UNIVERSIDAD NACIONAL DE INGENIERÍA

FACULTAD DE CIENCIAS

TESIS

"FORECAST MODELING OF SPATIO-TEMPORAL RASTER DATA USING PRINCIPAL COMPONENT ANALYSIS AND A NEURAL NETWORKS – WAVELET DECOMPOSITION MODEL"

PARA OPTAR EL GRADO ACADEMICO DE MAESTRO EN CIENCIAS
EN MATEMÁTICA APLICADA

PRESENTADA POR

CHRISTIAN AMAO SUXO

Asesor

Dr. OSWALDO JOSÉ VELÁSQUEZ CASTAÑÓN

Asesor externo

Dr. CARLOS ANTONIO ABANTO VALLE

LIMA – PERÚ
2018

ACKNOWLEDGEMENTS

I would first like to thank my external thesis advisor Ph.D. Carlos Antonio Abanto Valle of the school of statistics at the Federal University of Rio de Janeiro. Prof. Abanto was always attentive to my needs whenever I ran into a trouble spot or had a question about my research or writing. He consistently allowed this thesis to be my own work, but steered me in the right direction whenever he thought I needed it. I am forever grateful to you.

I would also like to thank the other expert proffesors who were involved in the realization of this research project. Thanks to Ph.D. Loretta Betzabe Rosa Gasco Campos and Dr. Alipio Francisco Ordóñez Mercado who gladly accepted to be my thesis reviewers. Without their passionate participation and helpful input, the thesis would not have been successfully conducted. A special thanks to Ph.D. Oswaldo Jose Velásquez Castañón and Ph.D. Eladio Teófilo Ocaña Anaya, professors of the master that wholeheartedly helped me with diverse burocratic processes. Without your support, the thesis would not have reached the sufficient formal requirements to be admitted.

Finally, I must express my very profound gratitude to my family and my unconditional friends for providing me with unfailing support and continuous encouragement throughout my master's studies and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you so much.

ABSTRACT

Forecast modeling of spatio-temporal raster data using principal component analysis and a neural networks - wavelet decomposition model

Christian Amao Suxo

Advisor: Oswaldo José Velásquez Castañón

Nowadays spatio-temporal forecasting has been drawing more and more attention from academic researchers and industrial practitioners for its great utility to plan and develop contingency measures against future adverse conditions. In this thesis, a methodology to forecast maps in spatio-temporal raster datasets is proposed. Following a *summarize-predict-and-rebuild* methodology, it 1) first suggests a reduction in the dimensionality of data using a principal component analysis, then 2) individual forecasts on the most significant components or eigenvectors are calculated using a neural networks - wavelet decomposition model. Finally, 3) a recursive algorithm, applied on the spectral inverse reconstruction of the individual forecasts, provides the final forecast maps.

The devised methodology led to three models according: the spatial principal component analysis (SPCA) model, the temporal principal component analysis (TPCA) model and the spatio-temporal principal component analysis (STPCA) model. In order to evaluate their forecasting accuracy, a simulation study was carried out by considering datasets with pure temporal, pure spatial and spatio-temporal variability. The results suggest using a TPCA (or SPCA) model when the temporal (or spatial) variability is predominant. For datasets with similar spatial and temporal variability information, the STPCA model provides the best forecast results. The research culminates with a real-world case study in monthly sea surface temperature anomalies of the Tropical Pacific Ocean.

Key-words: Raster dataset; forecast maps; forecasting accuracy; prediction.

RESUMEN

Modelado de pronóstico en datos espacio-temporales tipo raster usando un análisis de componentes principales y un modelo de redes neuronales con descomposición de ondículas

Autor: Christian Amao Suxo

Asesor: Oswaldo José Velásquez Castañón

En la actualidad, el pronóstico de datos espacio-temporales ha sido de especial interés para investigadores académicos y profesionales de la industria por su gran utilidad para planificar y desarrollar medidas de contingencia contra futuras condiciones adversas. En este trabajo se propone una metodología para el pronóstico de mapas tipo raster. Siguiendo una metodología de *resumir-predecir-y-reconstruir*, el método sugiere reducir la dimensionalidad de los datos usando un análisis de componentes principales para luego realizar pronósticos individuales sobre las componentes o autovectores más significativos. Finalmente, un algoritmo recursivo, aplicado sobre la reconstrucción inversa espectral de los pronósticos individuales, brinda el pronósticos final de los mapas.

La metodología propuesta da lugar a tres modelos: el modelo espacial de componentes principales (ECP), el modelo temporal de componentes principales (TCP) y el modelo espacio-temporal de componentes principales (ETCP). Con el fin de evaluar su capacidad de pronóstico, se realiza un estudio de simulación considerando datos con una estructura de variabilidad espacial pura, temporal pura y espacio-temporal. Los resultados sugieren usar un modelo TCP (o ECP) cuando la variabilidad temporal (o espacial) es predominante. Para datos con similar información en la variabilidad espacial y temporal, el modelo ETCP brinda los mejores resultados de pronóstico. El trabajo culmina con una aplicación real en datos mensuales de anomalías de temperatura superficial del mar del océano Pacífico Tropical.

Palabras clave: Datos raster; pronóstico de mapas; precisión de pronóstico, predicción.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Spatio-temporal forecasting models for raster datasets have gained widespread popularity in recent years due to its great applicability in different fields such as agriculture, economics and oceanography (Cressie and Majure, 1997; Pace et al., 1998). Indeed, spatio-temporal forecasting has been developing from individual spatial or temporal forecasting and gained vast attention recently for its promising performance in handling complex data, in which not only spatial but also temporal characteristic must be taken into account. This juxtaposition of space and time into the data has made the spatio-temporal forecasting a challenging task that requires models to take into account the interaction of these complex dinamycs. Furthermore, spatio-temporal datasets are often large and therefore require substantial computing resources to fit even simple models.

A brief review in literature shows that models for spatio-temporal data are usually built by integrating time series models with variogram-based models from spatial statistics. In the time series context, popular approaches include ARMA models (Box and Jenkins, 1976) for stationary data, and state-space models (Migon et al., 2005), which permit to model nonstationary components such as temporal trends and seasonality. In the spatial field, much of the literature goes around isotropic models (Cressie and Wikle, 2015) for spatial stationary datasets. Various methods also exist for nonstationary spatial processes, where the correlation depends on location as well as distance. For example, (Sampson and Guttorp, 1992) developed approaches based on transformations of the coordinate system.

The first spatio-temporal models often relied on the assumption of temporal stationarity. For example, the STARMA (Pfeifer and Jay Deutsch, 1980) and STARMAX (Stoffer, 1986) models were built by adding a spatial covariance structure to standard vector ARMA models. Later models proposed methods to deal with nonstationarity and spatial anisotropy. An example of this is the work of Guttorp et al. (1994), where with the use of the deformation technique of Sampson and Guttorp (1992) they could capture spatial anisotropy in a series of ozone readings. Wikle et al. (1998) used a hierarchical bayesian model with CAR priors for spatial effects for modeling monthly maximum atmospheric temperatures. Other approaches involving hierarchical bayesian models using CAR priors are also found in Cressie and Wikle (2015).

The main contribution of this thesis is a flexible, efficient and simple-to-manage methodology for spatio-temporal forecasting. Following this path, this work has as main objective to evaluate the forecast performance of a proposed *summarize-predict-and-rebuild* methodology. This methodology suggest, as first step, to reduce the dimensionality of data using a principal component analysis. This summary of information allows to work with the essential spatio-temporal variability. At a second step, individual forecasts on the most significant components or eigenvectors are found using a neural networks - wavelet decomposition model. This prediction part of the methodology allows to model only the significant information of a complex spatio-temporal dataset. Finally, the last step consists of applying a recoursive algorithm, on the spectral inverse reconstruction of the individual forecasts of the previous step, to provide the resulting forecast maps.

As a result of the *summarize-predict-and-rebuild* methodology, three models are induced: the temporal principal component analysis (TPCA) model, the spatial principal component analysis (SPCA) model and the spatio-temporal principal component analysis (STPCA) model. The TPCA and SPCA model differ in the second step of the devised methodology. While the TPCA model applies the forecasting model on the significant principal components, the SPCA model applies the model on the significant eigenvectors. For its side, the STPCA model behaves as an hybrid of the TPCA and SPCA models. In fact, it integrates the good forecasting results of both models in order to get an enhanced forecast accuracy.

The main advantage of the devised methodology is that it provides forecast maps without the need of geostatistics methods and, on the contrary, it works with methods that are more flexible and

simple to deal with. The proposed model works under the philosophy of understanding the whole knowing its parts. Indeed, the method works with the most relevant spatio-temporal variability of the raster dataset and seeks to model, in a detailed way, the different variability components in order to obtain an improved forecast.

This thesis is organized as follows. In chapter 2, a brief summary of the principal concepts and statistical techniques used in this work are explained. Concepts such as raster datasets, stationarity, spatial auto-covariance function, isotropy, and others are developed. Likewise, statistical methods such as the principal component analysis, the autoregressive neural network model and the discrete wavelet transform are also presented here. Chapter 3 displays the devised methodology of this thesis. An explicit mathematical and algorithmic description of the three proposed models are carried out here. Chapter 4 shows the results of a simulation study for checking the prediction and forecast performance of the proposed models. Chapter 5 develops a real-world case study in monthly sea surface temperature anomalies of the Tropical Pacific Ocean with the purpose to check the forecast and prediction performance of the methods with real datasets. Finally, chapter 6 gives the conclusions and future directions of this research.

Chapter 2

Preliminaries

2.1 Introduction

This chapter introduces some concepts and statistical techniques which will be used in the development of this thesis. Section 2.2 gives a description of raster datasets. Section 2.3 describes meaningful concepts to manage spatio-temporal raster datasets. Next, section 2.4 provides the mathematical development of the principal component analysis (PCA) technique and shows a brief literature review considering the use of the PCA to analyze spatio-temporal raster datasets. Thereby, section 2.5 introduces autoregressive neural network (AR-NN) models and section 2.6 deals with topics related to wavelet decomposition methods and their properties.

2.2 Raster datasets

In general, a raster dataset, or simply called raster, consists of a matrix of cells (or pixels) organized into rows and columns (or a grid) where each cell contains a value representing information. All cells in a raster set must be the same size, determining the *resolution*. The cells can be of any size, but they should be small as possible in order to accomplish a detailed analysis. In that way, a cell can represent a square kilometer, a square meter, or even a square centimeter. Figure 2.1 shows an example of how this process is done.

In raster datasets, each cell has a value. The cell values represent the phenomenon portrayed

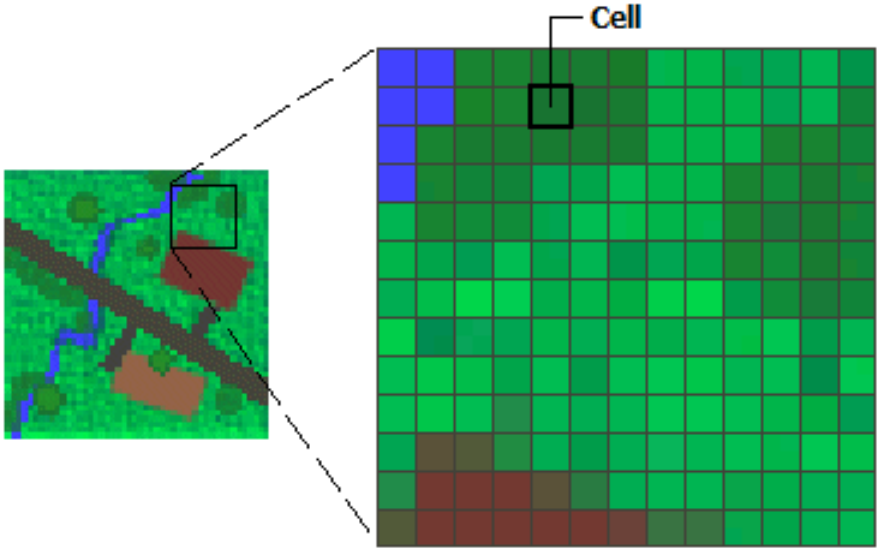

Figure 2.1: Representation of a scanned map with a raster dataset.

by the raster dataset such as a category, magnitude, height, or a spectral value. The category could be a land-use class such as grassland, forest or road. A magnitude might represent gravity, noise pollution or air temperature. Height (distance) could represent surface elevation above mean sea level, which can be used to derive slope, aspect and watershed properties. Spectral values are used in satellite imagery and aerial photography to represent light reflectance and color. For purposes of this work, the cell values will represent a magnitude.

Cell values can be either positive or negative, integer, or floating point. Integer values are used to represent categorical (discrete) data, and floating point values to represent continuous surfaces. Cells can also have a missing value to represent the absence of data. Figure 2.2 gives an idea what can represent the cell values. A cell value could represent the magnitude value in the center point of the cell or the entire area of the cell. For experimental and empirical analysis of this thesis the second option is considered.

The dimension of the cells must be chosen according to the analysis necessities. In fact, the cells can be as large or as small as needed to represent the surface conveyed by the raster dataset and the features within the surface. The size of the cell determines how coarse or fine the patterns or features will appear in the raster. The smaller the cell size, the smoother or more detailed the

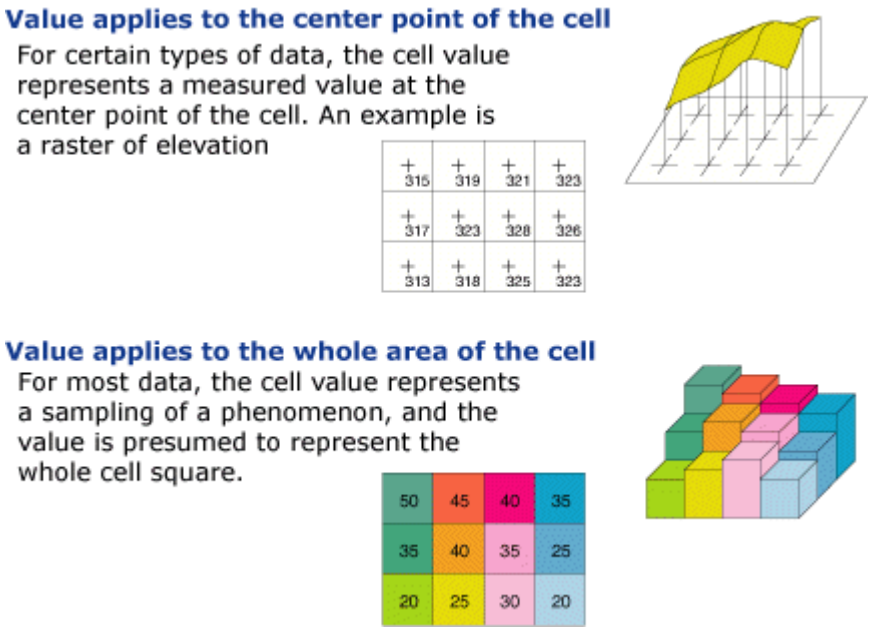

Figure 2.2: An example of how are built the cell values in raster data.

raster will be. However, the greater the number of cells, the longer it will take to process all the available information and it will increase the demand for storage space. If a cell size is too large, information may be lost or subtle patterns may be obscured. For example, if the cell size is larger than the width of a road, the road may not exist within the raster dataset. Figure 2.3 shows how a simple polygon feature will be represented by a raster dataset at various cell sizes.

