not coincide with the spatial geometric compactness and connectivity of clustered geo-locations. In Section 9, we briefly mention a possible external solution to these difficulties.

5. Spatio-temporal multiple variable synthesis

Although multivariate time series models often yield forecasts that are superior to those from univariate models [53], the application of a multivariate framework to a geosensor system is not straightforward. One possibility is to learn a simultaneous multivariate model, including all geosensor locations. However, this model building strategy still disregards the spatial dependence structure inherent to a geosensor network. As an alternative, we can consider a multivariate system for each geosensor location. This strategy is common, for example, in technical analysis of financial time series, where a single univariate time series is transformed into a multiple time series system consisting of various technical indicators, such as oscillators and momentum/volatility time series. However, also in this case, we should estimate and forecast $|K|$ separate multivariate models, without accounting for the spatial dimension.

In this study, this issue is addressed by borrowing knowledge from the spatial and temporal clustering pattern $\mathcal{P}(\mathcal{C})$. There are many non-equivalent ways of generating multiple variables per geosensor. By resorting to a weakly stationary model (e.g. the one proposed in Section 6), it is natural to consider second-order characteristics (i.e. mean and variance as a function of time) of the underlying data, averaged over the sensors grouped in each cluster $\mathcal{C}(k)$ (in accordance with the hypothesis that series which are grouped in the same cluster manifest a coherent temporal evolution). Using a cluster-wise mean, we can also define a cluster-wise speed measuring the local rate of change of each series included in the same cluster. In fact, speed is often unrelated to variance and can be seen as a useful independent predictor.

On the other hand, computing cluster-wise mean, variance and speed allows us to construct the same multivariate system for each geosensor falling into the same subset of the partition $\mathcal{P}(\mathcal{C})$. Our point of view is

that constraining sensors spanned across a possibly large cluster region to the same multivariate system can be seen as a modeling limit. Therefore, we consider it appropriate to add a geosensor-specific transformed series, that is constructed by means of a suitable distance-based averaging of the values recorded for the time series grouped in the same cluster (giving less weight to more distant sensors, see also [33]). In this way, a geosensor-specific speed series can be defined as well.

Formally, let $\mathcal{P}(\mathcal{C})$ be a spatio-temporal clustering pattern of dataset $\mathcal{D}(K, Y, T)$. For each time series location $k \in K$, denoting as $\mathcal{C}(k)$ (with $\mathcal{C}(k) \in \mathcal{P}(\mathcal{C})$) the cluster that groups the transformed time series $\tilde{y}(k,t)$ (according to Formula 8), five additional time series can be defined by accounting for the spatial and temporal-aware information enclosed in $\mathcal{C}(k)$. These variables are introduced, in order to yield a more accurate forecast for $y(k,t)$. They are calculated for all time points $t \in T$ and denoted as:

$$\left(\mu^{\mathcal{C}(k)}(t), \sigma^{\mathcal{C}(k)}(t), \mu^{k,\mathcal{C}(k)}(t), \omega^{\mathcal{C}(k)}(t), \omega^{k,\mathcal{C}(k)}(t)\right). \tag{11}$$

Mean variable $\mu^{\mathcal{C}(k)}(t)$. This defines the time series measuring the average value of the values recorded for the time series grouped in cluster $\mathcal{C}(k)$. Formally,

$$\mu^{\mathcal{C}(k)}(t) = \frac{1}{|\mathcal{C}(k)|} \sum_{k_i \in \mathcal{C}(k)} \tilde{y}(k_i, t). \tag{12}$$

Standard deviation variable $\sigma^{\mathcal{C}(k)}(t)$. This defines the time series measuring the standard deviation of the values recorded for the time series grouped in cluster $\mathcal{C}(k)$. Formally,

$$\sigma^{\mathcal{C}(k)}(t) = \sqrt{\frac{1}{|\mathcal{C}(k)|} \sum_{k_i \in \mathcal{C}(k)} \left(\tilde{y}(k_i, t) - \mu^{\mathcal{C}(k)}(t)\right)^2}. \tag{13}$$

Weighted mean variable $\mu^{k,\mathcal{C}(k)}(t)$. This defines the time series measuring a weighted average of the values recorded for the time series grouped in cluster $\mathcal{C}(k)$. Unlike $\mu^{\mathcal{C}(k)}(t)$ and $\sigma^{\mathcal{C}(k)}(t)$, which assume constant values inside cluster $\mathcal{C}(k)$, the indicator defined below varies across geo-locations $k \in K$. Formally,

$$\mu^{k,\mathcal{C}(k)}(t) = \frac{\sum_{k_i \in \mathcal{C}(k)} \mathsf{Sdiss}(k_i, k)\tilde{y}(k_i, t)}{\sum_{k_i \in \mathcal{C}(k)} \mathsf{Sdiss}(k_i, k)}, \quad t = 1, \ldots, T, \tag{14}$$

where $\mathsf{Sdiss}(\cdot, \cdot)$ is the spatial distance, computed as described in Formula 2. For example, supposing that a cluster groups sensors indexed as $k = 1, 3, 4$ (hence $\mathcal{C}(1) = \mathcal{C}(3) = \mathcal{C}(4)$), we have:

$$\begin{aligned}
\mu^{1,\mathcal{C}(1)}(t) &\propto \mathsf{Sdiss}(1,1)\tilde{y}(1,t) + \mathsf{Sdiss}(3,1)\tilde{y}(3,t) + \mathsf{Sdiss}(4,1)\tilde{y}(4,t) \\
\mu^{3,\mathcal{C}(3)}(t) &\propto \mathsf{Sdiss}(1,3)\tilde{y}(1,t) + \mathsf{Sdiss}(3,3)\tilde{y}(3,t) + \mathsf{Sdiss}(4,3)\tilde{y}(4,t) \\
\mu^{4,\mathcal{C}(4)}(t) &\propto \mathsf{Sdiss}(1,4)\tilde{y}(1,t) + \mathsf{Sdiss}(3,4)\tilde{y}(3,t) + \mathsf{Sdiss}(4,4)\tilde{y}(4,t).
\end{aligned}$$

Speed variable $\omega^{\mathcal{C}(k)}(t)$. This defines the time series measuring the ratio of the average values recorded in $\mu^{\mathcal{C}(k)}(t)$. Formally,

$$\omega^{\mathcal{C}(k)}(t) = \frac{\mu^{\mathcal{C}(k)}(t)}{\mu^{\mathcal{C}(k)}(t-1)}, \quad t = 2, \ldots, T. \tag{15}$$

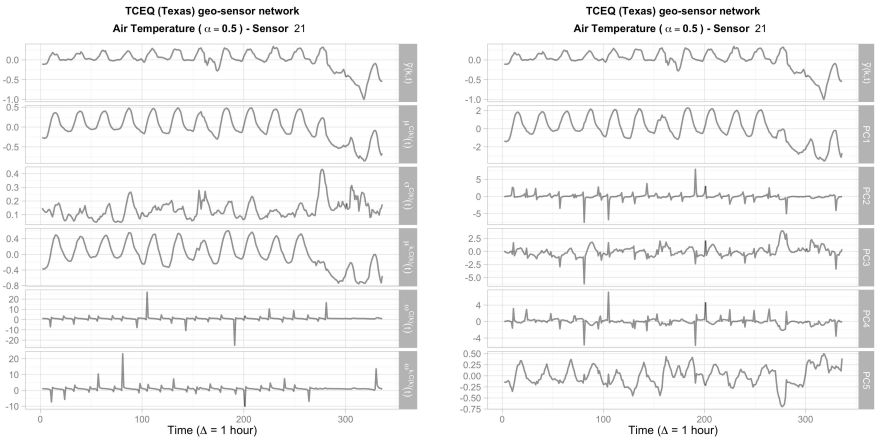

Figure 2: Variable *Air Temperature* (TCEQ geosensor network, $T = 336$, see Section 8 for further details) recorded by a geosensor. The partition algorithm (see Section 4) has been run with $\alpha = 0.5$. Left: the transformed time series $\tilde{y}(k,t)$ and the five additional time series (see Formula 11), defined to account for spatial and temporal information enclosed in cluster $\mathcal{C}(k)$, to which geosensor k is assigned. Right: transformed time series $\tilde{y}(k,t)$ and principal component scores of the additional time series. These are ordered so that PC_1 has the largest sample variance, PC_2 has the second largest sample variance, and so on.

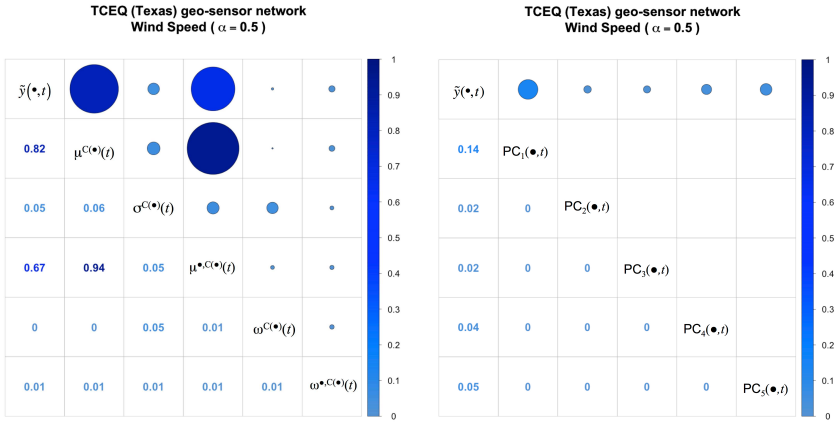

Figure 3: Variable *Wind Speed* (TCEQ geosensor network, $T = 336$, see Section 8 for further details). The partition algorithm (see Section 4) has been run with $\alpha = 0.5$. Left: average correlation matrix, averaged across sensors, of the correlation matrices of the full information set $\big(\tilde{y}(k,t), \mu^{\mathcal{C}(k)}(t), \sigma^{\mathcal{C}(k)}(t), \mu^{k,\mathcal{C}(k)}(t), \omega^{\mathcal{C}(k)}(t), \omega^{k,\mathcal{C}(k)}(t)\big)$. Right: average correlation matrix, averaged across sensors, of the correlation matrices of the partially orthogonalized information set $(\tilde{y}(k,t), \mathsf{PC}_1(k,t), \ldots, \mathsf{PC}_5(k,t))$.

Weighted speed variable $\omega^{k,\mathcal{C}(k)}(t)$. It defines the time series measuring the ratio of the weighted average values recorded in $\mu^{k,\mathcal{C}(k)}(t)$. In the same way as $\mu^{k,\mathcal{C}(k)}(t)$ and unlike $\omega^{\mathcal{C}(k)}(t)$, the latter assuming constant values inside cluster $\mathcal{C}(k)$, the indicator defined below varies across geo-locations $k \in K$. Formally,

$$\omega^{k,\mathcal{C}(k)}(t) = \frac{\mu^{k,\mathcal{C}(k)}(t)}{\mu^{k,\mathcal{C}(k)}(t-1)}, \quad t = 2, \ldots, T. \tag{16}$$

5.1. Post-processing

We note that variables $\mu^{\mathcal{C}(k)}(t)$ and $\mu^{k,\mathcal{C}(k)}(t)$ may be strongly contemporaneously correlated with each other. At each time point, they can be interpreted as a linear combination of the same set of observations (i.e. the observations from time series falling into the same subset of the clustering pattern $\mathcal{P}(\mathcal{C})$). In particular, Formulae 12 and 14 differ from each other only in the coefficients of the linear combination. An illustration of this phenomenon is shown on the left side of Figure 2. As a further illustration of the presence of collinearity, we can see the left side of Figure 3. It shows the average correlation matrix, averaged across sensors, of the correlation matrices of the full information set $\left(\tilde{y}(k,t), \mu^{\mathcal{C}(k)}(t), \sigma^{\mathcal{C}(k)}(t), \mu^{k,\mathcal{C}(k)}(t), \omega^{\mathcal{C}(k)}(t), \omega^{k,\mathcal{C}(k)}(t)\right)$, including both the observed time series and the newly generated ones. On average, as k varies, there is a strong contemporaneous correlation between $\tilde{y}(k,t)$ and $\mu^{\mathcal{C}(k)}(t)$, as well as between $\tilde{y}(k,t)$ and $\mu^{k,\mathcal{C}(k)}(t)$ (and such a correlation pattern is independent of the peculiar dataset). Taking this view, suppose that, for each geosensor k_i, the forecasting problem now includes the following multivariate system of time series:

$$\mathbf{z}(k_i,t) = \left(z^{(1)}(k_i,t), \ldots, z^{(s)}(k_i,t)\right) = \left(\tilde{y}(k_i,t), \mu^{\mathcal{C}(k_i)}(t), \sigma^{\mathcal{C}(k_i)}(t), \mu^{k_i,\mathcal{C}(k_i)}(t), \omega^{\mathcal{C}(k_i)}(t), \omega^{k_i,\mathcal{C}(k_i)}(t)\right). \tag{17}$$

We can try to learn a weakly stationary VAR (vector auto-regression, [28]) model of this multiple time series and use it, in order to forecast the target series $z^{(1)}(k_i,t) \equiv \tilde{y}(k_i,t)$ up to N-step ahead. The general form of a model VAR includes a set of s lagged equations, which can be written in the following vector form:

$$(\mathbf{I} - \mathbf{\Phi}_i(L))\mathbf{z}(k_i,t)^\top = \mathbf{c}_i + \boldsymbol{\varepsilon}(k_i,t), \tag{18}$$

in which $\boldsymbol{\varepsilon}(k_i,t)$ is an s-dimensional White Noise with independent components, $\mathbf{\Phi}_i(L) = \mathbf{\Phi}_{i1}L + \mathbf{\Phi}_{i2}L^2 + \cdots + \mathbf{\Phi}_{ip_i}L^{p_i}$, with each $\mathbf{\Phi}_{ij}$ $(j = 1, 2, \ldots, p_i)$ being an $s \times s$ matrix of coefficients, p_i the order of the model of geosensor i and $\mathbf{c}_i \in \mathbb{R}^{s\times 1}$ a vector of intercepts (with $s = 6$). For each geosensor, matrices $\mathbf{\Phi}_{ij}$ are supposed to satisfy a stability condition ([28], pg. 70), in order to prevent the set of simultaneous difference equations in Formula 17 from having unbounded non-stationary solutions.