As it was described, process of raster clearly results in a loss of information, from the real-valued coordinates of the points, through the integer cell counts. However, there are multiple gains. First one, the data structure is more compact and easy to visualize. Furthermore, it can be related to other rasters provided the locations and resolutions are properly conflated wherewith complex spatial statistical analysis could be performed.

2.3 Qualitative analysis of spatio-temporal raster datasets

Spatio-temporal raster datasets are raster datasets with an intrinsic temporal component, i. e. the cell values of raster data have a temporal dimension as well. The sea surface temperature, for a

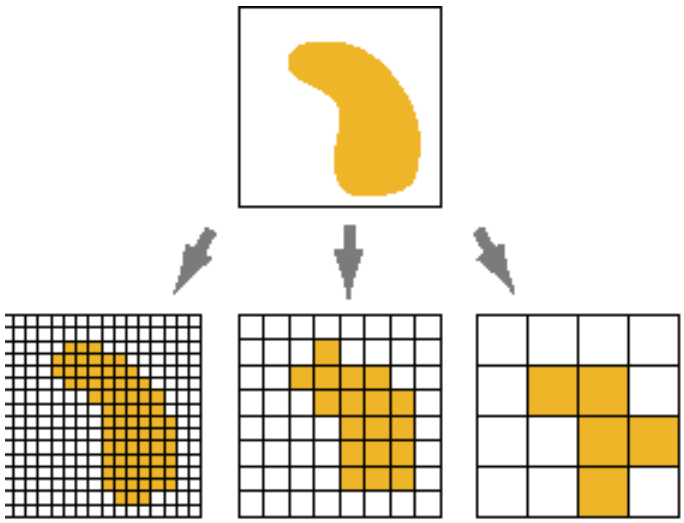

Figure 2.3: An example of how a simple polygon feature is read through raster dataset at different resolutions.

region collected over a period of time is an example of this kind of data. Figure 2.4 displays how the spatial and temporal dimensions are portrayed in spatio-temporal raster datasets.

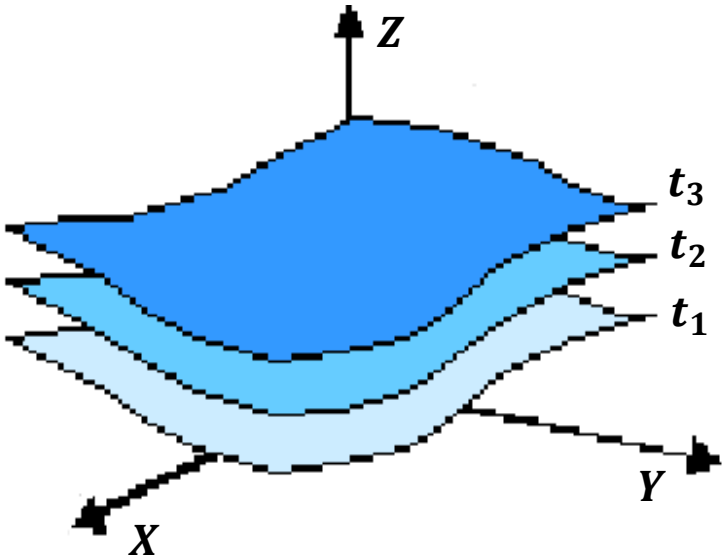

Figure 2.4: Visual representation of spatio-temporal raster datasets. X and Y axes represent the coordinate system for the spatial location and the Z axis represents the temporal evolution of rasters.

Due to the intrinsic spatial and temporal dimensions, the spatio-temporal statistical analysis is concerned with data variation in space and time. Hence the statistical effort will be concentrated in the analysis of temporal variability, spatial variability and its possible interactions. In order to deal with these problems, next subsections describe some necessary statistical concepts.

2.3.1 Analysis of time series: the temporal variability

Definition 1. *For a given probability space* $(\Omega, \mathcal{F}, \mathbb{P})$ *and a measurable space* $(S, \mathcal{S})$*, a stochastic process is a collection of* $S-$*valued random variables which can be written as*

$$\{X(t) : t \in P\}. \tag{2.1}$$

Where S *and* P *are known as the state space and parameter space of the stochastic process. If* T *is discrete (continuous), equation 2.1 is called a discrete (continuous) stochastic process.*

A *time series* can be seen as a realization of a stochastic process with a discrete-time observation support. In other words, a time series is a collection of observations made sequentally in time and indexed by integers. A time series shall be represented by the discrete stochastic process $\{x_t\}_{t=1}^{T}$, where T represents the length of the series. According to the number of variables analyzed through time, a time series can be either *univariate* ($x_t \in \mathbb{R}$) or *multivariate* ($x_t \in \mathbb{R}^n, n > 1$).

Two of the main goals of the analysis of time series are *modeling* and *forecasting*. The aim of modeling is to find a mechanism to describe the data generation process. In theory, this requires a complete knowledge of the laws of physics, biology, etc., which govern the evolution of the environment related to the involved time series. However, in practice there is little or no prior knowledge about all the factors that influence the evolution. In that case, it is possible to build a model using only the information coming from the time series, by studying its behavior in the past. In general, a good model should capture essential features of the long-term behavior of the system. On the other hand, the goal of forecasting, or *predicting*, is mainly to predict the future short-term evolution of the system. These two goals are not necessarily intersecting: a model that properly describes long-term governing mechanisms may fail in giving reliable short-term forecasts; and a model that is accurate in terms of short-term predictions may not give (and does not need to

give) a good insight into the long-term properties of the system (Qi and Zhang, 2001). The present thesis has as main objective the forecast.

Traditionally, methods of time series analysis are mainly concerned with decomposing the total variation of a series into four components of variability: trend, seasonal variation, cycle, and irregular fluctuations (see figure 2.5). These variation components do not usually occur alone; in fact, they can occur in any combination or can occur all together (Qi and Zhang, 2001).

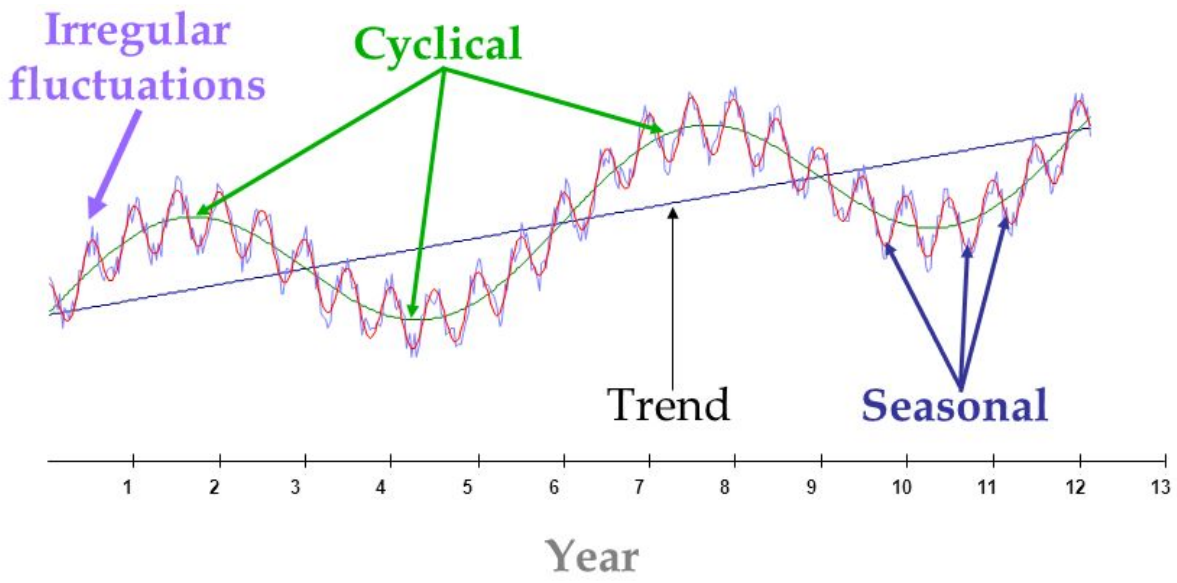

Figure 2.5: A time series with the four traditional variations: trend, seasonal variations, cycle and irregular fluctuations.

The crucial difference between time series, and situations usually considered in classical statistics, is that the measurements x_m and x_n; $m \neq n$, will be stochastically dependent. Indeed, almost always a sequence of measurements made in equally spaced time instants is regarded as a realization of a process which, in principle, has continuous sample paths $\{x_t\}_{t\in\mathbb{R}}$. Then, as s and t get close, the variables x_s and x_t tend to be dependent. However, the way how they depend defines the stationarity of the time series.

Definition 2. *A time series $\{x_t\}_{t=1}^{\infty}$ is weak stationary (or simply called stationary) if:*

1. $\mathbb{E}(x_t) = \mu$ *for all* t,

2. $V(x_t) = \sigma^2$ *for all* t,

3. $\gamma(x_{t_1}, x_{t_2}) = f(t_2 - t_1)$ *for* $t_1 < t_2$,

where $V(.)$ *and* $\gamma(.,.)$ *denote the variance and the auto-covariance function, respectively.*

Condition 1 defines *first-order stationarity* and conditions 2 and 3 define *second-order stationarity*. Both concepts give place to the stationarity. This concept is important since most forecasting methods assume that involved time series are stationary. An absence of stationarity can cause unexpected or bizarre behaviors, like t-ratios not following a t-distribution or high R-squared values assigned to variables that are not correlated at all. However, in real applications time series are not stationaries but data can be stationarized through mathematical transformations.

Statistical performance measures

In this study, several measures to evaluate the performance of different forecasting models are employed. Given a time series $\{x_t\}_{t=1}^{T}$ and corresponding predicted values $\hat{x}_t$, the performance measures to evaluate in-sample (adjustment) and out-sample (forecast) results are defined as

1. *The Mean Absolute Error* (MAE)

$$MAE = \frac{1}{T}\sum_{t=1}^{T} |x_t - \hat{x}_t|. \tag{2.2}$$

2. *Mean Absolute Percentage Error* (MAPE)

$$MAPE = \frac{1}{T}\sum_{t=1}^{T} \frac{|x_t - \hat{x}_t|}{|x_t|} \times 100\%. \tag{2.3}$$

2.3.2 Analysis of spatial datasets: the spatial variability

Spatial data are geo-referenced attribute measurements (continuous or discrete) where each measurement is associated with a location (point) or an entity (region or object) in a geographical space. Sample locations can have a regular or irregular spatial arrangement, i.e. data locations

on a regular lattice (rasters) or scattered in space (see figure 2.6). The domain informed by a measurement is called the *sample unit*, e.g. pixels (or cells) for raster datasets.

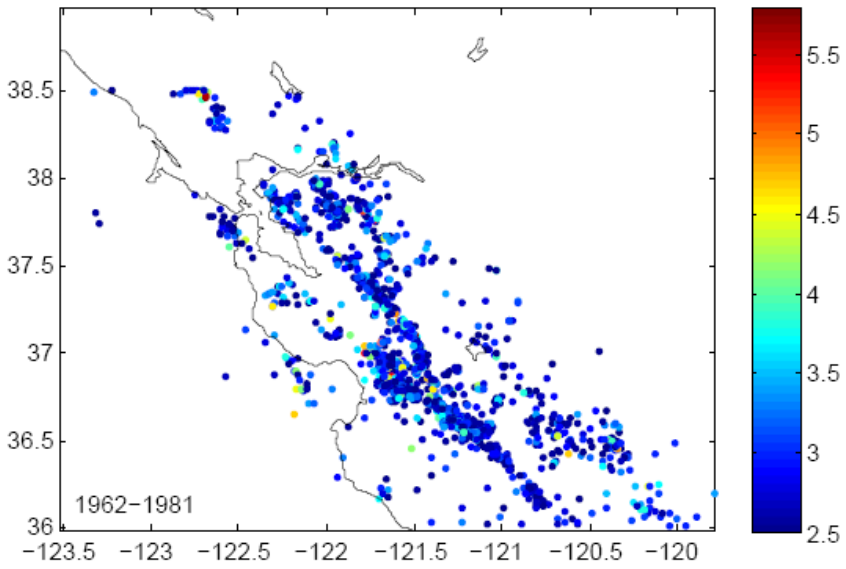

Figure 2.6: Measures of earthquake magnitudes (in Richter scale) at different point locations in a bay area since 1962 to 1981. This is an example of spatial data measured in specific point locations in a continuous space.

Formally, spatial statistics deals with the analysis of realizations of a stochastic process

$$\{Z_s : s \in D \subset \mathbb{R}^p, p > 0\}; \tag{2.4}$$

where s is the location in the p-dimensional Euclidean space and Z_s is a random variable in the location s. In this thesis, the interest is concentrated when $p = 2$.

According to the nature of D in equation (2.4), spatial statistics is subdivided into three large fields:

1. *Geostatistics*: It studies data of stochastic processes in which the parameter space $D \subset \mathbb{R}^2$ is continuous. However, in practice it depends on the researcher to select in which sites of the region of interest the measurement of the variables is made, that is, the researcher can select points of the space at convenience or follow some probabilistic sampling scheme. In that way,

it is said that the set D is fixed. It is important to highlight that in geostatistics the essential purpose is *interpolation* due to the spatial continuity.

2. *Aerial data analysis*: In this case the involved stochastic process has a discrete parameter space $D \subset \mathbb{R}^2$ and the selection of the measurement sites depends on the researcher, i.e. D is fixed. The sampling locations may be regularly (such as rasters) or irregularly spaced (see figure 2.7). The essential purpose of the analysis is to detect and model spatial patterns or trends in area values. Spatial interpolation is meaningless in this context unless it is necessary to input missing values.

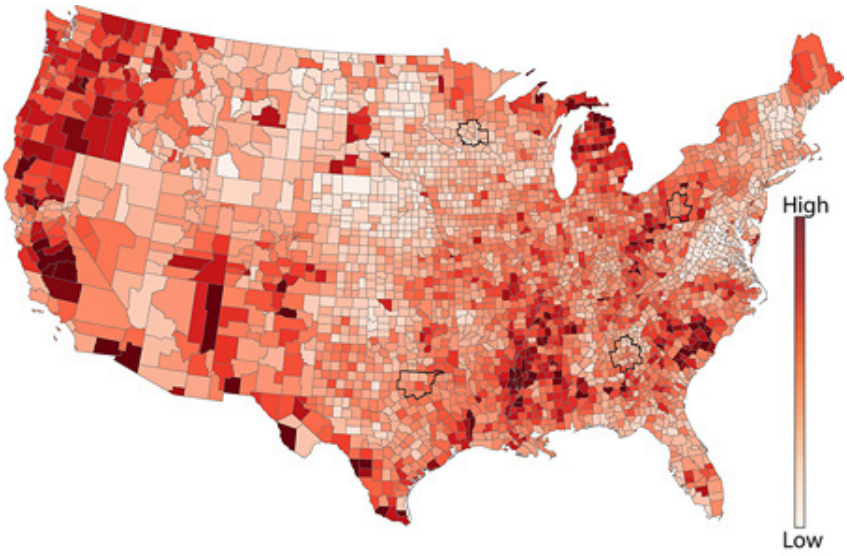

Figure 2.7: Measurements of an atributte in the different states of the United States. This is an example of aerial data irregularly spaced.

3. *Point patterns:* The main difference of the point pattern analysis with the two fields mentioned above lies in the fact that $D \subset \mathbb{R}^2$ is not fixed. D can be discrete or continuous but the site locations where the phenomenon to be studied occurs is given by nature. In general, here the main purpose is to determine if the distribution of individuals within the region is random, aggregate or uniform. Figure 2.6 shows an example of spatial point pattern.

A very important stage in the analysis of spatial data is the determination of the spatial autocorrelation structure. Analogous to how it happens in the case of time series, the establishment of

the association or similarity of the values according to the distance between them is by itself a result that allows to characterize the population in study and is also fundamental in the development of prediction models. So, the concept of spatial autocovariance function is introduced next.

Definition 3. *Given a spatial process* $\{Z_s : s \in D \subset \mathbb{R}^2\}$*, the spatial auto-covariance function is defined as*

$$C(s_1, s_2) = Cov(Z_{s_1}, Z_{s_2}) = \mathbb{E}[(Z_{s_1} - \mathbb{E}(Z_{s_1}))(Z_{s_2} - \mathbb{E}(Z_{s_1}))] \, , \forall s_1, s_2 \in D. \tag{2.5}$$

The spatial auto-covariance function, referred onwards as the spatial covariance function, completely determines the joint dispersion structure implied by the spatial process. To be precise, for any n and any arbitrary collection of sites $D = \{s_1, s_2, \ldots, s_n\}$, the $n \times 1$ vector of realizations $Z = (Z_{s_j})_{j=1,\ldots,n}$ will have the covariance matrix given by $\Sigma_Z = (C(s_i, s_j))_{i,j=1,\ldots,n}$, where by property Σ_Z is symmetric and positive-definite.

Definition 4. *The spatial process* $\{Z_s : s \in D \subset \mathbb{R}^2\}$ *is said to be stationary if*

$$\mathbb{E}(Z_s) = \mu, \forall s \in D; \tag{2.6}$$

$$V(Z_s) = \sigma^2, \forall s \in D; \tag{2.7}$$

$$C(s_1, s_2) = g(s_1 - s_2), \forall s_1 \neq s_2, s_1 \in D, s_2 \in D; \tag{2.8}$$

where $g : \mathbb{R} \to \mathbb{R}$ *is a function depending only on* $s_1 - s_2$*.*

Particularly, two stationary spatial processes will be used in the present work: *The isotropic and anisotropic processes.*

Definition 5. *A spatial process* $\{Z_s : s \in D \subset \mathbb{R}^2\}$ *is called isotropic if it is stationary and*

$$C(s_1, s_2) = g(\|s_1 - s_2\|); \tag{2.9}$$

where $\|.\|$ *is a norm operator and* $g(.)$ *is a function similar to definition 4.*

Definition 6. *A spatial process* $\{Z_s : s \in D \subset \mathbb{R}^2\}$ *is called anisotropic if it is stationary and*

$$C(s_1, s_2) = g\left(\left\|A^{-1/2}(s_1 - s_2)\right\|\right); \tag{2.10}$$

where $\|.\|$ *is a norm operator,* $g(.)$ *is a function similar to definition 4 and* A *is a* 2×2 *positive definite matrix, often called the anisotropy matrix.*

Note that for stationary processes, the spatial covariance function can be written as $C(h) = C(s, s + h)$. So, in order to specify an stationary process a valid covariance function must be provided, i.e. Σ_Z must be symetric and positive-definite. In particular, three spatial auto-covariance functions are of interest: The exponential, the spherical and the Mátern form. Chapter 4 deals with more details of them.

2.3.3 Analysis of spatio-temporal datasets: the spatio-temporal variability

Spatio-temporal datasets are characterized by having measures of variables in an specific geographical and temporal location. In contrast with non spatio-temporal datasets, spatio-temporal data can therefore be separated into three distinct components: geographic space, temporal space and attribute space. The spatio-temporal data can consist of a p-dimensional variable space, two dimensional geographic space, and one-dimensional temporal space, where the space–time components provide the framework for attribute space. Figure 2.4 shows an example of this structure.

Formally spatio-temporal data can be modeled through the stochastic process

$$\{Z(s,t) : s \in D \subset \mathbb{R}^2, t \in [0, +\infty)\}; \tag{2.11}$$

where s is the location in the bidimensional Euclidean space, t is the time position and $Z(s,t)$ is a random variable in the location s at time t.

Suppose a realization of a spatio-temporal model with $D = \{s_1, s_2, \ldots, s_n\}$ and $t \in \{t_1, t_2, \ldots, t_T\}$. Traditionally, the ultimate goal of spatio-temporal analysis is the *optimal prediction* (in space and

time) of the unobserved parts of the process, based on the observations

$$\boldsymbol{Z} = (Z(s_1, t_1), \ldots, Z(s_n, t_T))'. \tag{2.12}$$

To achieve this goal, a complete study of the space-time sources of variability is needed. *Spatial and temporal heterogeneity* and *spatial and temporal autocorrelation* are properties than make a difference between spatio-temporal and traditional datasets. Spatial (temporal) heterogeneity refers to the non-stationarity of geographic (temporal) processes, meaning that processes can vary locally and are not necessarily the same at each spatial (temporal) location. Commonly, this non-stationarity is modeled as a first-order (mean response) or second-order (auto-covariance) effect. Spatial (temporal) autocorrelation is the tendency of attributes at some location in space to be related. Spatial (temporal) autocorrelation, as it was mentioned before, is a second-order effect.

Definition 7. *Given the spatio-temporal process* $\{Z(s,t) : s \in D \subset \mathbb{R}^2, t \in [0, +\infty)\}$*, the spatio-temporal covariance function is defined as*

$$C(s_1, s_2, t_1, t_2) = Cov(Z(s_1, t_1), Z(s_2, t_2)) \ \ , \forall s_1, s_2 \in D, \ \forall t_1, t_2 \in [0, +\infty). \tag{2.13}$$

If the spatio-temporal process is first-order stationary (i.e. $\mathbb{E}(Z(s,t)) = \mu, \forall s, t$) then the process can have:

1. *Temporal stationarity*, if

$$C(s_1, s_2, t_1, t_2) = f(s_1, s_2, t_2 - t_1) \ \ , \forall s_1, s_2 \in D, \ \forall t_1, t_2 \in [0, +\infty). \tag{2.14}$$

2. *Spatial stationarity*, if

$$C(s_1, s_2, t_1, t_2) = f(s_1 - s_2, t_1, t_2) \ \ , \forall s_1, s_2 \in D, \ \forall t_1, t_2 \in [0, +\infty). \tag{2.15}$$

3. *Spatio-temporal stationarity*, if

$$C(s_1, s_2, t_1, t_2) = f(s_1 - s_2, t_1 - t_2) \ \ , \forall s_1, s_2 \in D, \ \forall t_1, t_2 \in [0, +\infty). \tag{2.16}$$

4. *Separability*, if

$$C(s_1, s_2, t_1, t_2) = f_1(s_1, s_2) f_2(t_1, t_2) \ \ , \forall s_1, s_2 \in D, \ \forall t_1, t_2 \in [0, +\infty). \tag{2.17}$$

Valid spatial covariance and temporal covariance models are readily available in the literature (see e.g., Ver Hoef and Cressie, 1993). They can be combined in a product form via the separability property (equation (2.17)) to give valid spatio-temporal covariance models.

Many of the problems of space and time modeling can be overcome by using separable processes. This subclass of spatio–temporal processes has several advantages, including rapid fitting and simple extensions of many techniques developed and successfully used in time series and classical geostatistics. Furthermore, this class of spatio–temporal processes offers enormous computational benefits, because the covariance matrix of $\boldsymbol{Z}$ (equation (2.12)) can be expressed as the Kronecker product of two smaller matrices that arise separately from the temporal and purely spatial processes, and then its determinant and inverse are easily determinable. Thus, separability is a desirable property for spatial–temporal processes.

2.4 Principal Component Analysis

The central idea of principal component analysis (PCA) is to reduce the dimensionality of a dataset consisting of a large number of interrelated variables, retaining the meaningful variability present in the dataset (Jolliffe, 2002). This goal is achieved by transforming the original data into a new set of variables, the principal components, which are uncorrelated. They are ordered is such a way that *few* ones retain most of the variation present in all of the original variables (see Figure 2.8).

More generally, in PCA a few linear combinations which can be used to summarize the data information, losing in the process as little information as possible. This attempt to reduce dimensionality can be described as "parsimonious summarization" of the information data. For example, figure 2.8 shows how a group of trivariate observations is transformed through PCA in other group of bivariate observations in order to get a better interpretation of a dataset.

The theory of PCA can be seen from the population and the sample perspectives. The math-

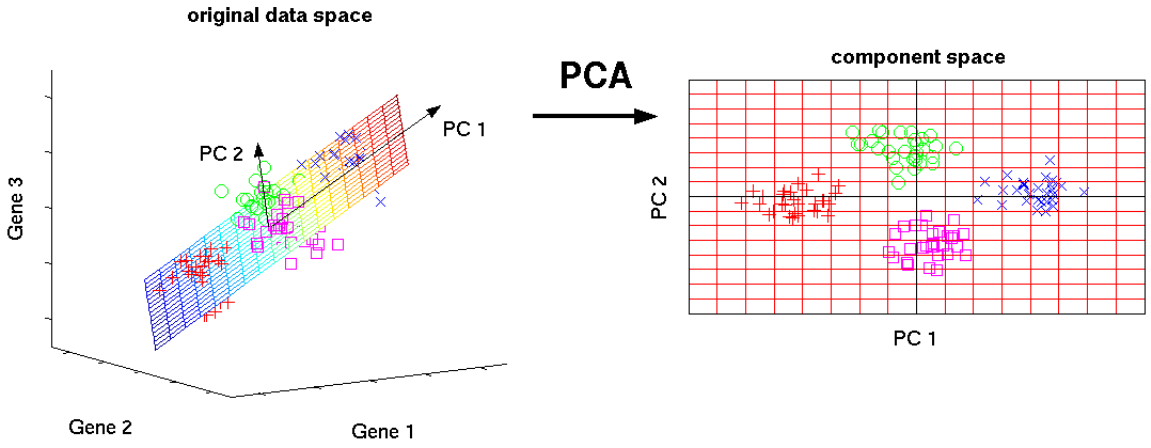

Figure 2.8: An example of how principal component analysis works.

ematical background of population and sample PCA are studied in subsections 2.4.1 and 2.4.2, respectively. Subsection 2.4.3 presents a brief discussion about PCA applied on a spatio-temporal raster dataset and finally subsection 2.4.4 discusses about some methods to retain the optimal number of principal components.

2.4.1 Population principal components

Algebraically, principal components are particular linear combinations of the p random variables $x_1, x_2, \ldots, x_p$. Geometrically, these linear combinations represent the selection of a new coordinate system obtained by rotating the original system with $x_1, x_2, \ldots, x_p$ as the coordinate axes. The new axes represent the directions with maximum variability and provide a simpler and more parsimonious description of the covariance (or correlation) structure.

Principal components depend only on the covariance matrix $\boldsymbol{\Sigma}$ (or the correlation matrix $\boldsymbol{\rho}$) of $x_1, x_2, \ldots, x_p$, and their development does not require a multivariate normal assumption. Let $\boldsymbol{\Sigma}$ be the variance-covariance matrix of the random vector $\mathbf{x}' = (x_1, x_2, \ldots, x_p)$. Let $\{\lambda_1, \ldots, \lambda_p\}$ dentote th eigenvalues of $\boldsymbol{\Sigma}$, such that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p \geq 0$.

Consider the linear combinations $y_i = \mathbf{a}_i'\mathbf{x}$, for $i = 1, 2, \ldots, p$. Then, their respectives variances

and covariances are given by

$$Var(y_i) = \mathbf{a}_i' \mathbf{\Sigma} \mathbf{a}_i \quad for \quad i = 1, 2, \ldots, p\,; \tag{2.18}$$

$$Cov(y_i, y_k) = \mathbf{a}_i' \mathbf{\Sigma} \mathbf{a}_k \quad for \quad i, k = 1, 2, \ldots, p. \tag{2.19}$$

The principal components are those *uncorrelated* linear combinations $y_1, y_2, \ldots, y_p$ whose variances in equation (2.18) are as large as possible.

The first principal component is the linear combination with maximum variance. That is, it maximizes $Var(y_1) = \mathbf{a}_1' \mathbf{\Sigma} \mathbf{a}_1$. It is clear that $Var(y_1)$ can be increased by multiplying any $\mathbf{a}_1$ by some constant. To eliminate this indeterminacy, it is convenient to restrict attention to coefficient vectors of unit length.

Definition 8. *Given a set of p random variables $\mathbf{x} = (x_1, \ldots, x_p)'$, the population principal components are the variables $y_1, \ldots, y_p$ obtained through the process:*

First principal component (y_1) = linear combination $\mathbf{a}_1'\mathbf{x}$ that maximizes
$Var(\mathbf{a}_1'\mathbf{x})$ *subject to* $\mathbf{a}_1'\mathbf{a}_1 = 1$.
Second principal component (y_2) = linear combination $\mathbf{a}_2'\mathbf{x}$ that maximizes
$Var(\mathbf{a}_2'\mathbf{x})$ *subject to* $\mathbf{a}_2'\mathbf{a}_2 = 1$ *and* $Cov(\mathbf{a}_1'\mathbf{x}, \mathbf{a}_2'\mathbf{x}) = 0$.

At the i^{th} step, for $i = 3, \ldots, p$

i^{th} principal component (y_i) = linear combination $\mathbf{a}_i'\mathbf{x}$ that maximizes
$Var(\mathbf{a}_i'\mathbf{x})$ *subject to* $\mathbf{a}_i'\mathbf{a}_i = 1$
and $Cov(\mathbf{a}_i'\mathbf{x}, \mathbf{a}_k'\mathbf{x}) = 0$ *for* $k < i$.

With this definition, it is possible to get a characterization of the population principal components through the eigenvalues and eigenvectors of $\mathbf{\Sigma}$. The next lemma will help to demonstrate this characterization.

Lemma 1. *Let $\mathbf{B}_{p\times p}$ be a positive definite matrix with eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p \geq 0$ and associated normalized eigenvector $\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_p$. Then*

$$\max_{\mathbf{x}\neq\mathbf{0}} \frac{\mathbf{x}'\mathbf{B}\mathbf{x}}{\mathbf{x}'\mathbf{x}} = \lambda_1 \quad (attained\ when\ \mathbf{x} = \mathbf{e}_1). \tag{2.20}$$

Moreover,

$$\max_{\mathbf{x}\perp\mathbf{e}_1,\ldots,\mathbf{e}_k} \frac{\mathbf{x}'\mathbf{B}\mathbf{x}}{\mathbf{x}'\mathbf{x}} = \lambda_{k+1} \quad (\textit{attained when } \mathbf{x} = \mathbf{e}_{k+1}, k = 1, 2, \ldots, p-1); \tag{2.21}$$

where the symbol $\perp$ is read "is orthogonal to."