Unfortunately, when a multivariate time series includes a subset of strongly redundant variables, both least squares (LS) estimation of a linear time series model and forecasting with the estimated model can become highly problematic. To show that this statement holds true, we need to introduce a little more notation. Let $\mathbf{B}_i = (\mathbf{c}_i, \mathbf{\Phi}_{i1}, \mathbf{\Phi}_{i2}, \ldots, \mathbf{\Phi}_{i2}, \ldots, \mathbf{\Phi}_{ip_i})$ and $\boldsymbol{\beta}_i = \mathrm{vec}(\mathbf{B}_i)$. Suppose also that p_i presample values $\mathbf{z}(k_i, -p_i+1), \ldots, \mathbf{z}(k_i, 0)$ are available (in practice, the first p_i values of the training set are used as presample values). We now define $Z_i = (Z_{i0}, \ldots, Z_{i,T-1})$, where for $t = 0, 1, \ldots, T-1$:

$$Z_{it} = (1 \ \vdots \ \mathbf{z}(k_i,t)^\top \ \vdots \ \ldots \ \vdots \ \mathbf{z}(k_i, t-p_i+1)^\top)^\top.$$

We can now formulate a technical result which ensures that under mild conditions the LS estimator $\hat{\boldsymbol{\beta}}_i$ converges at rate $\sqrt{T}$ in distribution to a limiting multivariate Gaussian distribution, that is [28]:

$$\sqrt{T}\left(\hat{\boldsymbol{\beta}}_i - \boldsymbol{\beta}_i\right) \xrightarrow{d} \mathcal{N}\left(0, \Gamma_i^{-1} \otimes \Sigma_{\boldsymbol{\varepsilon}_i}\right), \tag{19}$$

where $\Sigma_{\boldsymbol{\varepsilon}_i}$ is the covariance matrix of the White Noise $\boldsymbol{\varepsilon}(k_i,t)$ (see [28], p. 73, for an explicit expression of matrix Γ_i). In order to assess the asymptotic covariance matrix of the LS estimator in finite samples, we need to estimate matrices Γ_i and $\Sigma_{\boldsymbol{\varepsilon}_i}$. A consistent estimator of Γ_i is [28]:

$$\hat{\Gamma}_i^{-1} = T\left(Z_i Z_i^\top\right)^{-1}. \tag{20}$$

If at least any two series of the multivariate system are perfectly redundant, then $Z_i Z_i^\top$ is singular. Moreover, if at least any two series are strongly redundant, then the diagonal entries of $Z_i Z_i^\top$ will be very large. In the latter case, the parameter estimator $\hat{\boldsymbol{\beta}}_i$ will be asymptotically correct from a theoretical point of view, but highly imprecise in practice. This result has a direct impact on forecasting with the estimated model, in the sense that the forecast uncertainty, implied by parameter estimation, can be neglected only asymptotically. In finite samples, precise forecasts require precise estimators, and if this requirement is not satisfied, the estimated model will suffer from a highly erratic forecast [28]. Irrespective of the particular dataset, we actually found imprecise forecasts from the estimated model (see Formula 18) with input variables (see Formula 17).

In view of the above discussion, after generating multiple variables per geosensor, a natural post-processing step consists of converting the additional data into uncorrelated principal components using a standard orthogonal transformation. An example of the effect of this transformation can be seen on the right side of Figure 3. The principal components are contemporaneously uncorrelated with each other and, on average, are weakly correlated with the target variable $\tilde{y}(k,t)$. This set of almost orthogonal components may have nonzero lagged cross-correlations, which can be exploited by the simultaneous linear model (18) to forecast the target variable, without over-inflating the asymptotic variance of parameter estimates [31].

6. A partially orthogonalized model VAR

The actual learning process considers, for each geosensor $k_i \in K$, a multivariate time series system defined as follows:

$$\mathbf{y}(k_i,t) = \left(y^{(1)}(k_i,t), y^{(2)}(k_i,t), \ldots, y^{(s')}(k_i,t)\right) = \left(\tilde{y}(k_i,t), \mathsf{PC}_1(k_i,t), \ldots, \mathsf{PC}_{s'-1}(k_i,t)\right), \tag{21}$$

where $\mathsf{PC}_1(k_i,t), \ldots, \mathsf{PC}_{s'-1}(k_i,t)$ are the first $s'-1$ principal component scores of the additional time series synthesized over the cluster aggregating k_i, with $s' \leq s$. In particular, only the first $s'-1$ principal components, which explain at least 95% of the original series, are retained. What is the motivation behind this choice? As explained in Subsection 5.1, the principal components can be interpreted as latent orthogonal factors useful to improve the forecasting of the target series. However, given the strong contemporaneous correlation existing between $\mu^{\mathcal{C}(k)}(t)$ and $\mu^{k,\mathcal{C}(k)}(t)$, the first four principal components are expected to explain a very high proportion of the total variability exhibited by the original series. For example, the right panel of Figure 2 shows that the time series of scores associated with the smallest eigenvalue, $\mathsf{PC}_5(k,t)$, is noisy and unlikely to contain useful information. It can most likely be interpreted as a small irregular

component, containing the variation uncommon to $\mu^{\mathcal{C}(k)}(t)$ and $\mu^{k,\mathcal{C}(k)}(t)$, which has not been captured by $\mathsf{PC}_1(k,t)$. For these reasons, despite the small scale of the involved variables, we apply a noise reduction based on the choice of the principal components that account for a given quota of the total variability, thus shrinking the smallest eigenvalues toward zero. As we empirically investigate in Subsection 8.5, modeling with denoised input variables generally improves the forecasting accuracy of the proposed model. Therefore, we now learn the following VAR model of the partially orthogonalized multiple time series in (21):

$$(\mathbf{I} - \mathbf{\Phi}_i(L))\mathbf{y}(k_i,t)^\top = \mathbf{c}_i + \boldsymbol{\varepsilon}(k_i,t), \tag{22}$$

and use it to forecast the target series $y^{(1)}(k_i,t) \equiv \tilde{y}(k_i,t)$ up to N-step ahead. We observe that, in this way, every multivariate system includes $s' \leq 6$ (instead of $s = 6$ as before) time series, for each geosensor in K.

We also note that it is immaterial whether raw $y(k_i,t)$ or normalized $\tilde{y}(k_i,t)$ are entered into the model and subsequently forecasted. We would rather see normalized series being used, thus making it possible to safely average scale-dependent forecast accuracy measures across geosensors, as well as to compare these average accuracies across multiple datasets (see Section 8 for details on the experimental setup).

The selection of the order p_i is based on the minimization of the determinant of the one-step ahead forecast mean square error matrix. This corresponds to the final prediction error criterion ($\mathsf{FPE}(p_i)$, expressed as a function of the number of lagged differences p_i). Details on the FPE criterion, the ordinary least squares estimation of model coefficients and the form of estimated optimal linear predictors can be found in [28].

6.1. Some inherent limitations

Periodic change patterns. The presence of a stable periodic pattern of changes in the target time series, repeated over m time periods, can be accommodated by adding m seasonal dummy variables [35] to the VAR model so that, at any time period t, one of the seasonal dummies equals 1, while the others equal 0 (plus some identifiability constraints, see [28] for details).

Drifting data. Under certain conditions, a stationary VAR model may be unsuitable after differencing a system of drifting time series, and even principal component analysis is often unsuited to remove non-stationarity. When we analyze each time series separately, as a univariate time series, we can easily identify the type of non-stationarity and adopt a suitable transformation of the training data to remove it. For example, this procedure is an integral part of auto.ARIMA, that is an algorithm defined for univariate time series data. In auto.ARIMA the original time series are differentiated for an appropriate number of times, until a test for the presence of unit roots ceases to provide statistically significant signals and the transformed data can finally be considered (at least approximately) stationary. However, in multivariate cases, when we simultaneously analyze a set of time series, we will deal with various additional issues. A multivariate system with a long-run equilibrium relationship cannot drift too far from this equilibrium. These series are *cointegrated*, as they share common stochastic trends. In these cases, a VAR model on the (first or second)

differences is known to be inconsistent and we must resort to vector error correction models (VECM) [30]. These models have the positive characteristic that the deviation of the current state from its long-run relationship is seen as a part of the short-run dynamics. In any case, applying a VECM model is often a difficult process, which makes it unsuitable for automatic implementation. For example, determining the number of co-integration relationships requires a careful application of some non-standard likelihood ratio tests [20]. As a result of these difficulties, we disregard verifying whether non-standard and non-stationary dynamic patterns are present and eventually incorporating them into the model. It would appear that, under these circumstances, the proposed algorithm should be considered no more than a heuristic. This key issue will be discussed further in Section 8.5.

7. Learning complexity

Let us consider that: (1) dataset $\mathcal{D}(K, Y, T)$ comprises $|K|$ geophysical time series and each time series includes T time-stamped measures of a geophysical variable Y; (2) algorithm PAM is used to construct a spatio-temporal clustering model $\mathcal{P}(\mathcal{C})$ of $\mathcal{D}$ and an execution of PAM performs $\mathsf{N_{Iter}}$ iterations, at worst; number g_{est} of clusters constructed in $\mathcal{P}(\mathcal{C})$ may vary between $g_{\min}$ and $|K|-1$; g_{est} is determined by maximizing the average Silhouette index computed according to Formula 10; the dissimilarity matrix D is computed according to Formula 2; (3) algorithm PCA is used to remove the collinearity in a system of $s-1$ cluster-coupled spatio-temporal time series constructed for each geosensor ($s-1=5$, in this study); (4) algorithm VAR is used to construct a forecasting model of a multivariate system, which consists of a geophysical time series and the principal component representation of its cluster-coupled spatial time series. Based on these premises, the computational complexity of cVAR is computed by summing up the cost of performing spatio-temporal clustering, synthesizing spatio-temporal variables, computing the principal components of the systems of spatio-temporal variables and constructing the spatial multivariate forecasting models.

Spatio-temporal clustering phase. This takes dataset $\mathcal{D}$ as input, computes the spatio-temporal dissimilarity matrix D, determines the number of clusters g_{est} and returns the clustering model $\mathcal{P}(\mathcal{C})$ discovered from $\mathcal{D}$ with the number of clusters g_{est} and the dissimilarity matrix D. The execution of PAM is repeated by varying the number of constructed clusters g between $g_{\min}$ and $g_{\max}$, with $g_{\min} \geq 2$ and $g_{\max} \leq |K|-1$. At each execution of PAM, the average Silhouette index of the constructed clustering model is evaluated. The time cost of building D is $K + \frac{(|K|^2-|K|)(T+2)}{2}$, where K is the cost of scaling the time series, $(|K|^2-|K|)/2$ is the number of distinct pairs of time series in $\mathcal{D}$, while $T+2$ is the cost of computing the spatio-temporal distance between the scaled time series. Therefore, the time complexity of building D is $\mathrm{O}(|K|^2 T)$. The time complexity of executing PAM, in order to discover g clusters in $\mathcal{D}$, is $\mathrm{O}(\mathsf{N_{Iter}} g(|K|-g)^2)$.[1] The time

[1] In the implementation, we used the optimized version of PAM, where every medoid can be removed just once before the addition of all potential alternative medoids. Although this optimization produces a real reduction of the learning times (see [43], Section 8.4, pages 486-488), it does not change the asymptotic computation complexity of the clustering algorithm.

complexity of computing the average Silhouette index (see Formula 10) is $O(|K|^2)$. In short, the cost of this phase is $|K|^2T + \sum_{g=g_{\min}}^{g_{\max}} (\mathsf{N_{Iter}} g(|K|-g)^2 + |K|^2)$. By considering that $|K| - g < |K|$, this cost is $|K|^2T + \mathsf{N_{Iter}}|K|^2 \sum_{g=g_{\min}}^{g_{\max}} g + (g_{\max} - g_{\min} + 1)|K|^2$. Let us consider that $\sum_{g=g_{\min}}^{g_{\max}} g \leq \sum_{g=1}^{g_{\max}} g = \frac{g_{\max}(g_{\max}+1)}{2} \leq g_{\max}^2$ as $g_{\max} > 1$, while $g_{\max} - g_{\min} + 1 \leq g_{\max}$ as $g_{\min} > 1$. Hence, the cost of clustering is $O(|K|^2T + \mathsf{N_{Iter}} g_{\max}^2 |K|^2)$.