Proof. For details of the proof, see Johnson et al. (2014) □

Theorem 1. *Let $\mathbf{\Sigma}$ be the covariance matrix associated with the random vector $\mathbf{x}' = (x_1, x_2, \ldots, x_p)$. Let $(\lambda_1, \mathbf{e}_1)$, $(\lambda_2, \mathbf{e}_2)$, $\ldots$, $(\lambda_p, \mathbf{e}_p)$ denote the eigenvalue-eigenvector pairs of $\mathbf{\Sigma}$, such that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p \geq 0$. Then the i^{th} principal component is given by*

$$y_i = \mathbf{e}_i'\mathbf{x}, \quad for \;\; i = 1, 2, \ldots, p. \tag{2.22}$$

With these choices,

$$Var(y_i) = \mathbf{e}_i'\mathbf{\Sigma}\mathbf{e}_i = \lambda_i \;\; for \;\; i = 1, 2, \ldots, p; \tag{2.23}$$

$$Cov(y_i, y_k) = \mathbf{e}_i'\mathbf{\Sigma}\mathbf{e}_k = 0 \;\; for \;\; i \neq k. \tag{2.24}$$

Proof. By lemma 1, with $\mathbf{B} = \mathbf{\Sigma}$, gives

$$\max_{\mathbf{a}\neq\mathbf{0}} \frac{\mathbf{a}'\mathbf{\Sigma}\mathbf{a}}{\mathbf{a}'\mathbf{a}} = \lambda_1 \quad (\textit{attained when } \mathbf{a} = \mathbf{e}_1).$$

But $\mathbf{e}_1'\mathbf{e}_1 = 1$ since the eigenvectors are normalized. Thus,

$$\lambda_1 = \frac{\mathbf{e}_1'\mathbf{\Sigma}\mathbf{e}_1}{\mathbf{e}_1'\mathbf{e}_1} = \mathbf{e}_1'\mathbf{\Sigma}\mathbf{e}_1 = Var(y_1).$$

Similarly, using lemma 1 again, it follows that

$$\max_{\mathbf{a}\perp\mathbf{e}_1,\ldots,\mathbf{e}_k} \frac{\mathbf{a}'\mathbf{\Sigma}\mathbf{a}}{\mathbf{a}'\mathbf{a}} = \lambda_{k+1} \quad for \; k = 1, 2, \ldots, p-1.$$

For the choice $\mathbf{a} = \mathbf{e}_{k+1}$ with $\mathbf{e}'_{k+1}\mathbf{e}_i = 0$, for $i = 1, 2, \ldots, k$ and $k = 1, 2, \ldots, p-1$,

$$\frac{\mathbf{e}'_{k+1}\boldsymbol{\Sigma}\mathbf{e}_{k+1}}{\mathbf{e}'_{k+1}\mathbf{e}_{k+1}} = \mathbf{e}'_{k+1}\boldsymbol{\Sigma}\mathbf{e}_{k+1} = Var(y_{k+1}).$$

However, $\mathbf{e}'_{k+1}(\boldsymbol{\Sigma}\mathbf{e}_{k+1}) = \lambda_{k+1}\mathbf{e}'_{k+1}\mathbf{e}_{k+1} = \lambda_{k+1}$, therefore $Var(y_{k+1}) = \lambda_{k+1}$. It remains to show that if $\mathbf{e}_i$ and $\mathbf{e}_k$ are orthogonal vectors (for $i \neq k$) gives $Cov(y_i, y_k) = 0$. Now, the eigenvectors of $\boldsymbol{\Sigma}$ are orthogonal if all the eigenvalues $\lambda_1, \ldots, \lambda_p$ are distinct. If the eigenvalues are not all distinct, the eigenvectors corresponding to common eigenvalues may be chosen to be orthogonal. Therefore, for any two eigenvectors $\mathbf{e}_i$ and $\mathbf{e}_k$, $\mathbf{e}'_i\mathbf{e}_k = 0$ for $i \neq k$. Since $\boldsymbol{\Sigma}\mathbf{e}_k = \lambda_k\mathbf{e}_k$, premultiplication by $\mathbf{e}_i$ gives

$$Cov(y_i, y_k) = \mathbf{e}'_i\boldsymbol{\Sigma}\mathbf{e}_k = \lambda_k\mathbf{e}'_i\mathbf{e}_k = 0;$$

for any $i \neq k$, and the proof is complete. □

Due to this characterization of principal components, some observations must be taken into account. An enumeration of them are shown below:

1. If some λ_i are equal, the choices of the corresponding coefficient vectors, $\mathbf{e}_i$, and hence y_i, are not unique.

2. Principal components may also be obtained for standardized variables $z_i = (x_i - \mu_i)/\sqrt{\sigma_{ii}}$ $(i = 1, \ldots, p)$ where μ_i and σ_{ii} are the corresponding mean and variance of x_i. If so, $\mathbf{z} = (z_1, z_2, \ldots, z_p)'$ would be

$$\mathbf{z} = D^{-1/2}(\mathbf{x} - \boldsymbol{\mu}); \tag{2.25}$$

   where $\boldsymbol{\mu} = \mathbb{E}(\mathbf{x})$ and $D^{-1/2} = diag(1/\sqrt{\sigma_{ii}})_{i=1,\ldots,p}$. Clearly, $Cov(\mathbf{z}) = D^{-1/2}\boldsymbol{\Sigma}D^{-1/2} = \boldsymbol{\rho}$, which is the *correlation matrix* of $\mathbf{x}$. Thus, the principal components of standardized variables are associated to eigenvalues - eigenvectors of its correlation matrix.

3. The total population variance (the trace of population covariance matrix) remains constant

in the principal components. It means

$$\sum_{i=1}^{p} Var(x_i) = \sum_{i=1}^{p} Var(y_i). \tag{2.26}$$

To get this is enough to realize that $tr(\mathbf{\Sigma}) = \sum_{i=1}^{p} \lambda_i$.

For a complete review of the PCA methodology and other details see for example Jolliffe (2002) and Mardia et al. (1980).

2.4.2 Sample principal components

This subsection deals with properties of principal components obtained from a random sample $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ representing n independent drawings from some $p-$dimensional population with mean vector $\boldsymbol{\mu}$ and covariance matrix $\mathbf{\Sigma}$. This dataset yields the sample mean vector $\overline{\mathbf{x}}$, the sample covariance matrix $\mathbf{S}$ and the sample correlation matrix $\mathbf{R}$. The principal components obtained from the sample covariance matrix (or sample correlation matrix) are called *sample principal components*.

The process to obtain the sample principal components is analogous to the population PCA process but with subtle details. To see this, let

$$\mathbf{X} = (\mathbf{x}_1 : \mathbf{x}_2 : \cdots : \mathbf{x}_n)'; \tag{2.27}$$

be the data matrix due to the n $p-$variate observations. As PCA works with the information of variables, a partition of $\mathbf{X}$ by columns is necessary. Let $\mathbf{X} = (\mathbf{x}_{(1)} : \mathbf{x}_{(2)} : \cdots : \mathbf{x}_{(p)})$ be the matrix $\mathbf{X}$ partitioned by columns, then the sample principal components are defined as those linear combinations $\hat{\mathbf{y}}_i = a_{i1}\mathbf{x}_{(1)} + a_{i2}\mathbf{x}_{(2)} + \cdots + a_{ip}\mathbf{x}_{(p)} = \mathbf{X}\mathbf{a}_i$ which have maximum sample variance, constrained to $\mathbf{a}_i'\mathbf{a}_i = 1$. Next definition formalize this concept.

Definition 9. *Given a set of n independent observations of $p-$dimensional random variables, $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$, and the data matrix $\mathbf{X}$ defined as equation (2.27). Then, the sample principal components are the vectors $\hat{\mathbf{y}}_1, \ldots, \hat{\mathbf{y}}_p$ obtained through the process:*

- *First sample principal component: $\hat{\mathbf{y}}_1 = \mathbf{X}\mathbf{a}_1$ that maximizes the sample variance of $\mathbf{X}\mathbf{a}_1$*

subject to $\mathbf{a}_1'\mathbf{a}_1 = 1$.

- *Second sample principal component:* $\hat{\mathbf{y}}_2 = \mathbf{X}\mathbf{a}_2$ *that maximizes the sample variance of* $\mathbf{X}\mathbf{a}_2$ *subject to* $\mathbf{a}_2'\mathbf{a}_2 = 1$ *and sample covariance*$(\mathbf{X}\mathbf{a}_1, \mathbf{X}\mathbf{a}_2) = 0$.

- *At* i^{th} *step, from* $i = 3$ *to* p*:*
  i^{th} *sample principal component:* $\hat{\mathbf{y}}_i = \mathbf{X}\mathbf{a}_i$ *that maximizes the sample variance of* $\mathbf{X}\mathbf{a}_i$ *subject to* $\mathbf{a}_i'\mathbf{a}_i = 1$ *and sample covariance*$(\mathbf{X}\mathbf{a}_i, \mathbf{X}\mathbf{a}_k) = 0$, $k < i$.

Theorem 2 solves the optimization problem involved in definition of sample principal components and gives an explicit characterization of them.

Theorem 2. *Let* $\mathbf{S}$ *be the sample covariance matrix associated with a data matrix* $\mathbf{X} = (\mathbf{x}_1 : \mathbf{x}_2 : \cdots : \mathbf{x}_n)'$ *formed by* n *independent drawings from some* $p-$*dimensional population. Let* $(\hat{\lambda}_1, \hat{\mathbf{e}}_1)$, $(\hat{\lambda}_2, \hat{\mathbf{e}}_2)$, $\ldots$, $(\hat{\lambda}_p, \hat{\mathbf{e}}_p)$ *be the eigenvalue-eigenvector pairs of* $\mathbf{S}$*, where* $\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \cdots \geq \hat{\lambda}_p \geq 0$. *Then the* i^{th} *sample principal component is given by*

$$\hat{\mathbf{y}}_i = \mathbf{X}\hat{\mathbf{e}}_i \quad for\ i = 1, 2, \ldots, p. \tag{2.28}$$

Also, it follows that

$$\begin{aligned} sample\ variance(\hat{\mathbf{y}}_i) &= \mathbf{s}_{\hat{\mathbf{y}}_i, \hat{\mathbf{y}}_i} = \hat{\lambda}_i \quad i = 1, 2, \ldots, p; & (2.29)\\ sample\ covariance(\hat{\mathbf{y}}_i, \hat{\mathbf{y}}_k) &= \mathbf{s}_{\hat{\mathbf{y}}_i, \hat{\mathbf{y}}_k} = 0 \quad i \neq k. & (2.30) \end{aligned}$$

Proof. The proof is similar as its analogous population PCA process (See theorem 1). □

Given the sample principal components structure, it will be convenient to define the matrix of sample principal components in this way

$$(\hat{\mathbf{y}}_1 : \hat{\mathbf{y}}_2 : \cdots : \hat{\mathbf{y}}_p) = (\mathbf{X}\hat{\mathbf{e}}_1 : \mathbf{X}\hat{\mathbf{e}}_2 : \cdots : \mathbf{X}\hat{\mathbf{e}}_p) = \mathbf{X}(\hat{\mathbf{e}}_1 : \hat{\mathbf{e}}_2 : \cdots : \hat{\mathbf{e}}_p);$$

$$\mathbf{Z} = \mathbf{X}\mathbf{P}; \tag{2.31}$$

where $\mathbf{Z}_{n\times p}$ is called the matrix of sample principal components, also known as *the component score matrix*, and $\mathbf{P}$ the eigenvector matrix of $\mathbf{S}$ (o $\mathbf{R}$), also known as *the component loading matrix.* From now onwards, the sample principal components and the sample PCA process will be referred as merely *principal components* and *PCA* respectively.

Remark 1. *Observe that one can choose either the sample covariance (*$\mathbf{S}$*) or the sample correlation (*$\mathbf{R}$*) matrix to find the principal components. However, in this project the use of sample correlation matrix is preferred to the sample covariance matrix, because the sample covariance matrix will not provide very informative principal components if there are variables with widely higher variance than the others; furthermore, when using the sample covariance matrix it is more complicated to compare the results from different analyses (Jolliffe, 2002).*

2.4.3 PCA in a spatio-temporal context

As it was seen before, PCA maps the original n observations of a $p-$variate population in the data matrix $\mathbf{X}$ onto a new orthogonal space, such that the new axes are oriented in directions of largest variance in the dataset. This structure of dataset is considered *traditional data*; that is, the n measurements have a constant and specific position in space and time. So, what would happen if a PCA is used in a dataset with an explicit space and time structure?, would it be possible to apply a PCA over a spatio-temporal dataset?, if so, how would the results of a PCA be interpreted?. In the next paragraphs, a brief analysis of the consequences when using a PCA in a spatio-temporal dataset is shown.

A review in literature has shown that PCA was applied over a dataset with a space-time structure, especially in climatology and meteorology fields (see for example Richman, 1986; Preisendorfer and Mobley, 1988). When dealing with spatio-temporal structures three subdimensions need to be considered: temporal, spatial dimension and the attribute (or variable) dimensions, respectively. According to this, Richman (1986) established six different operational modes of use of PCA to deal with space-time series data: O, P, Q, R, S and T (see table 2.1). These modes are classified by having the data matrix $\mathbf{X}$ for PCA defined by two out of the three subdimensions of the space–time dataset. The six modes are shown in table 2.1.

Table 2.1: The six mode of decomposition and how is related to the structure of its respective data matrix $\mathbf{X}$ (equation (2.27)).

PC mode	Columns representation	Rows representation	Fixed dimension
O	time	attribute	space
P	attribute	time	space
Q	space	attribute	time
R	attribute	space	time
S	space	time	attribute
T	time	space	attribute

Different modes provide different insight results and interpretations of them (Richman, 1986). In particular, for purposes of this project only the S and T modes are of interest. Observe that both modes work only with one variable which is measured at a specific time position and space raster location. A brief discussion of the S and T mode in a spatio-temporal raster dataset is carried out below.

The S mode considers the sampling raster locations as variables and sampling times as data elements (see figure 2.9). Because of this, covariance/correlation matrix is calculated between each pair of sampling raster locations and not between two measured variables as traditional PCA. Therefore, location does play a role in calculation of the PCs. Also, the data elements are not more independent and interpretation of PCs must consider this intrinsic temporal characteristic. Mathematically, consider a bidimensional geographic location given by the latitude and longitude position. A spatio-temporal raster dataset could be seen as an array containing for each vertical level a three-dimensional, two-dimensional in space and one-dimensional in time, field X. This field is a function of time t, latitude θ, and longitude ϕ. Suposse that the horizontal coordinates are discretised to yield latitudes $\theta_j, j = 1, 2, \ldots, p_1$, and longitudes $\phi_k, k = 1, 2, \ldots, p_2$, and similarly for time, i.e. $t_i, i = 1, 2, \ldots, n$. This yields a total number of grid points $p = p_1 p_2$. The discretised field reads:

$$X_{ijk} = X(t_i, \theta_j, \phi_k); \tag{2.32}$$

for $1 \leq i \leq n$, $1 \leq j \leq p_1$, and $1 \leq k \leq p_2$. In order to improve the management of the field, for

applying PCA, the field X is transformed into a two-dimensional array: the data matrix $\mathbf{X}$ where the two spatial dimensions are concatenated together. Figure 2.9 summarizes this process.

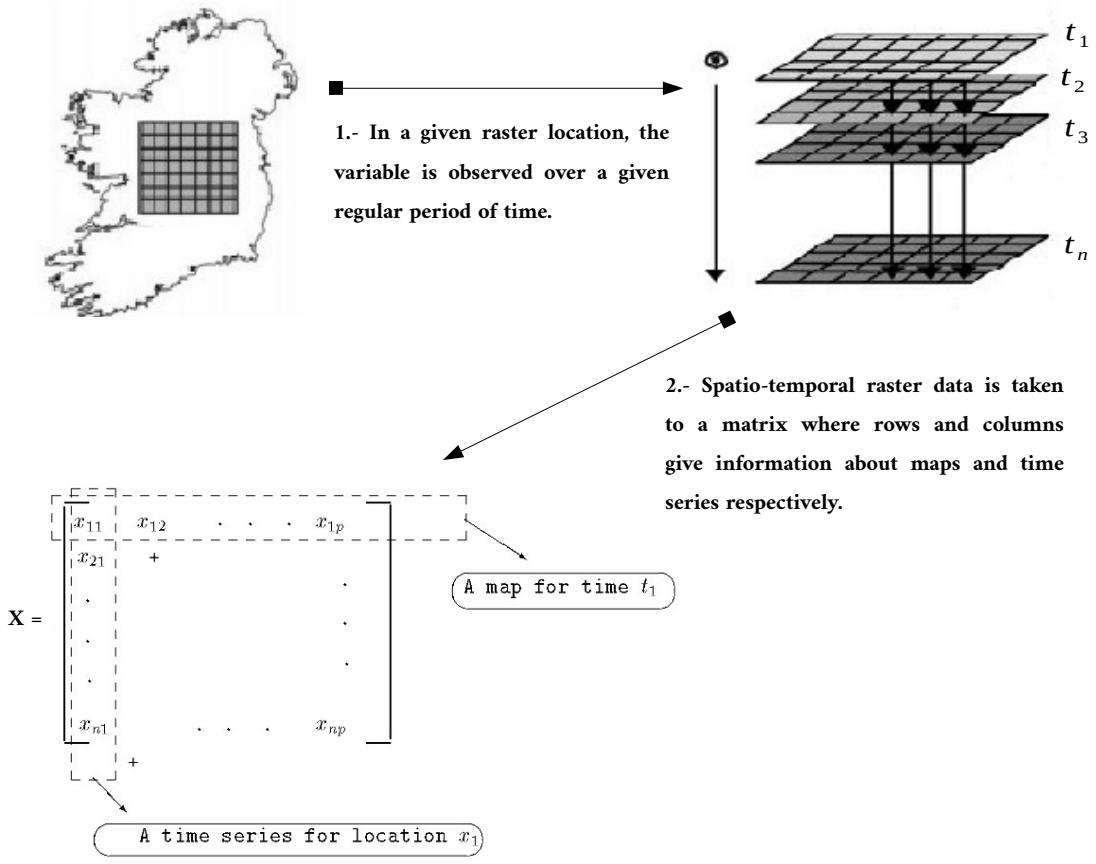

Figure 2.9: Scheme of how the data matrix is obtained when working with a spatio-temporal raster dataset.

When *PCA* is applied to $\mathbf{X}$, the S mode gives as results the spatial patterns of variability, their time variation, and gives a measure of the "importance" of each pattern. It finds spatial patterns of variability because each PC is a linear combination of all locations and a map of its weights can be produced for each PC. Here the calculated component loadings, of the respective PCs, at each sampling location are spatially interpolated to form a contour map, which is then inspected for detecting significant spatial patterns. Next, The PC attached to the corresponding eigenvector provides the sign and the overall amplitude of the component loadings as a function of time. Finally,

the corresponding proportion of variance explained gives a measure of its "importance" as spatial variability pattern.

Remark 2. *By construction, the eigenvectors of the S mode are stationary structures, i. e. they do not evolve in time. The PC attached to the corresponding eigenvector provides a simplified representation of the state of the spatio-temporal field at that time along that eigenvector. In other words, the eigenvector matrix do not change structure in time, they only change sign and overall amplitude to represent the state of the measured variable.*

On the other hand, working in a T mode gives different insight results. In this case, the sampling raster locations are considered as data elements and the sampling times as variables. Compared to the matrix $\mathbf{X}$ of the S mode, the T mode would be equivalent to apply a PCA over $\mathbf{X}'$. As a consequence, the component loadings (eigenvectors) would be time series whereas the PCs would plot up as a geographical map of PC scores. Therefore, it is expected that the T mode *isolates subgroups of time observations with similar spatial patterns* and, thereby, simplify the time series. This is contrary to the S mode that *isolates subgroups of locations which covary similarly.*

To end this discussion, what type of mode of decomposition, S or T, would one use? It would depend on what is the initial goal. For example, if the goal is to get a regionalization of a spatio-temporal dataset, the S mode would be better. On the other hand, if the aim is to summarize the information of the time dimension, the T mode would be chosen.

2.4.4 How many principal components to retain?

Up to now, PCA can be seen as a tool that retains meaningful information in the early axes whereas variation associated to "noise", measurement inaccuracy, is summarized in later axes. PCA has the ability to identify relationships by generating linear combinations of variables showing common trends of variation that can contribute substantially to the recognition of patterns even in datasets with a space-time structure. However, the issue of determining whether or not a given axis summarizes meaningful variation (i.e., non-trivial versus trivial components) remains unclear in many cases.

Going back at theorem 2, equation (2.29) shows that the sample variance corresponding to the

i^{th} component is the i^{th} ordered eigenvalue corresponding to $\mathbf{S}$ (or $\mathbf{R}$). Adding the fact that the sample total variability (trace of sample covariance matrix) fulfills that

$$Sample\ total\ variability = tr(\mathbf{S}) = \sum_{i=1}^{p} \hat{\lambda}_i \ ; \tag{2.33}$$

then, the contribution of variability corresponding to the i^{th} component is given by

$$Proportion\ of\ explained\ variation\ by\ \hat{\mathbf{y}}_i = \left(\frac{\hat{\lambda}_i}{\sum_{i=1}^{p} \hat{\lambda}_i} \right) \times 100\%. \tag{2.34}$$

Equation 2.34 gives a measure of the "importance" of each component. So, is it of interest to know how many PCs to retain? When the correct number of non-trivial PCs is not retained for subsequent analysis, either relevant information is lost (underestimation) or noise is included (overestimation), causing a distortion in underlying patterns of variation/covariation (Ferré, 1995). Therefore, determining the number of non-trivial principal components is very important because it provides a meaningful interpretation of multivariate data.

How many PCs are required to adequately explain variance shared by the variables? A variety of techniques to estimate the number of meaningful components have been proposed (see for example Jackson, 1995; Jolliffe, 2002). For example, there exist empirical tests such as Kaiser's rule (Kaiser, 1960), which takes the number of components that have corresponding eigenvalue greater than +1. Also there exist approaches using simulation such as parallel analysis (Peres-Neto et al., 2005), which replaces the threshold +1 given by the Kaiser's rule with the mean eigenvalues generated from independent normal variates through Monte Carlo simulations. Graphical tests that focus on the elbow of the eigenvalue plot and are not based on a statistical hypothesis test are also found in literature. These tests are known as the Cattell's scree tests (Cattell, 1966). These tests are useful and simple, but they do not enable a clear decision-making about the number of components to retain since the chose depends on the visualization of a graph. To address this limitation, the scree test optimal coordinate and the scree test acceleration factor are proposed by Raîche et al. (2013). These methods consist in numerical solutions to deal with the acceleration of the plot of the eigenvalues and to get formally and explicitly with the scree part of the plot. For purposes of

this work, the scree test optimal coordinate is chosen due to its simplicity and its better results (Raîche et al., 2013).

Scree Test Optimal Coordinate

This test attempts to determine the location of the elbow by measuring the gradients associated with eigenvalues and their preceding coordinates. This strategy is done by computing $p-2$ two-point regression models, and verifying if the observed eigenvalue is, or is not, greater than or equal to the one estimated by these models. The last of these positive verifications, beginning at the second eigenvalue, and without interruption of the verification, is used to determine the number of principal components to retain. In order to make the verifications, two criteria could be taken. First, following the idea of Kaiser's rule, as in equation (2.35), or to the location statistics criteria based on the idea of parallel analysis, as in equation (2.36):

$$n_{oc} = \sum_{i=1}^{p} 1_{\{\lambda_i \geq 1 \ \& \ \lambda_i \geq \hat{\lambda}_i\}}; \tag{2.35}$$

$$n_{oc} = \sum_{i=1}^{p} 1_{\{\lambda_i \geq LS_i \ \& \ \lambda_i \geq \hat{\lambda}_i\}}; \tag{2.36}$$

where n_{oc} is the number of components retained by the optimal coordinate method, LS_i is the location statistic which is usually given by the 95^{th} quantile of a simulation of eigenvalues from independent normal variates, and $\hat{\lambda}_i$ is referred as the the optimal coordinate. This is obtained according to linear regression using only the last eigenvalue and the $(i+1)^{th}$ eigenvalue as follows:

$$\hat{\lambda}_i = a_{i+1} + i \times b_{i+1}; \tag{2.37}$$

where $b_{i+1} = \frac{\lambda_p - \lambda_{i+1}}{p-i-1}$ and $a_{i+1} = \lambda_{i+1} - b_{i+1} \times (i+1)$ for $i = 1, 2, \ldots, p-1$.

2.5 Autoregressive Neural Network model

The aim of time series analysis is to extract information of a given data series, consisting of observations over time. This information is used to build a model of the dynamics, called process, which determines the data series. Such a model can be used for prediction of future values of the time series. For identification of the process, linear models like linear autoregressive processes (AR) and autoregressive moving average processes (ARMA) are a standard tool of econometrics (Box and Jenkins, 1976). However empirical experience shows that linear models are not always the best way to identify a process and do not always deliver the best prediction results. In this context Granger et al. (1993) speak of "hidden nonlinearity", which requires the adoption of nonlinear methods. Since the early 1990's a lot of nonlinear methods have arisen. They can be divided into parametric models, characterized by a fixed number of parameters in a known functional form, and the more general nonparametric models.

The method for nonlinear time series analysis discussed in this section - autoregressive neural network (AR-NN) model - is parametric. As will be seen later, due to its parametric nature and ability to approximate any function and therefore any nonlinearity, the AR-NN model is suitable to analyze time series with non-linear dynamics. Below, subsection 2.5.1 explains the mechanism of artifical neural networks. Then, subsection 2.5.2 deals with the theory of autoregressive processes. Finally, subsections 2.5.3 and 2.5.4 develops the mathematical theory of AR-NN models.

2.5.1 Artificial Neural Networks

Inspired by biological neural networks, Artifical Neural Networks (ANNs) are groups of elementary processing units called artificial neurons connected together to form a directed graph. These elementary processing units are called neurons. Figure 2.10 illustrates a single neuron in a simple neural network model. Nodes of the graph represent biological neurons and connections between them represent synapses. Unlike in biological neural networks, connections between artificial neurons are not usually added or removed after the network was created. Instead, connections are weighted and the weights are adapted by learning algorithms.

In figure 2.10, each of the inputs x_i has a weight w_i that represents the strength of that particular

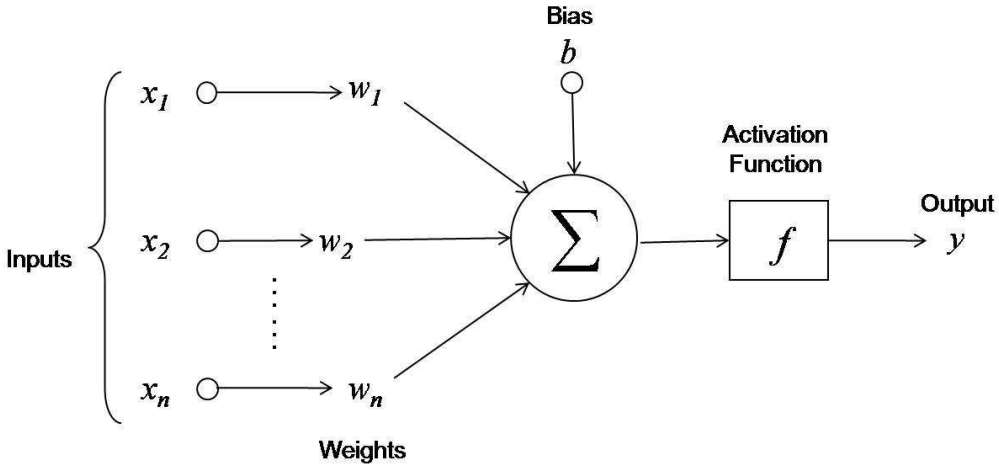

Figure 2.10: Artificial Neuron Model.

connection. The sum of the weighted inputs and the bias b are input to the activation function f to generate the output y. This process can be summarized with the following formula:

$$y = f\left(\sum_{i=1}^{n} w_i x_i + b\right) \tag{2.38}$$

An activation function controls the amplitude of the output of the neuron. Examples of activation functions are:

1. *Linear activation function* $(f(x) = x)$ returns the same number as was fed to it. This is equivalent to having no activation function.

2. *Log-sigmoid activation function* $\left(f(x) = (1 + e^{-x})^{-1}\right)$, sometimes called unipolar sigmoid function, squashes the output to the range between 0 and 1. This function is the most widely used sigmoid function.

3. *Hyperbolic tangent activation function* $\left(f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}\right)$, also called bipolar sigmoid function, is similar to a log-sigmoid function, but it generates outputs between -1 and 1.

4. *Symetric saturating linear function* $\left(f(x) = -\mathbf{1}_{\{x<-1\}} + x\mathbf{1}_{\{-1\leq x\leq 1\}} + \mathbf{1}_{\{x>1\}}\right)$ is a piecewise linear version of sigmoid function which provides outputs between -1 and $+1$.

5. *Hard limit function* $\left(f(x) = \mathbf{1}_{\{x \geq 0\}}\right)$ converts the inputs into 1 if the summed input is bigger than or equal to 0.

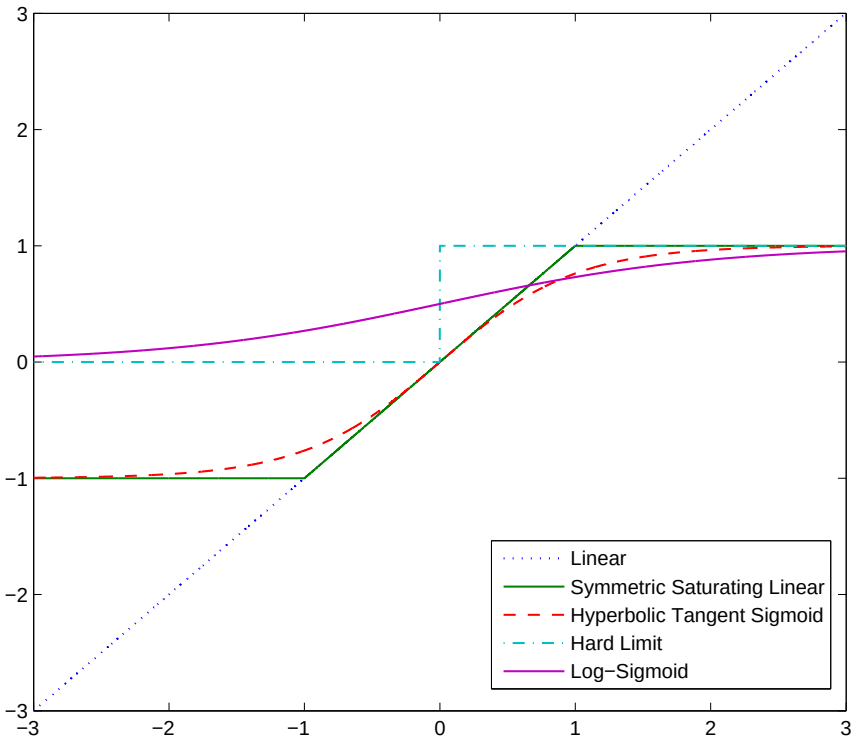

Figure 2.11: Different types of activation functions.