Spatio-temporal variable synthesis phase. This constructs one collection of spatially-coupled time series associated with every geophysical time series of $\mathcal{D}$. Spatially coupled time series are constructed by accounting for the spatio-temporal clustering model $\mathcal{P}(\mathcal{C})$ discovered from $\mathcal{D}$. In fact, they are defined on the basis of:

- the sets of mean variables $\mu^{\mathcal{P}(\mathcal{C})} (\stackrel{\mathsf{def}}{=} \{\mu^{\mathcal{C}(k)}(t) | \mathcal{C}(k) \in \mathcal{P}(\mathcal{C})\})$, standard deviation variables $\sigma^{\mathcal{P}(\mathcal{C})}$ $(\stackrel{\mathsf{def}}{=} \{\sigma^{\mathcal{C}(k)}(t) | \mathcal{C}(k) \in \mathcal{P}(\mathcal{C})\})$ and speed variables $\omega^{\mathcal{P}(\mathcal{C})}$ $(\stackrel{\mathsf{def}}{=} \{\omega^{\mathcal{C}(k)}(t) | \mathcal{C}(k) \in \mathcal{P}(\mathcal{C})\})$, which are constructed at the level of a single cluster $\mathcal{C}(k) \in \mathcal{P}(\mathcal{C})$, for all geosensors $k \in \mathcal{C}(k)$;
- the sets of weighted mean variables $\mu^{K,\mathcal{P}(\mathcal{C})}$ $(\stackrel{\mathsf{def}}{=} \{\mu^{k,\mathcal{C}(k)}(t) | k \in K$ and $\mathcal{C}(k) \in \mathcal{P}(\mathcal{C})$, such that $k \in \mathcal{C}(k)\})$ and weighted speed variables $\omega^{K,\mathcal{P}(\mathcal{C})}$ $(\stackrel{\mathsf{def}}{=} \{\omega^{k,\mathcal{C}(k)}(t) |$ with $k \in K$ and $\mathcal{C}(k) \in \mathcal{P}(\mathcal{C})$, such that $k \in \mathcal{C}(k)\})$, which are constructed at the level of a single time series $k \in K$.

The time cost of constructing $\mu^{\mathcal{P}(\mathcal{C})}$ is $O(|K|T)$. The time cost of constructing $\sigma^{\mathcal{P}(\mathcal{C})}$ is $O(|K|T)$. The time cost of constructing $\omega^{\mathcal{P}(\mathcal{C})}$ is $O(g_{\text{est}}T)$, where the speed values are computed from the mean values associated to $\mu^{\mathcal{P}(\mathcal{C})}$. The time cost of constructing $\mu^{K,\mathcal{P}(\mathcal{C})}$ is $O(|K|^2T)$, in the worst case, that is, one time series is compared to every other time series in $\mathcal{D}$ (i.e. two clusters are constructed: one cluster groups $|K|-1$ time series, the other cluster groups one time series). The time cost of constructing $\omega^{K,\mathcal{P}(\mathcal{C})}$ is $O(|K|T)$, where the weighted speed values are computed from the weighted mean values associated to $\mu^{K,\mathcal{P}(\mathcal{C})}$. Therefore, the time cost of populating the systems of spatio-temporal variables coupled to the geophysical time series is $3|K|T + g_{\text{est}}T + |K|^2T$. As the term $|K|^2T$ is asymptotically more complex than $3|K|T$ as $|K| \geq 3$, the time cost of this phase is $O(|K|^2T)$, in the worst case.

Spatio-temporal principal component construction phase. This inputs, for each geophysical time series, the system of its cluster-coupled spatial time series and outputs the principal components of this system. According to the theory reported in [16], the principal components of a system of $s-1$ variables, for which T samples are collected, can be determined by computing the Singular Value Decomposition (SVD) of a data matrix $T \times (s-1)$. This requires the covariance matrix computation and its eigenvalue decomposition. The cost of covariance matrix computation is $O(T(s-1)^2)$ [16], while the cost of its eigenvalue decomposition is $O((s-1)^3)$ [16]. By considering that, in general, $T \geq s-1$, the cost of determining the system of spatio-temporal principal components, for each geophysical time series, is $O(|K|T(s-1)^2)$. By considering that, in this study, we construct $s-1=5$ spatio-temporal variables, we can treat $s-1$ as a constant and drop it from the O expression. Therefore, the time complexity of this phase can be rewritten as $O(|K|T)$.

Spatial multivariate forecasting model construction phase. This inputs, for each geophysical time series, the system composed of this time series and the principal components of its $s-1$ cluster-coupled spatial time series. Let us consider that, in the worst case, the number of principal components is equal to $s-1$ (that is, when $s-1=s'-1$), so that the number of variables in the system is equal to s. For each geosensor k_i, a VAR model of this multivariate system is constructed with order p_i ($p_i \leq p_{\max}$). The coefficients of the VAR model are estimated by resorting to the OLS algorithm [28]. According to the theory reported in [28] (Section 3.2.1), the time cost of determining the coefficients of a VAR(p_i) model is $O(s^3p_i^3 + s^2p_i^2T)$. Let us consider that the actual order $p_{i-\text{est}}$ is automatically determined by varying p_i between 1 and $p_{\max}$ and choosing the value that minimizes the FPE criterion [28]. By accounting for this additional cost to determine $p_{i-\text{est}}$, the total time cost is $s^3\sum_{p_i=1}^{p_{\max}} p_i^3 + s^2T\sum_{p_i=1}^{p_{\max}} p_i^2$, that is, $s^3\frac{p_{\max}^2(p_{\max}+1)^2}{4} + s^2T\frac{p_{\max}(p_{\max}+1)(2p_{\max}+1)}{6}$ as $\sum_{p=1}^{p_{\max}} p^3 = \frac{p_{\max}^2(p_{\max}+1)^2}{2}$ and $\sum_{p=1}^{p_{\max}} p^2 = \frac{p_{\max}(p_{\max}+1)(2p_{\max}+1)}{6}$. Therefore, the cost of constructing a VAR model, for each geophysical time series, is $O(|K|(s^3p_{\max}^4 + s^2Tp_{\max}^3))$. By taking that into account, in this study, $s \leq 6$, we can treat s as a constant and drop it from the O expression. Therefore, the time complexity of this phase can be rewritten as $O(|K|(p_{\max}^4 + Tp_{\max}^3))$.

Global time cost. The global cost is the sum of the cost of the spatial clustering phase ($|K|^2T+\mathsf{N_{Iter}}g_{max}^2|K|^2$), the cost of the spatio-temporal variable synthesis phase ($|K|^2T$), the cost of the spatio-temporal variable dimensionality reduction phase ($|K|T$) and the cost of the spatial multivariate forecasting model construction phase ($|K|(p_{\max}^4 + Tp_{\max}^3)$). Therefore, the global cost is $O(\mathsf{N_{Iter}}g_{max}^2|K|^2 + |K|^2T + |K|p_{\max}^4 + |K|Tp_{\max}^3)$. In any case, it is $O(\mathsf{N_{Iter}}g_{\max}^2|K|^2 + |K|^2T + |K|Tp_{\max}^3)$ under the hypothesis that $T \geq p_{\max}$.

Final remarks. The time complexity analysis highlights that the time cost is quadratic in the network size (K) and linear in the time series length (T). The optimization of the efficiency of the algorithm is an important challenge to address, in order to scale the computation on large, data intensive geosensor networks. We note that the majority of the computation time is spent calculating the dissimilarity matrix, the clustering model and the forecasting models. This is confirmed by the analysis of the running times spent to complete these phases and reported in the empirical study (see Tables 2 and 6 in Section 8.5). By following the main stream of recent research in data mining and statistics [14], an effective means to speed up the learning process may be the use of big data technologies, in order to design the proposed algorithm. This is supported by recent studies, which show that implementations designed for specific parallel processing architectures (e.g. MapReduce) can contribute to scale the computation of various similarity measures [13, 46], as well as to improve the efficiency of clustering algorithms like PAM [51]. In addition, recent studies in statistics have started the investigation of efficient parallel versions of algorithms to learn univariate forecasting models (e.g. ARIMA, [26]). In any case, the investigation of how parallel processing architectures can be employed in the design of the presented algorithm is out of the current scope of this paper.

8. Empirical evaluation and discussion

8.1. Data sets

We consider twelve geosensor data sets measured at equally-spaced discrete time intervals across six geosensor networks. The collected data are characterized by a wide range of different dynamic behaviors (see Figure 4). We observe that this study does not attempt to perform any exploratory data analysis or subjective model identification based on autocorrelation properties. It relies upon automatic forecasting algorithms, in order to determine an appropriate time series model, estimate the parameters and compute the optimal linear forecasts.For each data set, the length of both the training and the testing phase (respectively T and N), as well as the unit of measurement (UM) and the sampling interval Δ are summarized in Table 1. A description of the networks and the measured variables is reported in the following.

TCEQ (Texas) geosensor network. This (http://www.tceq.state.tx.us/) was used to measure three variables, namely *Wind Speed*, *Air Temperature* and *Ozone Concentration*, through $|K| = 26$ geosensors, installed in Texas. For each variable, data were measured hourly from May 5 to 19, 2009. We used data from 2009-05-05 00:00 to 2009-05-18 23:00 for the training phase ($T = 336$) and data from 2009-05-19 00:00 to 2009-05-19 23:00 for the testing phase ($N = 24$).

MESA air pollution study geosensor network. This (http://depts.washington.edu/mesaair/) was used to measure the variable NO_x *Concentration*, through $|K| = 20$ geosensor, installed in California. Data were measured every two weeks from January 13, 1999 to September 23, 2009. We used data from January 13, 1999 to April 8, 2009 for the training phase ($T = 268$) and data from April 22, 2009 to September 23, 2009 for the testing phase ($N = 12$).

NREL eastern U.S. geosensor network. This (http://www.nrel.gov/) was used to measure the variable *Wind Speed*, through $|K| = 1326$ geosensors, installed in the eastern area of the United States. Data were measured at 80 meters above sea level, every 30 minutes, from January 1 to 4, 2004. We used data from 2004-01-01 00:00 to 2004-01-03 23:30 for the training phase ($T = 144$) and data from 2004-01-04 00:00 to 2004-01-04 23:30 for the testing phase ($N = 48$).

SAC geosensor network. This (http://climate.geog.udel.edu/~climate/) was used to measure the variable *Air Temperature*, through $|K| = 900$ geosensors, installed in South America. Each time series collected monthly-averaged measures from January 1999 to December 2010. We used data from January 1999 to December 2009 for the training phase ($T = 132$) and data from January 2010 to December 2010 for the testing phase ($N = 12$).

NSRDB geosensor network. This (http://www.ncdc.noaa.gov/data-access/land-based-station-data/land-based-datasets/solar-radiation) was used to measure three variables, namely *Global Solar Radiation*, *Direct Solar Radiation* and *Diffuse Solar Radiation*, through $|K| = 1071$ geosensors, installed across

Data title	Region	Phenomenon	UM	$\|K\|$	T	N	Δ
TCEQ	Texas, U.S	Wind Speed	mph	26	336	24	1 hour
		Air Temperature	F°	26	336	24	1 hour
		Ozone Concentration	ppb	26	336	24	1 hour
MESA	Los Angeles, U.S.	NO_x Concentration	ppb	20	268	12	2 weeks
NREL	Eastern U.S.	Wind Speed	m/s	1326	144	48	30 mins
SAC	South America	Air Temperature	C°	900	132	12	1 month
NREL/NSRDB	U.S.	Global Solar Radiation	W/m^{-2}	1071	77	24	1 hour
		Direct Solar Radiation	W/m^{-2}	1071	77	24	1 hour
		Diffuse Solar Radiation	W/m^{-2}	1071	77	24	1 hour
NCDC	U.S	Air Temperature	C°	72	93	12	1 month
		Precipitation	mm	72	93	12	1 month
		Solar Energy	MJ/m^{-2}	72	93	12	1 month

Table 1: Twelve sets of geosensor data, collected via six geosensor networks. Legend: UM $\equiv$ Unit of measurement; $|K| \equiv$ Number of geosensors; $T \equiv$ Length of the training phase; $N \equiv$ Length of the testing phase; $\Delta \equiv$ Sampling interval.