2.5.2 Autoregressive processes

Definition 10. *An autoregressive process (AR, in short) of order p is defined by*

$$x_t = F(\mathbf{X}_{t-1}) + \varepsilon_t; \tag{2.39}$$

where $\mathbf{X}_{t-1} = (x_{t-1}, x_{t-2}, \ldots, x_{t-p})'$, $F : \mathbb{R}^p \to \mathbb{R}$, $\mathbf{X}_{t-1}$ *and* ε_t *are independent and* $\varepsilon_t \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2)$ *(Gaussian white noise assumption).*

From definition 10, the first term on the right hand side of equation (2.39) is called *the predictable part*, and the second term *the stochastic part*. Also if $F(\mathbf{X}_{t-1})$ is a linear function, it is said that

x_t follows a linear AR, but when $F(\mathbf{X}_{t-1})$ is nonlinear, x_t is known as a nonlinear AR.

A linear $AR(p)$ is written as

$$x_t = \phi_0 + \phi_1 x_{t-1} + \phi_2 x_{t-2} + \cdots + \phi_p x_{t-p} + \varepsilon_t. \tag{2.40}$$

Applications show that in most cases the residuals hardly match the Gaussian white noise assumption. A linear solution of this problem are the ARMA processes proposed by Box and Jenkins (1976). They assume that the process does not only consist of a linear predictable part and an additive Gaussian white noise. Rather the stochastic part itself may be determined by a moving average (MA) process of the Gaussian white noise ε_t. So, an $ARMA(p, q)$ process is represented by the following equation (q indicates the maximum lag of the MA part):

$$x_t = \phi_0 + \phi_1 x_{t-1} + \cdots + \phi_{t-p} x_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q}. \tag{2.41}$$

Until today ARMA are the most frequently applied process models in time series analysis. The Wold decomposition theorem (introduced in Wold, 1938) justifies theoretically that one can estimate any covariance stationary process by an ARMA process.

Nonlinear AR models, on its side, try to overcome the problem of observed nonstandard features (hidden nonlinearity) in linear models. For this purpose, neural networks are able to approximate any (not specified) function, linear or not, arbitrary accurately. Next subsection links up the neural networks with a nonlinear AR in a detailed way.

2.5.3 The AR-NN structure

The AR-NN model contains a linear and a nonlinear part. The architecture of an AR-NN model with one hidden layer is shown in figure 2.12. According to the figure, yellow circles represent the input and output neurons (the variables), the black circle is the bias term (the constant of the model) and the red lines represent the shortcut weights (parameters) corresponding to the linear part. Furthermore, the nonlinear part is captured in the *hidden layer* with h hidden neurons represented by h orange diamonds which are responsible for returning a nonlinear transformation

of the weighted input neurons. The blue and green lines represent the weights between the input - hidden and hidden - output neurons respectively.

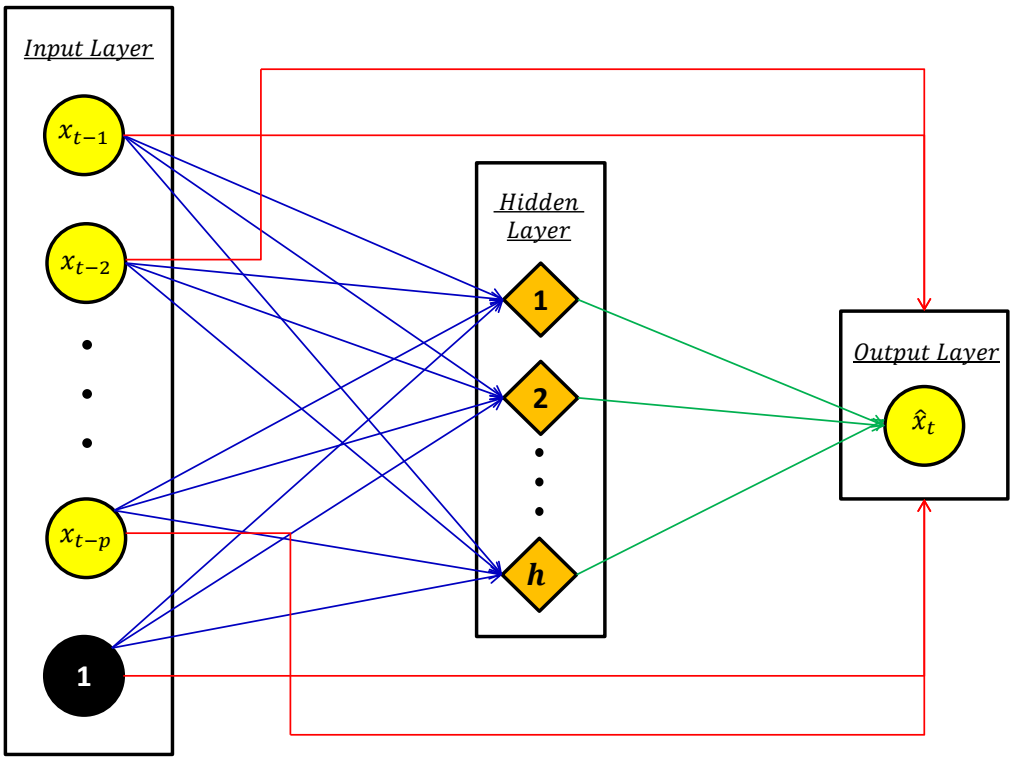

Figure 2.12: Architecture of an AR-NN model with one hidden layer.

Once knowing the architecture of an AR-NN model from graphs, the mathematical representation can be seen as follows:

$$\hat{x}_t = \phi_0 + \sum_{i=1}^{p} \phi_i x_{t-i} + H(\mathbf{X}_{t-1}); \tag{2.42}$$

where $\mathbf{X}_{t-1}$ is the same of equation (2.39) and $H(.)$ is a nonlinear function determined by a nonlinear activation function in the hidden layer and the weights of the input-hidden neurons. So, given the h hidden neurons - which transform the input variables - weighted by parameters η_{ij} plus a bias

η_{0j}, via a nonlinear activation function $\Psi(.)$, then H has the equation:

$$H(\mathbf{X}_{t-1}) = \sum_{j=1}^{h} \beta_j \Psi \left(\eta_{0j} + \sum_{i=1}^{p} \eta_{ij} x_{t-i} \right); \tag{2.43}$$

where β_j's are the weights between the hidden-output neurons. Replacing equation (2.43) in (2.42) and including the stochastic part, the AR-NN model is presented as

$$x_t = \phi_0 + \sum_{i=1}^{p} \phi_i x_{t-i} + \sum_{j=1}^{h} \beta_j \Psi \left(\eta_{0j} + \sum_{i=1}^{p} \eta_{ij} x_{t-i} \right) + \varepsilon_t. \tag{2.44}$$

In the literature (see for example Granger et al., 1993) sometimes neural networks without a linear part can be found. If the process does not show evidence of hidden nonlinearity, a linear AR (or ARMA) model must be preferable.

Particularly for estimation, it makes sense to write equation (2.44) in vector representation with vector input and scalar output. Therefore the following notations are introduced:

$$\Phi = (\phi_1, \phi_2, \ldots, \phi_p)';$$

$$\Gamma_j = (\eta_{1j}, \eta_{2j}, \ldots, \eta_{pj})' \; for \; j = 1, 2, \ldots, h;$$

$$\Theta = (\phi_0, \Phi', \eta_{01}, \Gamma_1', \beta_1, \ldots, \eta_{0h}, \Gamma_h', \beta_h)'.$$

The dimension of Θ is $r \times 1$ with $r = (p+2)h + p + 1$[1]. The first version of the vector representation of equation (2.44) is

$$x_t = \phi_0 + \Phi' \mathbf{X}_{t-1} + \sum_{j=1}^{h} \beta_j \Psi(\eta_{0j} + \Gamma_j' \mathbf{X}_{t-1}) + \varepsilon_t. \tag{2.45}$$

Using Θ the short representation of the $AR - NN$ in equation (2.44) is

$$x_t = G(\Theta, \mathbf{X}_{t-1}) + \varepsilon_t. \tag{2.46}$$

Some considerations concerning the optimal number of hidden neurons: A usual approach is

[1]Note that if the parameter σ^2 is added, then there would be a total of $(p+2)(h+1)$ parameters.

to specify the network using an arbitrary number of hidden neurons and later test the significance of each hidden neuron (Anders, 1997). A common rule of thumb is to set the number of hidden neurons equal to the median of input and output variables (it is $h = (p+1)/2$), (see Anders, 1997, for details). Of course this method does not take account for any technical needs like data specific behavior or the reaction of the activation function on the inputs. A method consistent with the procedure to augment a linear AR for a nonlinear part - if the data are nonlinear - is to extent the number of hidden neurons step by step: At the first step, only one hidden neuron is considered and then hidden neurons are added one by one until reaching the optimal model. White (1992), on its side, says that the number of observations of the input variables should not exceed the number of parameters by the factor 10 (it is $r \geq T/10$, so $h \geq \frac{1}{p+2}(\frac{T}{10}+1)-1)$ to avoid overparametrization. In this thesis, the criterion proposed by White is used due to its simplicity, its power to reduce overparametrization and to keep the computational effort straightforward.

Finally, some considerations about the selection of the activation function are considered. The universal approximation theorem states that any nonlinear function can be approximated by an activation function that is locally Riemann integrable, nonpolynomial and, for a matter of stationarity, bounded. In fact, the universal approximation property does not depend on any certain activation function (Hornik, 1993). Furthermore, the theorem says nothing about the existence of a unique solution of the approximation problem or about the estimation procedures for the neural network (Widmann, 2000). Therefore, is inessential which function is used. However, according to Dutta et al. (2005), sigmoid functions reduce the effect of outliers because they compress the data at the high and low end. For this reason, sigmoid activation functions are chosen for this work.

2.5.4 Modelling univariate AR-NN processes

Only estimating the parameters is certainly not sufficient to receive an appropiate model. The usual steps are the same as Box and Jenkins (1976) proposed: Variable selection, parameter estimation and model validation (parameter tests). As the objective of this work is to develop a computationally efficient learning algorithm, the concentration is focused on the variable selection and the parameter estimation steps.

Variable selection

Given a stationary time series with hidden nonlinearity in at least some lags, the next step is to decide the number of lags and detecting the number of hidden units. The second problem is solved using the White (1992) criterion (see subsection 2.5.3). For the first problem, the general procedure is carried out according to the Occam's razor principle, which meand to prefer the simplest model from a set of models with the same performance. In other words, only those lags and parameters should be included, which significantly improve the model.

In this work, the selection of lag order is detected by calculating information criteria (IC) for several lags and choosing the lag order belonging to the smallest IC. These criterion consider not only the absolute quality of the model (like the variance of the residuals which should be minimized) but also account for the amount of computation effort if the models becomes more complicated. The most common IC is used, the Akaike Information Criterion (AIC), which is defined as

$$AIC = T \cdot log(\hat{\sigma}^2) + 2r; \tag{2.47}$$

where T is the length of the time series, $\hat{\sigma}^2 = \sum_{i=1}^{T} \frac{\hat{\varepsilon}_t^2}{T-r}$ and $r = (p+2)(h+1) - 1$ with p as the number of lags and h as the number of hidden neurons. In order to adapt this strategy in a learning algorithm, a restriction like fixing the number of maximum evaluated lags is added.

Parameter Estimation

The most important step to concretize the AR-NN is the estimation of the weights (parameters). This equals the estimation of the parameters in linear time series analysis and is called *learning or training* in neural network theory. Many procedures exist to estimate the parameters of neural networks. Thus it has to be distinguished between supervised learning methods and unsupervised methods (Haykin et al., 2009). Supervised methods means that the estimation output is compared to a desired output and estimation takes the error signal into account. The error signal is defined as the difference between estimation and desired output. Unsupervised methods use no criteria to control the learning process, so they seem not applicable to statistics and especially time series analysis. Hence in the following the atention is concentrated on supervised learning procedures

only.

In general it can be said that there are two different classes of supervised learning procedures concerning the estimation of the parameters (Haykin et al., 2009). *Batch learning* is an iterative procedure where the weights are adjusted in each iteration after the presentation of all T inputs, while during *on - line learning* (sometimes referred to as stochastic learning) the weights are adjusted on element-by-element basis. This means that for each set of input and output neurons from 1 to T the weights are newly adjusted. The AR-NN process is estimated for the inputs and outputs at a certain time t only, for time $t+1$ the weights are adjusted again. The main advantages of on-line over batch learning methods are the lower computational complexity and the better adaptability to integrate new values if data arrive sequentially, but concerning the precision of the results it performs poorly (Bottou, 2003). Furthermore, on-line algorithms do not really converge because they are adjusted after each new input of a variable set. On the other hand, batch learning algorithms give accurate estimations of the gradient vector for a finite input dataset, which guarantees the convergence of the algorithms. Hence, due to its efficiency from a statistical point of view the batch learning algorithms are chosen in this thesis and are explained below.

To apply a batch algorithm, a *performance function* must be determined first. Like in the well known least square procedures the goodness of fit of an AR-NN model , and one possible performance function, can be determined by

$$Q(\Theta) = \frac{1}{2}\sum_{t=1}^{T}\left(x_t - G(\Theta, \mathbf{X}_{t-1})\right)^2; \tag{2.48}$$

where $G(\Theta, \mathbf{X}_{t-1})$ is the same as in equation (2.46). If this performance function is used, the parameter estimation procedures in the following are referred to as nonlinear least squares (NLS) method in literature and are not restricted to neural network only. Of course it is possible to use other performance functions like the likelihood function (Anders, 1997), but they are less common.

As the performance function should also be valid for future values of the time series, the expectation of function 2.48 has to be minimized. Therefore an optimal nonlinear least squares estimator

for Θ, $\hat{\Theta}$, can be found by solving the problem

$$\hat{\Theta} = \underset{\Theta \in \mathbf{\Theta}}{arg\,min}\, \mathbb{E}(Q(\Theta)); \tag{2.49}$$

where $\mathbf{\Theta}$ denotes the network weight space.

Theorem 3. *Let $Q(\Theta)$ be as in equation (2.48), $F(\mathbf{X}_{t-1})$ as in equation (2.39) and $G(\Theta, \mathbf{X}_{t-1})$ as in equation (2.46), then*

$$\mathbb{E}(Q(\Theta)) = \frac{1}{2}\mathbb{E}[\sum_{t=1}^{T} \varepsilon_t^2] + \frac{1}{2}\mathbb{E}[\sum_{t=1}^{T} (F(\mathbf{X}_{t-1}) - G(\Theta, \mathbf{X}_{t-1}))^2]; \tag{2.50}$$

where ε_t is the corresponding Gaussian white noise of the model 2.39.