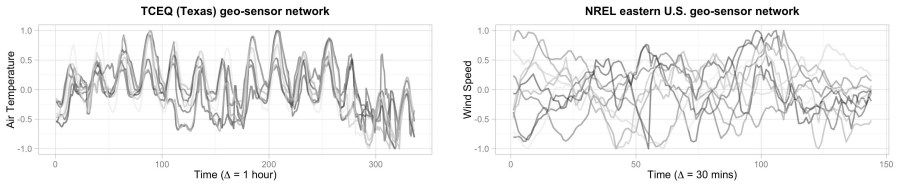

Figure 4: Training data of a subset of the normalized time series $\tilde{y}(k,t)$, measured across two geosensor networks. Top: variable *Air Temperature* (TCEQ geosensor network, $T = 336$) – Data are characterized by a well-defined daily periodicity, with varying amplitudes across the geosensors and non-stationary waveforms. Bottom: Variable *Wind Speed* (NREL eastern U.S. geosensor network, $T = 144$) – Data exhibit erratic patterns, resembling the dynamic behavior of a random walk without drift.

the United States. Data were hourly measured from July 17 to 21, 2004. We used data from 2004-07-17 01:00 to 2004-07-20 05:00 for the training phase ($T = 77$) and data from 2004-07-20 06:00 to 2004-07-21 05:00 for the testing phase ($N = 24$).

NCDC geosensor network. This (`http://www.ncdc.noaa.gov/data-access/land-based-station-data/land-based-datasets/climate-normals`) was used to measure three variables, namely *Air Temperature*, *Precipitation* and *Solar Energy*, through $|K| = 72$ geosensors installed in the United States. Data were measured monthly from August 2005 to April 2014. We used data from August 2005 to April 2013 for the training phase ($T = 93$) and data from May 2013 to April 2014 for the testing phase ($N = 12$).

8.2. Experimental setup

We have compared cVAR to various univariate forecasting techniques, such as auto.ARIMA [22], which neglects spatial autocorrelation, as well as to its spatial competitors sARIMA [38] and cARIMA [39], which

account for spatial correlation in a univariate time series setting. In addition, we have evaluated the performance of cVAR along the choice of the smoothing parameter α and the initial value ℓ_0. In particular, we have analyzed cVAR in combination with both the fast choice $\alpha = 0.5$, $\ell_0 = \mathsf{Tdiss}(k_i, k_j, 0) \stackrel{\mathsf{def}}{=} [\tilde{y}(k_i, 1) - \tilde{y}(k_j, 1)]^2$ and the automatic choice of global values of α and ℓ_0 computed with the minimization of the in-sample sum-of-squares error (see details in Section 4.2). We have also evaluated cVAR in combination with $\ell_0 = \mathsf{Tdiss}(k_i, k_j, 0)$ and the choice of local values $\alpha(t)$. This local choice is performed by adapting the local estimation method presented in [52], in order to determine a local value $\alpha(t)$ for each time point t. A few details on all these competing algorithms are reported in Section 8.4.

For each data set, the time series are split into training and testing data sets. The pre-processing parameters are determined on the training data (including the scaling parameter reported in Formula 8 for data normalization) and subsequently applied to the testing data. We run the compared algorithms on the training data and use the learned models to forecast the testing data. The learning phases are conducted on normalized training data $\tilde{y}(k_i, t)$. As mentioned in Section 6, this allows us to average scale-dependent forecast accuracy measures across geosensors, as well as to make sensible comparisons involving multiple data sets. In addition, normalized forecast data $\hat{\tilde{y}}(k_i, t)$ can be easily back-transformed to the original scale. If $\{\hat{\tilde{e}}(k_i, t)\}$ indicates a set of out-of-sample forecast errors, based on forecasts $\hat{\tilde{y}}^{(1)}(k_i, T+1), \ldots, \hat{\tilde{y}}^{(1)}(k_i, T+N)$ of the normalized target series, we evaluate forecast accuracy using a standard metric, such as [23]:

$$\text{Root Mean Square Error} = \mathsf{RMSE} = \sqrt{\text{mean}\left\{\hat{\tilde{e}}(k_i, t)^2\right\}}. \tag{23}$$

Before carrying out the model-building phase, the unknown period m_i of a geosensor k_i is estimated by resorting to a model-free procedure, called *Enright's periodogram*. For this purpose, we adopt a popular implementation [47], known as 'chi square' periodogram (see also [42]), that uses the χ^2 distribution. Whenever a stationary periodic component with period $m_i > 1$ is estimated, the current non-seasonal model is suitably extended to a seasonal one.

8.3. Some difficulties with extended input data computation

We note that a cluster $\mathcal{C}(k)$ may be discovered as a singleton cluster (i.e. it groups time series k only). Since, in this case, weighted average $\mu^{k,\mathcal{C}(k)}(t)$ (see Formula 14) diverges, it is replaced by the normalized series recorded in geosensor k. In addition, in the presence of a singleton cluster $\mathcal{C}(k)$, we have that $\mu^{\mathcal{C}(k)}(t) \equiv \mu^{k,\mathcal{C}(k)}(t) \equiv \tilde{y}(k, t)$, $\sigma^{\mathcal{C}(k)}(t) \equiv 0$ and $\omega^{\mathcal{C}(k)}(t) \equiv \omega^{k,\mathcal{C}(k)}(t)$. In this case, the additional multivariate time series in Formula 11 has rank 2 and, consequently, $s' - 1$ is equal to 2 in Formula 21.

Finally, if average value $\mu^{\mathcal{C}(k)}(t)$ is equal to 0 at time $t-1$, then speed $\omega^{\mathcal{C}(k)}(t)$ is infinite. We perform a correction of this variable, in order to replace infinite values with suitable ones. The easiest way to deal with this case is to consider each divergent velocity as missing and replace it with the median of the observed values for that variable. Although this strategy can lead to underestimating both the in-sample standard error and the out-of-sample mean square prediction error [15], divergent speeds are so rare that they cannot affect the forecasting results significantly. Even more occasionally, the same problem can occur for $\omega^{k,\mathcal{C}(k)}(t)$.

8.4. Competing algorithms

8.4.1. auto.ARIMA

This algorithm uses a multiplicative seasonal ARIMA$(p_i, d_i, q_i) \times (P_i, D_i, Q_i)_{m_i}$ process with seasonal period m_i to model the time evolution of a time series [4]. This specification can be compactly written as:

$$\phi_i(B)\Phi_i(B^{m_i})\nabla^{d_i}\nabla_{m_i}^{D_i}\tilde{y}(k_i,t) = c_i + \theta_i(B)\Theta_i(B^{m_i})\varepsilon(k_i,t), \tag{24}$$

where $\varepsilon(k_i,t)$ is a White Noise process, $\phi_i(z)$ and $\theta_i(z)$ are AR and MA polynomials of orders p_i and q_i respectively and $\Phi_i(z)$ and $\Theta_i(z)$ are seasonal polynomials of orders P_i and Q_i respectively (it is assumed that both polynomials have no roots inside the unit circle, $|z| < 1$) Furthermore, $\nabla\tilde{y}(k_i,t) = (1-B)\tilde{y}(k_i,t)$ is the lag-1 differencing operator, which is a special case of the more general lag-h differencing operator $\nabla_h\tilde{y}(k_i,t) = (1-B^h)\tilde{y}(k_i,t)$. In this study, we have set $D_i = 0$ for each geosensor k_i. This is equivalent to the assumption that any cyclical pattern, whenever present in the data, is stationary over time.

The estimation of seasonal ARIMA is analogous to that for ARMA. The first step is the model order identification, in order to determine parameters $(p_i, q_i, P_i, Q_i, d_i)$. For this purpose, Hyndman and Khandakar [22] propose a step-wise algorithm to transverse the model-space efficiently, according to few order constraints. These constraints are provided over (p_i, q_i, P_i, Q_i), in order to determine the model with the lowest corrected Akaike Information Criterion (AICc) value. For non-seasonal ARIMA models, the AICc can be written as [21]:

$$\mathsf{AICc}(i) = \mathsf{AIC}(i) + \frac{2(p_i+q_i+u_i+1)(p_i+q_i+u_i+2)}{T-p_i-q_i-u_i-2}, \tag{25}$$

where $u_i = 1$ if $c_i \neq 0$ and $u_i = 0$ if $c_i = 0$, $\mathsf{AIC}(i) = -2\log(\hat{L}_i) + 2(p_i+q_i+u_i+1)$ is the standard Akaike Information Criterion and $\hat{L}_i$ is the maximized likelihood of the data. Therefore, AICc equals AIC plus a penalty for extra-parameters. This discourages the selection of overly complex models. For seasonal models, the number $P_i + Q_i$ of seasonal parameters can be suitably taken into account in Formula 25.

A detailed description of the steps is reported in [21, 22]. Preliminarily, the number of differences d_i is determined using repeated applications of the KPSS second-order stationarity test [29]. Then, the general specification (see Formula 24) can be restricted to non-seasonal models, in which $m_i = 1$ for each geosensor and there is the following implied standard ARIMA model:

$$\varphi_i(B)\nabla^{d_i}\tilde{y}(k_i,t) = c_i + \vartheta_i(B)\varepsilon(k_i,t), \tag{26}$$

with $\varphi(B) = \Phi(B)\phi(B)$ and $\vartheta(B) = \theta(B)\Theta(B)$. Subsequently, the seasonal period m_i is estimated and the algorithm is re-run to select an appropriate seasonal model order. In both cases, the selected model is used to produce forecasts. We distinguish between these two model search strategies (involving either non-seasonal or seasonal models), using auto.ARIMA and auto.ARIMA$^{\pi}$ notations respectively.

8.4.2. sARIMA *and* cARIMA

Both sARIMA [38] and cARIMA [39] use a specialized spatial-aware version of auto.ARIMA. In sARIMA, the best model for geosensor k_i is selected according to a spatial-aware AICc, which is computed as the average of

the AICc values of geosensors located within a circle of a given radius. This allows us to account for spatial correlation without replacing real data with aggregated ones. Similarly, cARIMA uses a spatio-temporal clustering pattern $\mathcal{P}(\mathcal{C})$ computed for the geosensor data $\mathcal{D}(K, Y, T)$. It performs the model selection using a cluster-wise AICc value, which is computed as the average over the AICc values of geosensors grouped in cluster $\mathcal{C}(k)$. Once the model order is globally determined for the cluster $\mathcal{C}(k)$, the model coefficients are locally determined for each time series $\tilde{y}(k_i, t)$, where $k_i \in \mathcal{C}(k)$.

Let $\mathcal{N}(i)$ denote a circular neighborhood of fixed radius r of geosensor k_i. Then, the spatial-aware AICc, valid for the sARIMA algorithm, can be expressed as:

$$\mathsf{AICc}^{\star}(i) = \frac{\mathsf{AICc}(i) + \sum_{k_j \in \mathcal{N}(i),\, j \neq i} w(i,j)\mathsf{AICc}(j)}{1 + \sum_{k_j \in \mathcal{N}(i),\, j \neq i} w(i,j)}, \tag{27}$$

where $w(i, j)$ is the Gaussian kernel computed with the bandwidth equal to radius r of the circular neighborhood and the argument equal to the squared Euclidean distance d_{ij}^2 between geosensors k_i and k_j:

$$w(i,j) = \begin{cases} \exp\left\{\frac{1}{2r^2}\mathsf{Sdiss}(k_i,k_j)^2\right\} & \text{if } \mathsf{Sdiss}(k_i,k_j) \leq r \\ 0 & \text{otherwise} \end{cases}. \tag{28}$$

The model order is determined globally within neighborhood $\mathcal{N}(i)$, and then a local model with the same optimal number of parameters is fitted to the corresponding $\tilde{y}(i, t)$. It is possible after simple algebraic manipulations to express Formula 27 as:

$$\begin{aligned} \mathsf{AICc}^{\star}(i) &= \frac{\mathsf{AIC}(i) + \sum_{k_j \in \mathcal{N}(i),\, j \neq i} w(i,j)\mathsf{AIC}(j)}{1 + \sum_{k_j \in \mathcal{N}(i),\, j \neq i} w(i,j)} + \frac{2(p_{\mathcal{N}(i)} + q_{\mathcal{N}(i)} + u_{\mathcal{N}(i)} + 1)(p_{\mathcal{N}(i)} + q_{\mathcal{N}(i)} + u_{\mathcal{N}(i)} + 2)}{T - p_{\mathcal{N}(i)} - q_{\mathcal{N}(i)} - u_{\mathcal{N}(i)} - 2} \\ &= \text{Average AIC inside } \mathcal{N}(i) \text{ + common global extra-penalty imposed inside } \mathcal{N}(i). \end{aligned} \tag{29}$$

In a similar way, cARIMA determines the optimal model order cluster-wise, on the basis of the average AICc per cluster [39], which is constant inside the cluster $\mathcal{C}(i) \in \mathcal{P}(\mathcal{C})$, to which geosensor k_i belongs:

$$\mathsf{AICc}^{\star\star}(i) = \frac{1}{|\mathcal{C}(i)|} \sum_{k_j \in \mathcal{C}(i)} \mathsf{AIC}(j) + \frac{2(p_{\mathcal{C}(i)} + q_{\mathcal{C}(i)} + u_{\mathcal{C}(i)} + 1)(p_{\mathcal{C}(i)} + q_{\mathcal{C}(i)} + u_{\mathcal{C}(i)} + 2)}{T - p_{\mathcal{C}(i)} - q_{\mathcal{C}(i)} - u_{\mathcal{C}(i)} - 2}. \tag{30}$$

8.4.3. Local set-up of the smoothing parameter α

We adapt the local estimation method, recently presented in [52], in order to determine local values $\alpha(t)$. These values are then considered in the recursive computation of a smoothed version of the temporal dissimilarity $\mathsf{Tdiss}(k_i, k_j)$. In particular, let $\mathsf{W}(t) = \{\mathsf{W}(t, k_i, k_j)\}$ be the $|K| \times |K|$ local t-stamped dissimilarity matrix, with $\mathsf{W}(t, k_i, k_j) = [\tilde{y}(k_i, t) - \tilde{y}(k_j, t)]^2$. By considering the theory presented in [52], $\mathsf{W}(t)$ can be dealt with as an example of a non-stationary random process indexed by discrete time steps t, for which we assume that the following linear decomposition, $\mathsf{W}(t) = \Phi(t) + \mathsf{N}(t)$, holds for $t = 0, 1, 2, \ldots, T$.

In this decomposition, $\Phi(t)$ is an unknown deterministic matrix (namely true dissimilarity matrix) of unobserved states. This matrix changes over time to reflect long-term drifts in the differences, while $\mathsf{N}(t)$ is a zero mean noise matrix. A better noise-free estimate of the true dissimilarity matrix can be obtained

using the smoothed proximity matrix defined by $\widehat{\Phi}(t) = \alpha(t)\widehat{\Phi}(t-1) + (1-\alpha(t))\mathsf{W}(t)$, for $t = 1, 2, \ldots T$ with $\widehat{\Phi}(0) = \mathsf{W}(0) \stackrel{\mathsf{def}}{=} W(1)$, $\alpha(t)$ corresponding to the smoothing factor that controls the rate at which past dissimilarities are forgotten. Therefore, for all pairs k_i, k_j, $\hat{\Phi}(T, k_i, k_j)$ can be considered as a noise-free version of the temporal dissimilarity $\mathsf{Tdiss}(k_i, k_j) \stackrel{\mathsf{def}}{=} \mathsf{Tdiss}(k_i, k_j, T)$ that we defined in Formulae 4–6. As reported in [52], for fixed t, we can determine $\alpha(t)$ by minimizing the squared Frobenius norm of the difference $L(\alpha(t)) = \|\widehat{\Phi}(t) - \Phi(t)\|_F^2$. This corresponds to computing $\alpha(t)$ as follows (see details in [52]):

$$\alpha(t) = \frac{\sum_{i=1}^{|K|}\sum_{j=1}^{|K|} \mathsf{Var}(\mathsf{N}(t, k_i, k_j))}{\sum_{i=1}^{|K|}\sum_{j=1}^{|K|}\left[(\widehat{\Phi}(t-1, k_i, k_j) - \Phi(t, k_i, k_j))^2 + \mathsf{Var}(\mathsf{N}(t, k_i, k_j))\right]}. \tag{31}$$

Procedurally, we apply the setting described in [52] to compute Formula 31. We replace $\Phi(t, k_i, k_j)$ with the spatial sample mean of $\mathsf{W}(t)$ and $\mathsf{Var}(\mathsf{N}(t, k_i, k_j))$ with the spatial sample variance of $\mathsf{W}(t)$, for fixed t.

8.5. Results and Discussion

In this section, we illustrate the results of the empirical evaluation performed with the data sets described in Section 8.1. We have compared the algorithms previously described, namely cVAR, auto.ARIMA, sARIMA and cARIMA, using a code entirely written in R (version 3.3 as of 2016–05–03).[2]

We start this study by analyzing the computation time spent by cVAR performing the clustering phase and completing the synthesis of the cluster-based spatio-temporal variables. These computation times are collected in the following three cases: $\alpha = 0.5$, α globally estimated (α = est.) and α locally estimated (α = local). The results, reported in Table 2, are organized as follows:

1. $\mathsf{Time}_1 \equiv$ CPU time spent (in seconds) computing the spatio-temporal dissimilarity matrix and complete clustering;
2. $\mathsf{Time}_2 \equiv$ CPU time spent (in seconds) synthesizing the cluster-based spatio-temporal variables;
3. $\mathsf{Time}_3 \equiv$ CPU time spent (in seconds) performing the Singular Value Decomposition (SVD) and determining the principal component scores of the spatio-temporal variables.

In correspondence with the time complexity analysis reported in Section 7, the results reported in Table 2 show that the majority of the computation time is spent determining the spatio-temporal dissimilarity matrix D, estimatng the number of groups g_{est} and determining the clusters of geosensors. The number of clusters has been estimated by maximizing the average Silhouette index, with g ranging between $g_{\min} = 2$ and $g_{\max} = 20$.[3] In this evaluation, the order of the computation times is, affected by the programming environment used. In particular, the performances are influenced by the dynamism of the language R and

[2] R is a functional high-level programming language for statistical computing and graphics. This language is widely used among statisticians and data miners for developing statistical software and data analysis [41].

[3] For dataset MESA $g_{\max} = 19$. This threshold is decided by considering that MESA includes $|K| = 20$ geosensors.

Data title	Phenomenon	$\|K\|$	Time_1	Time_2	Time_3	Time_1	Time_2	Time_3	$\mathsf{Time}_1^\star$	Time_2	Time_3
			$\alpha = 0.5$			$\alpha =$ est.			$\alpha =$ local		
TCEQ	Wind Speed	26	1.23	0.33	0.03	1.41	0.21	0.02	0.02	0.19	0.03
	Air Temperature	26	1.16	0.35	0.02	1.58	0.18	0.02	0.02	0.18	0.02
	Ozone Concentration	26	1.29	0.33	0.02	1.68	0.21	0.02	0.01	0.19	0.02
MESA	NO_x Concentration	20	0.71	0.15	0.02	0.87	0.15	0.01	0.02	0.17	0.02
NREL	Wind Speed	1326	2830.34	50.05	0.62	3346.66	48.33	0.61	38.68	50.52	0.58
SAC	Air Temperature	900	1298.63	19.99	0.40	1538.15	20.34	0.40	10.81	19.18	0.40
NREL/NSRDB	Global Solar Radiation	1071	1796.13	16.91	0.40	1947.90	16.93	0.46	25.51	13.95	0.39
	Direct Solar Radiation	1071	1788.12	17.19	0.42	1894.03	16.83	0.44	26.45	13.38	0.45
	Diffuse Solar Radiation	1071	1784.35	16.83	0.47	1888.24	17.40	0.44	27.40	16.99	0.39
NCDC	Air Temperature	72	8.12	0.14	0.03	9.37	0.12	0.03	0.04	0.16	0.03
	Precipation	72	8.45	0.12	0.03	9.35	0.11	0.03	0.05	0.58	0.04
	Solar Energy	72	8.53	0.12	0.03	9.43	0.15	0.03	0.05	0.20	0.03

Table 2: $\mathsf{Time}_1 \equiv$ CPU time (in seconds) spent computing the spatio-temporal dissimilarity matrix and complete clustering; $\mathsf{Time}_2 \equiv$ CPU time (in seconds) spent synthesizing the cluster-based spatio-temporal variables (see Section 5); $\mathsf{Time}_3 \equiv$ CPU time (in seconds) spent performing the Singular Value Decomposition (SVD) and determining the principal component scores of the spatio-temporal variables. The results describe the performances with $\alpha = 0.5$, α globally estimated ($\alpha =$ est.) and α locally estimated ($\alpha =$ local), respectively. We observe that: $\mathsf{Time}_1^\star$ measures the CPU time (in seconds) spent performing the clustering phase only. The local set-up of the smoothing parameter α has been performed using a code written in Java. Hence, the time spent computing the noise-free temporal dissimilarity is not directly comparable.

the mechanism of lazy evaluation of the functions [50]. On the other hand, the computation time of the SVD can be considered almost irrelevant in this analysis. In fact, in this case R only acts as a wrapper that manages the results obtained using low-level LAPACK libraries written in Fortran. In any case, these considerations may become less relevant when accounting for the fact that the computation described in Table 2 is done only once and for all geosensors in the network.

We proceed in this study by analyzing the forecasting errors achieved for the testing sets. These results are shown in Table 3. In accordance with the setting introduced in Section 8.4, we distinguish all the model search strategies that we have discussed so far, according to whether the search space does not contain (resp.: contains) periodic components. For example, cVAR and cVAR^π denote, respectively, a search strategy based on a model not including (resp.: including) seasonal dummy variables. In any case, periodic components are dealt with properly according to the model involved (for example, cARIMA^π runs over the space of seasonal ARIMA models). If the periodic components are included in the model, their periodicity is estimated a-priori by resorting to the procedure adopted in $\mathsf{auto.ARIMA}^\pi$. Except for auto.ARIMA, the results reported in this study are all achieved with the periodic components included in the model search. This decision is motivated by the observation that the models computed by cVAR^π, sARIMA^π and cARIMA^π have systematically yielded more accurate forecasts than the models computed without the periodic components. Therefore, for simplicity, we have avoided reporting the results performed by cVAR, sARIMA and cARIMA.

In accordance with the introduced notation, let θ be a generic parameter that conditions the order of

		Average RMSE								
Data title	Phenomenon	auto.ARIMA	auto.ARIMA$^{\pi}$	sARIMA$^{\pi}$	cARIMA$^{\pi}$	cARIMA$^{\pi}$	cARIMA$^{\pi}$	cVAR$^{\pi}$	cVAR$^{\pi}$	cVAR$^{\pi}$
					$\alpha = 0.05$	$\alpha =$ est.	$\alpha =$ local	$\alpha = 0.05$	$\alpha =$ est.	$\alpha =$ local
TCEQ	Wind Speed	0.32	**0.31**	0.34	0.36	0.36	0.36	0.32	0.32	0.32
	Air Temperature	0.48	0.41	0.40	0.36	0.36	0.36	0.22	0.22	**0.21**
	Ozone Concentration	0.69	0.58	0.58	0.65	0.64	0.65	**0.53**	0.54	0.54
MESA	NO_x Concentration	0.21	**0.18**	**0.18**	0.20	0.21	0.21	0.27	0.27	0.27
NREL	Wind Speed	**0.39**	**0.39**	0.41	0.41	0.40	0.41	0.44	0.43	0.42
SAC	Air Temperature	0.20	0.21	0.16	0.20	0.20	0.20	**0.15**	**0.15**	**0.15**
NREL/NSRDB	Global Solar Radiation	0.34	0.26	0.35	0.42	0.35	0.62	**0.17**	**0.17**	**0.17**
	Direct Solar Radiation	0.51	0.45	0.52	0.55	0.58	0.55	**0.45**	**0.45**	**0.45**
	Diffuse Solar Radiation	0.47	0.43	0.48	0.45	0.47	0.46	**0.30**	**0.30**	**0.30**
NCDC	Air Temperature	0.19	0.24	0.19	0.16	0.21	0.28	**0.12**	**0.12**	**0.12**
	Precipation	**0.26**	**0.26**	0.27	**0.26**	**0.26**	**0.26**	0.28	0.29	0.27
	Solar Energy	0.19	0.22	0.16	0.15	0.23	0.19	0.37	0.14	**0.13**
Overall Mean		0.35	0.33	0.34	0.35	0.36	0.38	0.30	**0.28**	**0.28**
Overall Median		0.33	0.29	0.34	0.36	0.36	0.36	0.29	0.28	**0.27**

Table 3: Twelve data sets collected via six geosensor networks (see details in Section 8.1). Average RMSE ≡ Forecasting root mean-square errors averaged per geosensor. The lowest errors are in bold.

the considered model. Then, $\theta_{\max}$ denotes the maximum achievable order, so that the model search phase explores candidate models having order θ ranging between 1 and $\theta_{\max}$. For example, $p_{\max}$ may denote the maximum order of the auto-regressive part of an ARIMA model, as well as the maximum order of a VAR model. In the experimental setting considered in this study, the following assignments have been used:[4]

- auto.ARIMA: $p_{\max} = 5$, $q_{\max} = 5$, $d_{\max} = 2$;
- auto.ARIMA$^{\pi}$: $p_{\max} = 5$, $q_{\max} = 5$, $d_{\max} = 2$, $P_{\max} = 2$, $Q_{\max} = 2$;
- sARIMA$^{\pi}$: $p_{\max} = 5$, $q_{\max} = 5$, $d_{\max} = 2$, $P_{\max} = 2$, $Q_{\max} = 2$. The radius of the Gaussian kernel (28) is the maximum Euclidean distance between each pair of nearest geosensors, that is, $r = \max_{i \in K} \min_{j \in K - \{i\}} \mathsf{Sdiss}(k_i, k_j)$;
- cARIMA$^{\pi}$: $p_{\max} = 5$, $q_{\max} = 5$, $d_{\max} = 2$, $P_{\max} = 2$, $Q_{\max} = 2$;
- cVAR$^{\pi}$: $p_{\max} = 5$ (periodic components are introduced via centered dummy variables).