Proof.

$$\mathbb{E}(Q(\Theta)) = \frac{1}{2}\mathbb{E}[\sum_{t=1}^{T} (x_t - G(\Theta, \mathbf{X}_{t-1}))^2] = \frac{1}{2}\mathbb{E}[\sum_{t=1}^{T} (x_t - F(\mathbf{X}_{t-1}) + F(\mathbf{X}_{t-1}) - G(\Theta, \mathbf{X}_{t-1}))^2]$$

$$= \frac{1}{2}\mathbb{E}[\sum_{t=1}^{T} (x_t - F(\mathbf{X}_{t-1}))^2] + \mathbb{E}[\sum_{t=1}^{T} (x_t - F(\mathbf{X}_{t-1}))(F(\mathbf{X}_{t-1}) - G(\Theta, \mathbf{X}_{t-1}))] +$$

$$\frac{1}{2}\mathbb{E}[\sum_{t=1}^{T} (F(\mathbf{X}_{t-1}) - G(\Theta, \mathbf{X}_{t-1}))^2]$$

Due to equation (2.39), $(x_t - F(\mathbf{X}_{t-1})) = \varepsilon_t$. It remains to show that $\mathbb{E}[\sum_{t=1}^{T} (x_t - F(\mathbf{X}_{t-1}))(F(\mathbf{X}_{t-1}) - G(\Theta, \mathbf{X}_{t-1}))] = 0$. Indeed,

$$\mathbb{E}[\sum_{t=1}^{T} (x_t - F(\mathbf{X}_{t-1}))(F(\mathbf{X}_{t-1}) - G(\Theta, \mathbf{X}_{t-1}))] = \sum_{t=1}^{T} \mathbb{E}[(x_t - F(\mathbf{X}_{t-1}))(F(\mathbf{X}_{t-1}) - G(\Theta, \mathbf{X}_{t-1}))]$$

$$= \sum_{t=1}^{T} \mathbb{E}[\mathbb{E}[(x_t - F(\mathbf{X}_{t-1}))(F(\mathbf{X}_{t-1}) - G(\Theta, \mathbf{X}_{t-1}))|\mathbf{X}_{t-1}]]$$

$$= \sum_{t=1}^{T} \mathbb{E}[\mathbb{E}[(x_t - F(\mathbf{X}_{t-1}))|\mathbf{X}_{t-1}](F(\mathbf{X}_{t-1}) - G(\Theta, \mathbf{X}_{t-1}))]$$

$$= \sum_{t=1}^{T} \mathbb{E}[\mathbb{E}[\varepsilon_t | \mathbf{X}_{t-1}](F(\mathbf{X}_{t-1}) - G(\Theta, \mathbf{X}_{t-1}))]$$

Since $\mathbb{E}(\varepsilon_t | \mathbf{X}_{t-1}) = 0$, the proof is completed. □

Theorem 3 says important features about $\hat{\Theta}$. The first term of equation (2.50) states that $\hat{\Theta}$ minimizes the errors of the stochastic part. The second term states that, on order to find $\hat{\Theta}$, a minimum is reached if $G(\Theta, \mathbf{X}_{t-1}) = F(\mathbf{X}_{t-1})$. In this case, equation (2.48) reduces to

$$Q(\Theta) = \frac{1}{2} \sum_{t=1}^{T} \varepsilon_t^2; \tag{2.51}$$

where ε_t is the same as equation (2.39). Moreover, Note that ε_t can be seen as a function of Θ because

$$\varepsilon_t = \varepsilon_t(\Theta) = (x_t - G(\Theta, X_{t-1})). \tag{2.52}$$

Given a performance function, all numeric parameter estimation algorithms for neural networks work the same way: Starting with a random initial parameter vector Θ_0 and then iteratively finding the optimal parameter vector by minimizing the performance function. However, not always the global minimum is reached. In fact, the batch learning algorithms only lead to local minima. In addition the choice of the initial weight vector Θ_0 influences the outcome of the algorithm.

In general, the algorithm is carried out according to the flow chart in figure 2.13. The weights are updated after each iteration and the performance function is calculated. If a stopping criterion is reached, the algorithm is quitted. The stopping criterion can be a restriction concerning the performance function, for example the difference between the value of performance function in two consecutive iterations should be below a certain value (Anders, 1997). Alsa the maximal number of iterations, i_{max} can be used as a stopping criterion and the optimal parameter vector is calculated by the following steps:

1. `Start the algorithm with the initial weight vector` Θ^0.
2. `After each iteration, save` $Q(\Theta_i)$ `and` Θ_i.
3. `Quit the algorithm after` i_{max} `iterations.`

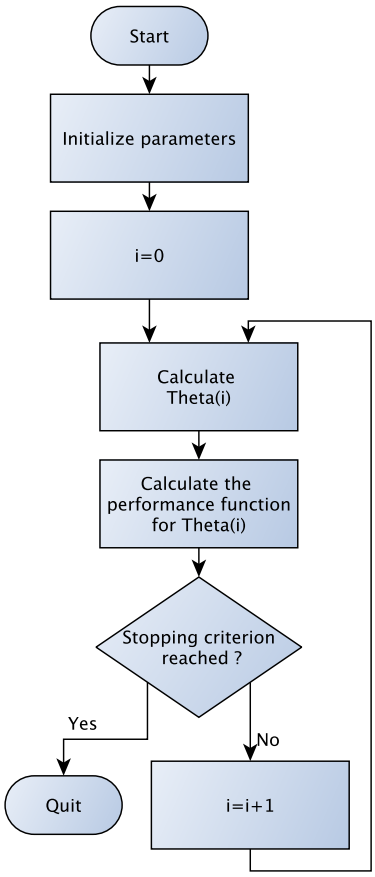

Figure 2.13: Flow chart of the iterative parameter estimation process.

4. `Among the saved values, search` i^* such that

$$Q(\Theta_{i^*}) = \underset{i \in \{0,1,\ldots,i_{max}\}}{arg\,min} \; Q(\Theta_i).$$

5. Θ_{i^*} `is the optimal parameter vector.`

Such a procedure can be interpreted as search for a global minimum within a finite horizon of iterations. Often the performance function converges to a certain constant within a finite number of iterations. Therefore a good local minimum within a finite number of iterations, i_{max}, often is in fact a global minimum. In order to complete the algorithm the method is adapted with the Levenberg-Marquardt algorithm.

The Levenberg-Marquardt algorithm is considered as one of the most powerful learning methods for neural networks, especially when the performance functions are of the error-sum-of-squares form (Bishop, 1995). The algorithm combines the steepest descent algorithm of Rumelhart et al. (1985) and Newton's method. As a consequence, the advantages of this method are that it converges rapidly like Newton's method and it can not diverge because of the steepest descent algorithm influence (Haykin et al., 2009). For these reasons, this algorithm is used for the experimental and empirical results of this work.

The algorithm works by looking for a suitable representation of the Hessian matrix if the performance function $\nabla^2 Q(\Theta) = \left(\frac{\partial^2 Q(\Theta)}{\partial\theta_i \partial\theta_j}\right)_{i,j=1,\ldots,r}$. In that way, it is possible to show that $\nabla^2 Q(\Theta)$ can be estimated by the cross product of the Jacobian matrices of $\varepsilon_t(\Theta)$ (see equation (2.52)), $J(\varepsilon_t(\Theta)) = \left(\frac{\partial \varepsilon_t(\Theta)}{\partial\theta_i}\right)_{\substack{t=1,\ldots,T \\ i=1,\ldots,r}}$. Indeed

$$\nabla^2 Q(\Theta) = \nabla^2 \left(\frac{1}{2}\sum_{t=1}^{T} \varepsilon_t(\Theta)^2\right) = \nabla(J(\varepsilon_t(\Theta))' E); \tag{2.53}$$

where $E = (\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_T)'$. On element by element basis, for the i^{th} row of equation (2.53) and the j^{th} weight, for $i, j = 1, \ldots, r$, by using the product rule of differentiation:

$$\frac{\partial\left(\sum_{t=1}^{T} \varepsilon_t \frac{\partial \varepsilon_t}{\partial\theta_i}\right)}{\partial\theta_j} = \sum_{t=1}^{T}\left(\frac{\partial\varepsilon_t}{\partial\theta_j}\frac{\partial\varepsilon_t}{\partial\theta_i} + \varepsilon_t \frac{\partial^2\varepsilon_t}{\partial\theta_j\partial\theta_i}\right). \tag{2.54}$$

The second term on the right hand of equation (2.54) is approximately zero (see Bishop (1995)). Due to this fact, is showed that:

$$\nabla^2 Q(\theta) \approx J(\varepsilon_t(\Theta))' J(\varepsilon_t(\Theta)). \tag{2.55}$$

According to the second order gradient descent method (a Newton's method), it consider the next learning algorithm:

$$\Theta_{i+1} = \Theta_i - (\nabla^2 Q(\Theta_i))^{-1} \nabla Q(\Theta_i). \tag{2.56}$$

Replacing equation (2.55) and the fact that $\nabla Q(\Theta) = J(\varepsilon_t(\Theta))'E$ in equation (2.56):

$$\Theta_{i+1} = \Theta_i - [J(\varepsilon_t(\Theta))'J(\varepsilon_t(\Theta))]^{-1}J(\varepsilon_t(\Theta))'E_i. \tag{2.57}$$

The fact that the pure crossproduct of the Jacobian matrices sometimes leads to singularities, as application shows, might be problematic. However, this problem can be reduced by adding a positive definite symetric matrix to the Hessian matrix which includes the identity matrix $\mathbb{I}_r$ and a sufficient large parameter λ. This extra term for the Hessian matrix of $Q(\Theta)$ gives the final representation of the Levenberg - Marquardt algorithm as follows

$$\Theta_{i+1} = \Theta_i - [J(\varepsilon_t(\Theta))'J(\varepsilon_t(\Theta)) + \lambda\mathbb{I}_r]^{-1}J(\varepsilon_t(\Theta))'E_i. \tag{2.58}$$

λ is multiplied by a factor τ if an iteration results in an increased of $Q(\Theta)$. If an interation reduces $Q(\Theta)$, λ is divided by τ. For computational reasons the parameter λ should at least be different from zero such that the matrix $[J(\varepsilon_t(\Theta))'J(\varepsilon_t(\Theta)) + \lambda\mathbb{I}_r]$ is positive definite (Haykin et al., 2009). The flowchart in figure 2.14 explains how the algorithm runs.

2.6 Wavelet decomposition of discrete time series

Wavelets can be casually described as oscillatory basis functions, cleverly constructed to posses several attractive features not enjoyed by "big waves" (sines and cosines); for example multiscale structure, ability to represent a variety of functions in a sparse manner, or simultaneous localisation in time and frequency. Since their invention in the early eighties; wavelets have received enormous attention both in mathematical community (Hernández and Weiss, 1996) and in applied sciences (Jaffard et al., 2001). Particularly, in the statistics field, wavelets are used to analyze continuous and discrete time series in order to get information about the behavior of its corresponding high and low frequency signals. Nowadays, wavelets could be also applied within a stage of modeling by using wavelet decomposition method in discrete time series. Basically, the wavelet decomposition uses a pair of filters to decompose iteratively the original discrete time series. It results in subcomponents of the original time series at different frequency bands that are easier to model and predict. Next

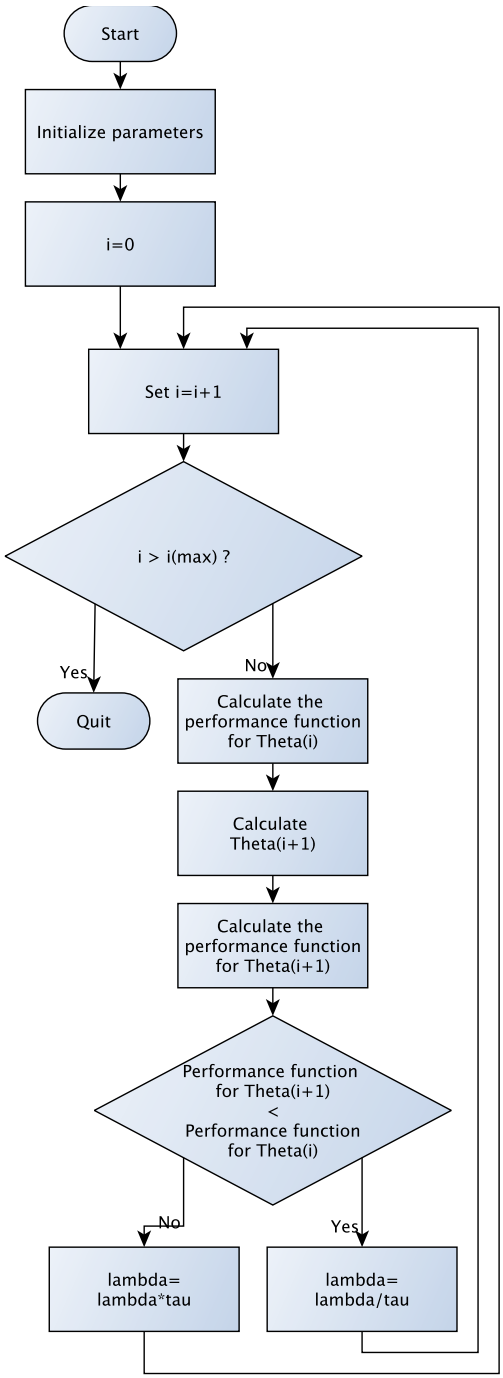

Figure 2.14: Flow chart of the Levenberg-Marquardt algorithm.

subsections formalize the concept of wavelets until arriving to the wavelet decomposition process for discrete time series.

2.6.1 Wavelets

In order to formalize wavelets, some previous notations are necessary. Given $f, g : \mathbb{R} \to \mathbb{C}$, the inner product of functions f and g is defined as

$$< f, g >= \int_{\mathbb{R}} f\bar{g} d\lambda;$$

where the integral is taken using the traditional λ - Lebesgue measure. This inner product induces naturally the norm $||f||_2 =< f, f >^{1/2}$. Then, the $\mathbb{L}^2(\Omega)$ space is defined as

$$\mathbb{L}^2(\Omega) = \{f : \Omega \to \mathbb{C} / \int_{\Omega} ||f||_2 d\lambda < +\infty\}. \tag{2.59}$$

Definition 11. *$\{f_n\}_{n\in\mathbb{Z}}$ is an orthonormal basis for $\mathbb{L}^2(\mathbb{R})$ if $\{f_n\}_{n\in\mathbb{Z}}$ is an orthonormal system; i.e. $< f_i, f_j >= \delta_{ij}$ where δ_{ij} is the Kronecker delta function for $i, j \in \mathbb{Z}$, and for every $f \in \mathbb{L}^2(\mathbb{R})$,*

$$f = \sum_{k\in\mathbb{Z}} c_k f_k \tag{2.60}$$

Now it is possible to give a formal definition of a wavelet. It is important to emphasize that wavelets can be defined in multiple ways (see e.g., Hernández and Weiss, 1996). Definition given by Haar (1910) is used because it fits better with our application purposes.

Definition 12. *A function $\psi \in \mathbb{L}^2(\mathbb{R})$ is an orthonormal wavelet (or simply a wavelet) provided the system $\{\psi_{j,k} : j, k \in \mathbb{Z}\}$ is an orthonormal basis for $\mathbb{L}^2(\mathbb{R})$, where*

$$\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k) \quad for\ all \quad j, k\ in\ \mathbb{Z} \tag{2.61}$$

Definition 12 says a wavelet is a function whose dyadic dilations and translations form an orthonomal basis of $\mathbb{L}^2(\mathbb{R})$. Indices j and k are commonly called scale (or dilation) and location (or translation) parameters, respectively. Next is detailed how a wavelet could be obtained.

2.6.2 Multiresolution Analysis

The *multiresolution analysis* (Mallat, 1989) is used for two reasons. From an applied approach, it allows to give an appropriate decomposition of high and low frequency signals to discrete time series. From a theoretical approach, it permits to find wavelets according to definition 12. The applied approach will be discussed in subsection 2.4.4 and here is discussed briefly the theoretical use.