We can observe that cVAR$^{\pi}$ yields the lowest overall mean RMSEs (overall mean = 0.28 for $\alpha =$ est., overall mean = 0.28 for $\alpha =$ local; see Table 3 and the comparison of box plot distributions in Figure 5). We have also investigated the effectiveness of our idea of learning the forecasting model after retaining the principal components that explain only 95% of the constructed cluster-coupled spatio-temporal time series. In all considered data sets, the forecasting accuracy achieved by the model with de-noised inputs

[4] Parameter assignments are consistent with the default set-ups used in R to determine both ARIMA and VAR models.

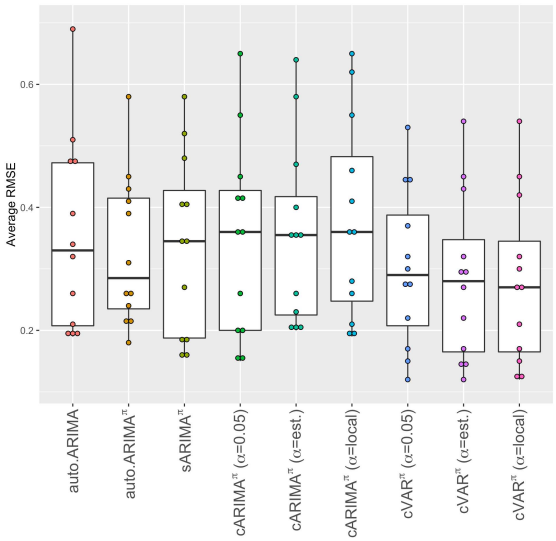

Figure 5: Box plot distributions of the average forecasting accuracies of the algorithms compared in this study. The line that divides the box into two parts indicates the 2nd quartile (overall median RMSE). The overall mean RMSEs are shown in Table 3.

($s' < s$) is higher than the accuracy achieved by the 'full' model ($s' = s = 6$, no de-noising threshold). In particular, the greatest accuracy gain is achieved on data set NREL/Wind Speed, where the following forecasting accuracies have been produced without de-noising the principal components (i.e. with $s' = s = 6$ in Formula 21): RMSE $= 0.58$ with $\alpha = 0.5$; RMSE $= 0.46$ with $\alpha =$ est.; RMSE $= 0.43$ with $\alpha =$ local.

The statistical significance of the results shown in Table 3 is assessed by using a suitable statistical method, in order to test the differences between more than two related samples. We use the Quade non-parametric test, that is an extension of the Wilcoxon signed-ranks test, in order to overcome the limitations of traditional repeated-measure parametric analysis of variance (ANOVA; [9]). Although the Quade test is less well-known than the more traditional Friedman test, it has more assumptions than the latter, so it is also more powerful [34]. The null hypothesis H_0 is that the compared algorithms have identical forecasting errors. The hypothesis H_a is that at least one algorithm is different from at least one other algorithm. In this study, the null hypothesis is rejected with $p < 0.01$ (test statistic: $Q = 4.3187$, with 8 and 88 degrees of freedom).

Since the null hypothesis is rejected, we proceed with post-hoc tests for the pairwise comparisons of the forecasting errors [11]. A post-hoc test following a significant Quade test is described in the literature [36]. The results are shown in Table 4, in which p-values are adjusted with the Benjamini-Hockberg procedure, to control the Type I family-wise error rate for all the $[\ell(\ell-1)]/2$ hypothesis (with $\ell = 9$) at level 0.05 [6]. The reported p-values show that cVAR$^{\pi}$, with either globally or locally estimated α, is significantly more accurate than all the other algorithms. Similarly, cVAR$^{\pi}$ with $\alpha = 0.5$ is significantly more accurate than other algorithms, except for auto.ARIMA$^{\pi}$ and cARIMA$^{\pi}$. However, the latter result can be caused by the

	auto.ARIMA	auto.ARIMA$^{\pi}$	sARIMA$^{\pi}$	cARIMA$^{\pi}$ ($\alpha = 0.5$)	cARIMA$^{\pi}$ ($\alpha =$ est.)	cARIMA$^{\pi}$ ($\alpha =$ local)	cVAR$^{\pi}$ ($\alpha = 0.5$)	cVAR$^{\pi}$ ($\alpha =$ est.)
auto.ARIMA$^{\pi}$	0.34							
sARIMA$^{\pi}$	0.64	0.64						
cARIMA$^{\pi}$($\alpha = 0.5$)	0.55	0.73	0.85					
cARIMA$^{\pi}$($\alpha =$ est.)	0.73	0.19	0.42	0.33				
cARIMA$^{\pi}$($\alpha =$ local)	0.64	0.13	0.33	0.26	0.85			
cVAR$^{\pi}$($\alpha = 0.5$)	**0.01**	0.14	**0.04**	0.06	<**0.01**	<**0.01**		
cVAR$^{\pi}$($\alpha =$ est.)	<**0.01**	**0.02**	<**0.01**	**0.01**	<**0.01**	<**0.01**	0.48	
cVAR$^{\pi}$($\alpha =$ local)	<**0.01**	<**0.01**	<**0.01**	<**0.01**	<**0.01**	<**0.01**	0.27	0.72

Table 4: The lower triangle of the matrix that contains the p-values of the pairwise comparisons of average forecasting accuracies of the nine algorithms compared in this paper. The reported p-values are adjusted with the Benjamini-Hockberg procedure, in order to control the Type I family-wise error rate (at level 0.05).

weak performance achieved with the dataset NCDC-Solar Energy. We will provide a detailed discussion of the results achieved with this dataset in the following. Finally, this statistical analysis shows that auto.ARIMA, auto.ARIMA$^{\pi}$, sARIMA$^{\pi}$ and cARIMA$^{\pi}$, whatever α is estimated, can be considered substantially equivalent.

As shown by the empirical results, the performance of cVAR degenerates if the phenomenon analyzed has an erratic time course, that is reminiscent of the dynamic behavior of a random walk (see, for example, the trend of wind speed measurements, plotted in Figure 4 for a subset of geosensors). In this case, the univariate algorithms generally achieve the highest performances, because they can account for the non-stationarity of the phenomenon (see the discussion in Section 6.1). Conversely, the proposed multivariate algorithm achieves the highest performances when the analyzed phenomenon is characterized by a number of patterns, which can be different to fit non-stationary data, but are globally stable over time. An example is the temporal pattern exhibited by the temperature measurements plotted in Figure 4 for a subset of geosensors. In fact, we can note that this field exhibits a set of non-stationary waveforms with variable amplitude. However, the spatio-temporal dissimilarity computed between these series is able to capture a global pattern, due to the mechanism implied in the calculation of Tdiss(k_i, k_j) (see Formulae 4-6). In this case, the clustering algorithm is able to partition geosensors according to their overall pattern, in such a way that the intra-cluster data variability is significantly lower than the global variability.

In particular, when the clustering goal is achieved, the cluster-based variables are informative of the real autocorrelation property of the geosensor data. In contrast, when data exhibit an erratic behavior, the temporal part of the dissimilarity computation does not provide any substantial contribution to the measure of the spatio-temporal dissimilarity (see Formula 1). Thus, the discovered clusters can be a spurious aggregation of close geosensors, which are inappropriate representatives of the autocorrelation property. In this case, it is possible that the cluster-based variables, which are injected into the VAR model (see Formula 17) over-fit data rather than improve the accuracy of the forecasting model.

Further considerations concern parameter α. In fact, in the majority of data sets, the forecasting accuracy

Data title	Phenomenon	$\|K\|$	g_{est}	Sil	g_{est}	Sil	g_{est}	Sil	Rand	Rand_A	Rand	Rand_A	Rand	Rand_A
			$\alpha = 0.5$		$\alpha =$ est.		$\alpha =$ local		0.5–est.		0.5–local		est.–local	
TCEQ	Wind Speed	26	4	0.52	4	0.51	4	0.72	1.00	1.00	1.00	1.00	1.00	1.00
	Air Temperature	26	4	0.56	4	0.58	4	0.77	1.00	1.00	0.94	0.87	0.94	0.87
	Ozone Concentration	26	4	0.48	6	0.52	4	0.69	0.84	0.64	0.95	0.89	0.85	0.59
MESA	NO_x Concentration	20	4	0.38	5	0.34	5	0.53	0.95	1.00	0.95	1.00	1.00	1.00
NREL	Wind Speed	1326	4	0.37	4	0.37	3	0.54	0.94	0.84	0.74	0.84	0.75	0.40
SAC	Air Temperature	900	2	0.50	2	0.50	2	0.69	0.94	0.89	0.94	0.87	0.99	0.98
NREL/NSRDB	Global Solar Radiation	1071	3	0.40	3	0.40	3	0.58	0.99	0.97	1.00	0.99	0.98	0.96
	Direct Solar Radiation	1071	3	0.39	3	0.37	3	0.57	0.93	0.84	1.00	1.00	0.93	0.84
	Diffuse Solar Radiation	1071	3	0.40	3	0.39	3	0.58	0.95	0.90	1.00	1.00	0.96	0.90
NCDC	Air Temperature	72	6	0.42	2	0.39	2	0.57	0.63	0.27	0.61	0.22	0.82	0.64
	Precipation	72	2	0.36	2	0.34	12	0.53	0.97	0.94	0.57	0.57	0.57	0.57
	Solar Energy	72	2	0.39	3	0.38	20	0.56	0.85	0.95	0.54	0.62	0.65	0.65

Table 5: Twelve data sets collected via six geosensor networks (see details in Section 8.1). Legend: $|K| \equiv$ Number of geosensors; $g_{\text{est}} \equiv$ Estimated number of clusters; Sil $\equiv$ Average Silhouette width of the discovered cluster pattern (Formula 10); Rand: Rand index to compare the two different clustering outcomes ($\alpha = 0.5$ vs. $\alpha =$ est., $\alpha = 0.5$ vs. $\alpha =$ local, $\alpha =$ est. vs. $\alpha =$ local). The index gives a value between 0 and 1, where 1 means that the two clustering outcomes match identically [19]; Rand_A: Adjusted Rand index, taking into account that random chance will cause some objects to occupy the same clusters.

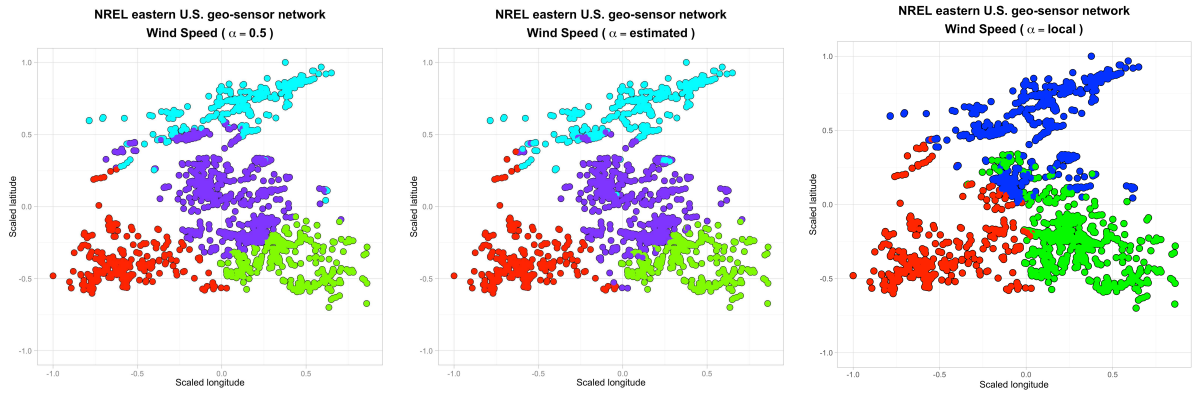

Figure 6: Clustering pattern discovered by cVAR^π, dataset NREL, field 'Wind Speed'. (Left) $\alpha = 0.5$. (Right) α is empirically estimated according to the procedure described in Section 4.2. The Rand indices and Adjusted Rand indices of the three clustering outcomes are respectively: $(0.94, 0.84)$ for $\alpha = 0.5$ vs. $\alpha =$ est.; $(0.74, 0.84)$ for $\alpha = 0.5$ vs. $\alpha =$ local; $(0.75, 0.40)$ for $\alpha =$ est. vs. $\alpha =$ local (see also Table 5).

is not greatly conditioned by the way α is determined. When we compare the results of clustering achieved by changing the way α is determined, we can observe that, if the processed data sets (e.g. NREL for which $|K| = 1326$) originate from networks with a high number of stations, densely distributed in the sensed area, the majority of differences in the cluster configurations are along the boundaries of the partitions obtained. An example of this phenomenon is shown in Figure 6, that plots clusters discovered from data set NREL with $\alpha = 0.5$, $\alpha =$ est. and $\alpha =$ local, respectively. Using both Rand Index and Adjusted Rand Index, we observe a moderate to high degree of similarity between the three clustering outcomes (see Table 5), except for NCDC-Air Temperature data set. In any case, the forecasting accuracy (measured in terms of average

RMSE, Table 3) does not change greatly, despite these differences observed in the cluster boundaries. The selection of α is critical only for data set NCDC-Solar Energy (average RMSE $= 0.37$ for $\alpha = 0.5$, average RMSE $= 0.14$ for $\alpha =$ est., average RMSE $= 0.13$ for $\alpha =$ local, see Table 3 again). However, this data set originates from a network characterized by a relatively low number of stations ($|K| = 72$).

This phenomenon can be explained by analyzing the curve of the average Silhouette index constructed during the clustering phase, by varying the candidate number of cluster g between $g_{\min} = 2$ and $g_{\max} = 20$, in order to output g_{est} (Figure 7). In particular, we observe that the average Silhouette index curve, constructed with $\alpha = 0.5$, achieves two close local peaks (local maximum points) in correspondence with $g = 2$ and $g = 6$. The final choice $g_{\text{est}} = 2$ is a numerical decision (as, in this point, the curve achieves the global maximum). However, the clustering pattern discovered with $\alpha = 0.5$ (two clusters, Figure 7) has a lower capability of constructing cluster-based variables that model the autocorrelation information appropriately, compared to the clustering pattern discovered with either locally estimated (three clusters, Figure 7) or globally estimated α (twenty clusters, with a competing local minimum at $g = 7$, Figure 7). In these cases, a valid solution may be to decide the value of α on the basis of the accuracy of the forecasts yielded on a validation set. In other words, after having discretized α in a suitable way in the range of $]0, 1[$, we will choose that value which will obtain the lowest average RMSE in the validation set. This strategy can be easily applied in datasets, like NCDC, which are generated by a small network. In fact, in these cases, the entire procedure for determining the final clustering pattern is not particularly demanding in terms of CPU time - see the value of Time_1 in Table 2.

We complete this study, by comparing the computation times of the forecasting phase of the compared algorithms. Table 6 collects the following measure:

- $\mathsf{Time}_4 \equiv$ CPU time spent (in seconds) determining the model order, estimating the final model and computing the optimal linear forecasts.

The amount Time_4 coincides with the total time of the calculation performed by auto.ARIMA, auto.ARIMA$^{\pi}$ and sARIMA$^{\pi}$. In contrast, the total time of the calculation performed by cARIMA$^{\pi}$ is $\mathsf{Time}_1 + \mathsf{Time}_4$, while the total time of the calculation performed by cVAR$^{\pi}$ is $\mathsf{Time}_1 + \mathsf{Time}_2 + \mathsf{Time}_3 + \mathsf{Time}_4$.

The results show that sARIMA$^{\pi}$ is particularly inefficient. Its final complexity depends on the number of geosensors in the network, as well as on the average number of geosensors that fall into every circular neighborhood. In the presence of spatially dense networks, such as NREL, the size of every neighborhood can be particularly large. In contrast, the computational complexity of auto.ARIMA and auto.ARIMA$^{\pi}$ depends largely on the time spent in the search for the optimal model. Although this search has been simplified, due to a suitable step-wise procedure for traversing the model space [22], it still must explore a two-dimensional space by taking into account the order of both the auto-regressive and the moving average part. In the case of cARIMA$^{\pi}$, this bottleneck is less relevant, as the search procedure is repeated for a limited number of times, that is, for each cluster of the geosensor partition induced by $\mathcal{P}(\mathcal{C})$. Similar considerations can be formulated for cVAR$^{\pi}$. In this case, the complexity of the forecasting phase is further simplified by the fact

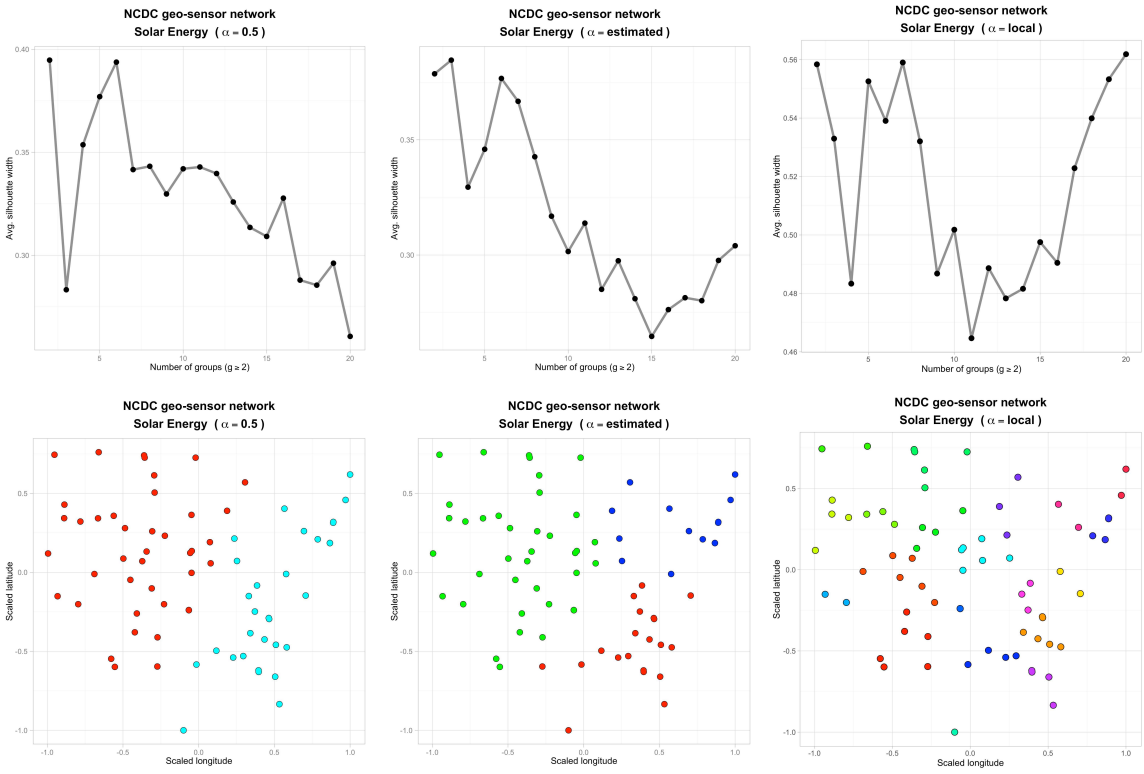

Figure 7: Average Silhouette index curve and clustering pattern discovered by cVAR$^\pi$, NREL dataset, 'Solar Energy' phenomenon. (Left) $\alpha = 0.5$. (Middle) globally estimated α, see Section 4.2. (Right) locally estimated α, see Section 8.4.3.

		Time$_4$								
Data title	Phenomenon	auto.ARIMA	auto.ARIMA$^\pi$	sARIMA$^\pi$	cARIMA$^\pi$	cARIMA$^\pi$	cARIMA$^\pi$	cVAR$^\pi$	cVAR$^\pi$	cVAR$^\pi$
					$\alpha = 0.05$	$\alpha =$ est.	$\alpha =$ local	$\alpha = 0.05$	$\alpha =$ est.	$\alpha =$ local
TCEQ	Wind Speed	10.82	40.95	70.12	11.46	10.15	10.13	1.45	1.45	1.92
	Air Temperature	16.11	75.90	25.52	4.00	3.83	4.24	1.75	1.47	1.46
	Ozone Concentration	10.87	90.91	42.64	4.42	10.23	4.21	1.57	1.42	2.30
MESA	NO_x Concentration	11.55	126.72	56.47	8.46	6.86	7.09	1.06	1.56	0.94
NREL	Wind Speed	280.40	718.30	7049.24	108.72	153.29	207.94	47.90	47.96	55.70
SAC	Air Temperature	288.70	528.25	322.17	42.05	38.48	38.30	28.57	28.09	29.15
NREL/NSRDB	Global Solar Radiation	296.28	610.59	380.75	179.93	169.66	283.62	47.71	50.88	50.16
	Direct Solar Radiation	221.44	447.69	241.17	91.86	99.45	88.06	69.01	83.57	54.73
	Diffuse Solar Radiation	244.02	370.44	282.74	145.88	103.36	110.86	36.31	35.57	36.58
NCDC	Air Temperature	24.36	39.72	687.53	26.20	244.81	31.12	1.92	2.88	1.90
	Precipation	8.99	13.93	219.14	25.04	23.37	13.21	2.02	2.18	2.15
	Solar Energy	20.95	38.77	1195.31	211.69	64.81	48.54	2.10	2.16	2.00

Table 6: Twelve data sets collected via six geosensor networks (see details in Section 8.1). Legend: Time$_4$ $\equiv$ CPU time (in seconds) spent determining the model order, estimating the final model and computing the optimal linear forecasts.

that the search space is performed per cluster and is one-dimensional (as it concerns only the auto-regressive part). This peculiarity makes cVAR$^\pi$ more efficient than its competitors. In fact, in some cases, the quantity

Time_4 is two orders of magnitude lower for cVAR^{π} than for its competitors.

9. Conclusion and future work

This paper describes a new algorithm, called cVAR, that combines multiple time series analysis and cluster analysis, in order to enhance the accuracy of forecasts. New variables are computed through spatio-temporal clusters and principal component analysis, in order to summarize the dynamic structure of spatial correlation over time. In this way, the knowledge of a time series measured by a specific geosensor is enriched with the spatial-aware time series calculated for new cluster-based variables. A stationary VAR model is used, in order to analyze the dynamic structure of this system of variables and apply this as a forecasting model of geosensor data. The empirical evaluation investigates the viability of the proposed algorithm in different real-world forecasting applications. We compared cVAR to the baseline univariate algorithm auto.ARIMA, which neglects spatial structure of data, as well as spatial-aware competitors sARIMA and cARIMA. The results show that cVAR yields, under well-defined conditions, more accurate forecasts than the univariate competitors. The proposed algorithm has clear advantages (in terms of predictive accuracy) when it is applied to geo-physical phenomena which exhibit patterns that are stable in time, possibly non-stationary, but characterized by a common evolution, once the geosensors used for sensing the data have been properly partitioned into clusters. In this case, the defined cluster-based spatio-temporal variables are effectively informative of the autocorrelation property across every given cluster of the determined partition. The intra-cluster variability is reduced by comparing it to the total, while the accuracy of the forecasts benefits from accounting for autocorrelation in the modeling phase. This behavior was observed for several cyclical natural phenomena described (in particular, temperature and solar radiation).

These considerations cannot be applied to phenomena (such as wind speed) which have a non-stationary form, due to their unpredictable random behavior. Globally, they behave as a multivariate random walk. In this case, it is very difficult to remove the non-stationarity in a suitable manner, while clusters may be spurious aggregations of geosensors (without any link to a global pattern in the clustering). Under these conditions, it may happen that the accuracy of the presented forecasting algorithm does not outperform that of standard univariate algorithms, which incorporate geosensor-wise mechanisms to remove the non-stationarity of sensed data. VECM models are linear parametric models that can be used to deal with non-stationary data in multivariate models. However, in our opinion, as discussed in Section 6.1, they are not particularly suitable for automatic forecasting of the majority of geophysical variables sensed through geosensor networks. An alternative is to relax the restrictions imposed by the parametric models, focusing attention on algorithms based on intrinsically non-parametric models. Following this premise, we plan to extend our analysis by considering non-parametric alternatives to stationary cVAR models, which go beyond the limitations suffered by such models, without losing the interpretability of the results. One promising alternative is Multivariate Singular Spectrum Analysis (MSSA, [17]), which is a non-parametric time series decomposition technique in a small set of easily interpretable components. The only restriction is that

MSSA can be applied in forecasting time series that, at least approximately, satisfy a certain recurrent linear formula. Apart from this, one of its advantages is that it can be used without making any assumption such as the stationarity and Gaussianity of the data (which are the pillars of inference in VAR models [28]).

As mentioned in Section 4.4, a further difficulty arises from the fact that a bias in determining the right number of clusters can occur if internal measures of clustering quality are used. As we are dealing with time series forecasting, a natural external criterion would consist in choosing the number of clusters to minimize the 1-step forecast error on the training set. This approach is likely to have a negative impact on the global computational complexity, although we intend to investigate it in future work because of its potentially positive impact on forecasting quality.

Another interesting research task will consist in quantifying the impact of using the same weight between spatial distance (Sdiss) and temporal dissimilarity (Tdiss) in defining diss in Formula 1. Also in this case, the relative weight of these two control factors can be adjusted so as to minimize the global forecasting error.

10. Acknowledgments and additional information

We would like to acknowledge the support of the European Commission through the project MAESTRA - Learning from Massive, Incompletely annotated, and Structured Data (Grant number ICT-2013-612944). The authors wish to thank Lynn Rudd for her help in reading the manuscript. We would like to extend our gratitude to the valuable reviews and contributions by the three anonymous referees. A suite of richly commented R scripts, implementing all the phases of the cVAR algorithm, is available for download at http://www.di.uniba.it/~appice/software/CVAR/index.htm.

References

[1] A. Appice, A. Ciampi, D. Malerba, Summarizing numeric spatial data streams by trend cluster discovery, Data Min. Knowl. Discov. 29 (1) (2015) 84–136.

[2] A. Appice, P. Guccione, D. Malerba, A. Ciampi, Dealing with temporal and spatial correlations to classify outliers in geophysical data streams, Inform. Sciences 285 (2014) 162–180.

[3] A. Appice, S. Pravilovic, D. Malerba, A. Lanza, Enhancing regression models with spatio-temporal indicator additions, in: M. Baldoni, C. Baroglio, G. Boella, R. Micalizio (eds.), AI*IA 2013: Advances in Artificial Intelligence, vol. 8249 of Lecture Notes in Computer Science, Springer International Publishing, 2013, pp. 433–444.

[4] D. Asteriou, S. Hall, ARIMA models and the Box-Jenkins methodology, in: Applied Econometrics (Second ed.), Palgrave MacMillan, 2011, pp. 265–286.

[5] S. M. Barbosa, M. E. Silva, M. J. Fernandes, Multivariate autoregressive modelling of sea level time series from TOPEX/Poseidon satellite altimetry, Nonlinear Proc. Geoph. 13 (2) (2006) 177–184.

[6] Y. Benjamini, Y. Hochberg, Controlling the false discovery rate: A practical and powerful approach to multiple testing, J. Roy. Stat. Soc. B 57 (1) (1995) 289–300.

[7] D. Birant, A. Kut, ST-DBSCAN: An algorithm for clustering spatial-temporal data, Data Knowl. Eng. 60 (1) (2007) 208–221.

[8] M.-Y. Cheng, J. Fan, V. Spokoiny, Dynamic nonparametric filtering with application to volatility estimation, in: Recent advances and trends in nonparametric statistics, Elsevier B. V., Amsterdam, 2003, pp. 315–333.

[9] W. Conover, Practical Nonparametric Statistics, Wiley series in probability and statistics: Applied probability and statistics, Wiley, 1999.

[10] X. De Luna, M. G. Genton, Predictive spatio-temporal models for spatially sparse environmental data, Stat. Sinica 15 (2) (2005) 547–568.

[11] J. Demšar, Statistical comparisons of classifiers over multiple data sets, J. Mach. Learn. Res. 7 (2006) 1–30.

[12] E. Egrioglu, U. Yolcu, C. Aladag, E. Bas, Recurrent multiplicative neuron model artificial neural network for non-linear time series forecasting, Neural Process. Lett. 41 (2) (2015) 249–258.

[13] T. Elsayed, J. J. Lin, D. W. Oard, Pairwise document similarity in large collections with mapreduce, in: ACL 2008, Proceedings of the 46$^{\text{th}}$ Annual Meeting of the Association for Computational Linguistics, June 15-20, 2008, Columbus, Ohio, USA, Short Papers, The Association for Computer Linguistics, 2008, pp. 265–268.

[14] J. Fan, F. Han, H. Liu, Challenges of big data analysis, NSR National Science Review 1 (2) (2014) 293–314.

[15] A. Gelman, J. Carlin, H. Stern, D. Dunson, A. Vehtari, D. Rubin, Bayesian Data Analysis, Third Edition, Chapman & Hall/CRC Texts in Statistical Science, Taylor & Francis, 2013.

[16] G. H. Golub, C. F. Van Loan, Matrix Computations (4th Edition), JHU Press, 2013.

[17] N. Golyandina, A. Zhigljavsky, Singular Spectrum Analysis for Time Series, SpringerBriefs in Statistics, Springer-Verlag Berlin Heidelberg, 2013.

[18] P. Guttorp, A. M. Schmidt, Covariance structure of spatial and spatiotemporal processes, Wiley Interdisciplinary Reviews: Computational Statistics 5 (4) (2013) 279–287.

[19] L. Hubert, P. Arabie, Comparing partitions, J. Classif. 2 (1) (1985) 193–218.

[20] K. Hubrich, H. Lütkepohl, P. Saikkonen, A review of systems cointegration tests, Econometric Rev. 20 (3) (2001) 247–318.

[21] R. J. Hyndman, G. Athanasopoulos, Forecasting: Principles and Practice, OTexts, 2013.

[22] R. J. Hyndman, Y. Khandakar, Automatic time series forecasting: The `forecast` package for R, J. Stat. Software 27 (3) (2008) 1–22.

[23] R. J. Hyndman, A. B. Koehler, Another look at measures of forecast accuracy, Int. J. Forecasting 22 (4) (2006) 679–688.

[24] Y. Kamarianakis, P. Prastacos, Space–time modeling of traffic flow, Comput. Geosci. 31 (2) (2005) 119–133.

[25] S. Kisilevich, F. Mansmann, M. Nanni, S. Rinzivillo, Spatio-temporal clustering, in: O. Maimon, L. Rokach (eds.), Data Mining and Knowledge Discovery Handbook, Springer US, 2010, pp. 855–874.

[26] L. Li, F. Noorian, D. J. M. Moss, P. H. W. Leong, Rolling window time series prediction using mapreduce, in: J. Joshi, E. Bertino, B. M. Thuraisingham, L. Liu (eds.), Proceedings of the 15$^{\text{th}}$ IEEE International Conference on Information Reuse and Integration, IRI 2014, Redwood City, CA, USA, August 13-15, 2014, IEEE, 2014, pp. 757–764.

[27] Q. Liu, M. Deng, J. Bi, W. Yang, A novel method for discovering spatio-temporal clusters of different sizes, shapes, and densities in the presence of noise, Int. J. Digit. Earth 7 (2) (2014) 138–157.

[28] H. Lütkepohl, New Introduction to Multiple Time Series Analysis, Springer-Verlag Berlin Heidelberg, 2005.

[29] H. Lütkepohl, M. Krätzig, Applied Time Series Econometrics, Themes in Modern Econometrics, Cambridge University Press, 2004.

[30] V. Martin, S. Hurn, D. Harris, Econometric Modelling with Time Series: Specification, Estimation and Testing, Cambridge University Press, 2012.

[31] D. S. Matteson, R. S. Tsay, Dynamic orthogonal components for multivariate time series, J. Am. Stat. Assoc. 106 (496) (2011) 1450–1463.

[32] J.-M. Montero-Lorenzo, G. Fernndez-Avils, J. Mondjar-Jimnez, M. Vargas-Vargas, A spatio-temporal geostatistical approach to predicting pollution levels: The case of mono-nitrogen oxides in madrid, Comput. Environ. Urban Syst. 37

(2013) 95 – 106.

[33] O. Ohashi, L. Torgo, Wind speed forecasting using spatio-temporal indicators, in: ECAI 2012 - 20th European Conference on Artificial Intelligence. Including Prestigious Applications of Artificial Intelligence (PAIS-2012) System Demonstrations Track, Montpellier, France, August 27-31 , 2012, 2012, pp. 975–980.

[34] D. Paulson, Applied Statistical Designs for the Researcher, Chapman & Hall/CRC Biostatistics Series, Taylor & Francis, 2003.

[35] B. Pfaff, Analysis of Integrated and Cointegrated Time Series with R, 2nd ed., Springer, New York, 2008, ISBN 0-387-27960-1.

[36] T. Pohlert, The Pairwise Multiple Comparison of Mean Ranks Package (PMCMR), R package (2014).

[37] D. Pokrajac, Z. Obradovic, Improved spatial-temporal forecasting through modelling of spatial residuals in recent history, in: Proceedings of the First SIAM International Conference on Data Mining, SDM 2001, Chicago, IL, USA, April 5-7, 2001, 2001, pp. 1–17.

[38] S. Pravilovic, A. Appice, D. Malerba, An Intelligent Technique for Forecasting Spatially Correlated Time Series, in: M. Baldoni, C. Baroglio, G. Boella, R. Micalizio (eds.), AI*IA 2013: Advances in Artificial Intelligence SE – 39, vol. 8249 of Lecture Notes in Computer Science, Springer International Publishing, 2013, pp. 457–468.

[39] S. Pravilovic, A. Appice, D. Malerba, Integrating cluster analysis to the ARIMA model for forecasting geosensor data, in: T. Andreasen, H. Christiansen, J.-C. Cubero, Z. Ra (eds.), Foundations of Intelligent Systems, vol. 8502 of Lecture Notes in Computer Science, Springer International Publishing, 2014, pp. 234–243.

[40] K. Qin, Y. Chen, Y. Zhan, F. Cheng, Spatial clustering considering spatio-temporal correlation, in: Geoinformatics, 2011 19th International Conference on, 2011, pp. 1–4.

[41] R Core Team, R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria (2015). https://www.R-project.org/

[42] R. Refinetti, G. C. Lissen, F. Halberg, Procedures for numerical analysis of circadian rhythms, Biol. Rhythm. Res. 38 (4) (2007) 275–325.

[43] A. Reynolds, G. Richards, B. de la Iglesia, V. Rayward-Smith, Clustering rules: A comparison of partitioning and hierarchical clustering algorithms, J. Math. Model. Algorithms 5 (4) (2006) 475–504.

[44] P. J. Rousseeuw, Silhouettes: A graphical aid to the interpretation and validation of cluster analysis, J. Comput. Appl. Math. 20 (1987) 53–65.

[45] P. Saengseedam, N. Kantanantha, Spatial time series forecasts based on Bayesian linear mixed models for rice yields in Thailand, in: Proceedings of the International MultiConference of Engineers and Computer Scientists, IMECS 2014, vol. II, Newswood Limited, 2014, pp. 1007–1012.

[46] S. Schelter, C. Boden, V. Markl, Scalable similarity-based neighborhood methods with MapReduce, in: P. Cunningham, N. J. Hurley, I. Guy, S. S. Anand (eds.), Sixth ACM Conference on Recommender Systems, RecSys '12, Dublin, Ireland, September 9-13, 2012, ACM, 2012, pp. 163–170.

[47] P. G. Sokolove, W. N. Bushell, The chi square periodogram: Its utility for analysis of circadian rhythms, J. Theor. Biol. 72 (1) (1978) 131–160.

[48] A. Struyf, M. Hubert, P. Rousseeuw, Clustering in an object-oriented environment, J. Stat. Software 1 (4) (1997) 1–30.

[49] R. S. Tsay, Multivariate Time Series Analysis. With R and Financial Applications, Wiley, 2014.

[50] H. Wickham, Advanced R (Chapman & Hall/CRC The R Series), 1st ed., Chapman and Hall/CRC, 2014.

[51] Y. Xianfeng, L. Liming, A New Data Mining Algorithm based on MapReduce and Hadoop, Int. J. Signal Process. Image Process. Pattern Recognit. 7 (2) (2014) 131–142.

[52] K. S. Xu, M. Kliger, A. O. Hero III, Adaptive evolutionary clustering, Data Min. Knowl. Discov. 28 (2) (2014) 304–336.

[53] E. Zivot, J. Wang, Modeling Financial Time Series with S-PLUS®, Springer New York, New York, NY, 2006.