Definition 13. *A multiresolution analysis (MRA) of* $\mathbb{L}^2(\mathbb{R})$ *consists of a sequence of closed subspaces* V_j, $j \in \mathbb{Z}$, *de* $L^2(\mathbb{R})$ *satisfying the following conditions:*

- *C1:* $V_j \subset V_{j+1}$ *for all* $j \in \mathbb{Z}$.
- *C2:* $f \in V_j$ *if and only if* $f(2(.)) \in V_{j+1}$ *for all* $j \in \mathbb{Z}$.
- *C3:* $\bigcap_{j\in\mathbb{Z}} V_j = \{0\}$.
- *C4:* $\overline{\bigcup_{j\in\mathbb{Z}} V_j} = \mathbb{L}^2(\mathbb{R})$.
- *C5:* $\exists\varphi \in V_0$ *such that* $\{\varphi(. - k) : k \in \mathbb{Z}\}$ *is an orthonormal basis for* V_0.

The function φ *of condition C5 is called a scaling function of the given MRA.*

Let $\varphi_{j,k}(x) = 2^{j/2}\varphi(2^j x - k)$; since $\varphi_{0,k} = \varphi(x-k)$ then $\varphi_{0,k} \in V_0$ for all $k \in \mathbb{Z}$ due to condition *C5*. Moreover, if $j \in \mathbb{Z}$, condition *C2* implies that $\{\varphi_{j,n} : n \in \mathbb{Z}\}$ is an orthonormal basis for V_j. Taking this fact, how to build an orthonormal wavelet from a MRA?

Let W_0 be the orthogonal complement of V_0 in V_1; i.e., $V_1 = V_0 \oplus W_0$. Then, by dilating the elements of W_0 by 2^j, a closed subspace W_j of V_{j+1} is obtained such that

$$V_{j+1} = V_j \oplus W_j \quad for\ each\ j \in \mathbb{Z}. \tag{2.62}$$

Since $V_j \to \{0\}$ as $j \to -\infty$, it follows that

$$V_{j+1} = V_j \oplus W_j = \bigoplus_{l=-\infty}^{j} W_l \quad for\ all\ j \in \mathbb{Z}. \tag{2.63}$$

Since $V_j \to \mathbb{L}^2(\mathbb{R})$ as $j \to \infty$, then

$$\mathbb{L}^2(\mathbb{R}) = \bigoplus_{j=-\infty}^{\infty} W_j. \tag{2.64}$$

To find an orthonormal wavelet, therefore, only is needed to find a function $\psi \in W_0$ such that $\{\psi(. - k) : k \in \mathbb{Z}\}$ is an orthonormal basis for W_0. In fact, if this is the case, then $\{2^{j/2}\psi(2^j. - k) : k \in \mathbb{Z}\}$ is an orthonormal basis for W_j for all $j \in \mathbb{Z}$ due to condition *C2* and the definition of W_j. Hence $\{\psi_{j,k} : j, k \in \mathbb{Z}\}$ is an orthonormal basis for $\mathbb{L}^2(\mathbb{R})$, which shows that ψ complies definition 12. So, how to find such a function ψ?.

Consider $V_0 = W_{-1} \oplus V_{-1}$ and observe that $\frac{1}{2}\varphi(\frac{\cdot}{2}) \in V_{-1} \subset V_0$. By condition *C5*, this function can be expressed in terms of the basis $\{\varphi(. + k) : k \in \mathbb{Z}\}$ to obtain

$$\frac{1}{2}\varphi(\frac{1}{2}x) = \sum_{k\in\mathbb{Z}} \alpha_k \varphi(x + k); \tag{2.65}$$

where $\alpha_k = \frac{1}{2}\int_{\mathbb{R}} \varphi(\frac{1}{2}x)\overline{\varphi(x + k)}dx$; the convergence in (2.65) is in $\mathbb{L}^2(\mathbb{R})$ and $\sum_{k\in\mathbb{Z}} \|\alpha_k\|^2 < \infty$. Taking Fourier transforms, it follows

$$\hat{\varphi}(2\eta) = \hat{\varphi}(\eta) \sum_{k\in\mathbb{Z}} \alpha_k e^{ik\eta} = \hat{\varphi}(\eta) m_0(\eta); \tag{2.66}$$

where $i = \sqrt[2]{-1}$, $\hat{\varphi}$ is the Fourier transform of φ given by

$$\hat{f}(\omega) = \int_{-\infty}^{\infty} f(t)e^{-iwt}dt; \tag{2.67}$$

and

$$m_0(\eta) = \sum_{k\in\mathbb{Z}} \alpha_k e^{ik\eta} \tag{2.68}$$

is a 2π-periodic function in $\mathbb{L}^2([-\pi, \pi))$. The function m_0 is called the *low pass filter* associated with the scaling function φ. The next proposition finally gives a characterization of ψ.

Proposition 1. *Suppose φ is a scaling function for an MRA $\{V_j\}_{j\in\mathbb{Z}}$, and m_0 is the associated*

low-pass filter; then a function $\psi \in W_0 = V_1 \cap V_0^{\perp}$ *is an orthonormal wavelet for* $\mathbb{L}^2(\mathbb{R})$ *if and only if*

$$\hat{\psi}(2\eta) = e^{i\eta} v(2\eta) \overline{m_0(\eta + \pi)} \hat{\varphi}(\eta) \quad a.e.\ on\ \mathbb{R}; \tag{2.69}$$

for some 2π *- periodic measurable function* v *such that*

$$\|v(\eta)\| = 1 \quad a.e.\ on\ [-\pi, \pi)$$

Proof. For the proof see Hernández and Weiss (1996). □

For simplicity, consider $v(\eta) = 1$ and ψ of equation (2.69). ψ is a wavelet if and only if

$$\hat{\psi}(2\eta) = e^{i\eta} \overline{m_0(\eta + \pi)} \hat{\varphi}(\eta). \tag{2.70}$$

By replacing $m_0(\eta)$, equation (2.68), and α_k, equation (2.65), it follows that

$$\hat{\psi}(\eta) = \left(\sum_{k \in \mathbb{Z}} (-1)^k \overline{\alpha_k} e^{-i(k-1)\frac{\eta}{2}} \right) \hat{\varphi}(\frac{1}{2}\eta). \tag{2.71}$$

Taking the inverse Fourier transform this finally gives

$$\psi(x) = 2 \sum_{k \in \mathbb{Z}} (-1)^k \overline{\alpha_k} \varphi(2x - (k-1)). \tag{2.72}$$

Two remarks must be taken into account. First, it is possible to find a wavelet that not necessarily comes from a MRA (for examples, see Hernández and Weiss, 1996). However, these kind of wavelets are in general "rare" and they are not useful for applications. Second, any wavelet of compact support comes from a MRA (for a proof see Hernández and Weiss, 1996). Compact support results in a finite multiplication in the discrete wavelet transform (see subsection 2.6.3), what consequently yields to simple practical implementation of wavelet analysis. Since there is no need to approximate the wavelet function (to have a compact support), all computations are exact. For this reason, in practical applications it is usual to use compactly supported wavelets satisfying

the next *admissibility conditions*[2]:

$$\int_{-\infty}^{\infty} \frac{|\hat{\psi}(\omega)|^2}{|\omega|} d\omega < \infty. \tag{2.73}$$

It could be proved that equation (2.73) implies that

$$\int_{-\infty}^{\infty} \psi(x) dx = 0 \tag{2.74}$$

It means, while the admissibility condition can be viewed as a requeriment that ψ should be localised in frequency, equation (2.74) can be interpreted as both, localisation in time (as it implies $\psi \in \mathbb{L}^1(\mathbb{R})$) and oscillation.

Definition 14. *The wavelet ψ has p vanishing moments if*

$$\int_{-\infty}^{\infty} x^k \psi(x) dx = 0 \quad for \quad k \in \{0, 1, \ldots, p\}. \tag{2.75}$$

The vanishing moments property (definition 14), together with localisation properties (equation (2.73) and (2.74)), implies that wavelets are often capable of representing signals in a sparse manner. Next subsection deals with the traditional method to decompose discrete time series in its different frequency signals, the discrete wavelet transform.

2.6.3 Discrete Wavelet Transform (DWT)

Before further explanation of the *Discrete Wavelet Transform* (DWT), some important preliminaries are necessary. See Frazier (2006) for further details on any of them. First, in favor to work with discrete time series of length N, it is essential to give them a formal metric space. This space is given by

$$\ell^2(\mathbb{Z}_N) = \{(z[0], z[1], \ldots, z[N-1]) : z[j] \in \mathbb{R}, 0 \leq j \leq N-1\}. \tag{2.76}$$

$\ell^2(\mathbb{Z}_N)$ has the usual componentwise addition and scalar multiplication over $\mathbb{R}$, so $l^2(\mathbb{Z}_N)$ is an

[2]This admissibility condition is an alternative definition for a wavelet given by Daubechies (1992)

N-dimensional vector space over $\mathbb{R}$. The inner product on $\ell^2(\mathbb{Z}_N)$ is

$$\langle z, w \rangle = \sum_{k=0}^{N-1} z[k]w[k] \;\; \forall z, k \in \ell^2(\mathbb{Z}_N) \tag{2.77}$$

with the associated norm

$$\|z\|_2 = \left(\sum_{k=0}^{N-1} z^2[k] \right)^{1/2} \quad \forall z \in \ell^2(\mathbb{Z}_N). \tag{2.78}$$

Observe that since $\ell^2(\mathbb{Z}_N)$ has a well defined inner product and $\|z\|_2, \infty, \forall z \in \ell^2(\mathbb{Z}_N)$, then $\ell^2(\mathbb{Z}_N)$ is a Hilbert space, i.e. $\ell^2(\mathbb{Z}_N)$ is a real inner product space that is also a complete metric space.

For the sake of simple notation, it is worth to make a convention by defining a periodic extension of a vector to be defined at all integers. To do this, define the periodic extension of the vector $z = (z[0], z[1], \ldots, z[N-1]) \in \ell^2(\mathbb{Z}_N)$ as $z[t+N] = z[t] \forall t \in \mathbb{Z}$.

Definition 15. *Suppose $z \in \ell^2(\mathbb{Z}_N)$. Then, $\hat{z}$ is the Discrete Fourier Transform (DFT) of z if each component of $\hat{z}$ is given by*

$$\hat{z}[k] = \sum_{t=0}^{N-1} z[t] e^{-2\pi i k t/N} \;\; for\ k = 0, 1, \ldots, N-1; \tag{2.79}$$

where $i = \sqrt{-1}$.

Furthermore, some additional operators are defined.

Definition 16. *Suppose $z \in \ell^p(\mathbb{Z}_N)$ and $k \in \mathbb{Z}$. Then R_k is the translation by k operator if*

$$R_k z[t] = z[t-k] \; for\ t \in \mathbb{Z}. \tag{2.80}$$

$R_k z$ is called translate of z by k.

Definition 17. *For $z, w \in \ell^2(\mathbb{Z}_N)$, the convolution $z * w \in \ell^2(\mathbb{Z}_N)$ is the vector with components*

$$z * w[k] = (z * w)[k] = \sum_{t=0}^{N-1} z[k-t]w[t], \ \forall k. \tag{2.81}$$

Definition 18. *For any $z \in \ell^2(\mathbb{Z}_N)$, $\tilde{z}$ is called the reflection of z if*

$$\tilde{z}[t] = z[-t] = z[N-t], \forall t. \tag{2.82}$$

Given these definitions, two important properties are met

$$z * w[k] = \langle z, R_k \tilde{w} \rangle; \tag{2.83}$$

$$z * \tilde{w}[k] = \langle z, R_k w \rangle. \tag{2.84}$$

The main idea of the DWT is to decompose the original Hilbert space $\ell^2(\mathbb{Z}_N)$ into two subspaces, a space of approximations $\mathcal{V}$ and a space of details $\mathcal{W}$. This is known as analysis at the 1^{st} level. Since the Hilbert space of approximations is closed and has countable basis, it is separable and can be decomposed further (Frazier, 2006). In general, this is known as analysis at the J^{th} level. Let $\mathcal{V}_j$ and $\mathcal{W}_j$ denote space of approximations and details at the j^{th} level, respectively. Then, from the MRA and without loss of generality

$$\begin{aligned} \ell^2(\mathbb{Z}_N) &= \mathcal{V}_1 \oplus \mathcal{W}_1, \\ \mathcal{V}_1 &= \mathcal{V}_2 \oplus \mathcal{W}_2, \\ &\vdots \\ \mathcal{V}_{J-1} &= \mathcal{V}_J \oplus \mathcal{W}_J; \end{aligned}$$

or in more compact form

$$\ell^2(\mathbb{Z}_N) = \mathcal{V}_J \oplus \mathcal{W}_J \oplus \mathcal{W}_{J-1} \oplus \cdots \oplus \mathcal{W}_1. \tag{2.85}$$

Both subspaces $\mathcal{V}_j$ and $\mathcal{W}_j$ for some j have the same dimensions, what consequently means that spaces $\mathcal{V}_j$ and $\mathcal{W}_j$ have dimension $N/2^j$. That is the reason why N has to be dividable by 2^J, where J is the maximum level of analysis. Note that the nomenclature *space of approximations* and *space of details* has its origin in two essential wavelet properties. The first of them is the possibility to construct the best approximation of any vector by orthogonal projection (what the DWT does) and the second is, concerning the wavelets on a particular level of analysis, mutual exclusivity of wavelets in a frequency domain (Kölzow, 1994).

Now, given the necessary definitions a brief sketch of the DWT theory is introduced. Full *DWT* theory can be found in Frazier (2006) and Kölzow (1994), together with proofs of propositions stated here. Let $N = 2M$ and $\varphi, \psi \in \ell^2(\mathbb{Z}_N)$. Then the set $B = \{R_{2k}\varphi\}_{k=0}^{M-1} \cup \{R_{2k}\psi\}_{k=0}^{M-1}$ is an orthonormal basis in $\ell^2(\mathbb{Z}_N)$ if and only if the matrix

$$A[n] = \frac{1}{\sqrt{2}} \begin{bmatrix} \hat{\varphi}[n] & \hat{\psi}[n] \\ \hat{\varphi}[n+M] & \hat{\psi}[n+M] \end{bmatrix} \tag{2.86}$$

is orthogonal for all $n = 0, 1, \ldots, N-1$. Such an orthonormal basis B is called wavelet basis at the 1^{st} level, and vectors φ, ψ are called its generators. Vector φ is called a *father wavelet* and vector ψ is called a *mother wavelet*. Furthermore, let φ be a vector such that the set $\{R_{2k}\varphi\}_{k=0}^{M-1}$ is orthonormal. Thereafter, ψ could be constructed as:

$$\psi[n] = (-1)^n \varphi[N+1-n] \quad for\ n = 1, 2, \ldots, N; \tag{2.87}$$

and then $B = \{R_{2k}\varphi\}_{k=0}^{M-1} \cup \{R_{2k}\psi\}_{k=0}^{M-1}$ is a wavelet basis at the 1^{st} level for $\ell^2(\mathbb{Z}_N)$. So, given a finite discrete time series $z \in \ell^2(\mathbb{Z}_N)$, its coefficients in the basis B can be expressed as the inner products of z with the basis vectors. This is

$$[z]_B = (z * \tilde{\varphi}[0], z * \tilde{\varphi}[2], \ldots, z * \tilde{\varphi}[N-2], z * \tilde{\psi}[0], z * \tilde{\psi}[2], \ldots, z * \tilde{\psi}[N-2]). \tag{2.88}$$

Definition 19. *Let $z \in \ell^2(\mathbb{Z}_N)$, $w \in \ell^2(\mathbb{Z}_M)$ with $N = 2M$. Define a downsampling operator*

$D : \ell^2(\mathbb{Z}_N) \to \ell^2(\mathbb{Z}_M)$ *where*

$$D(z)[t] = z[2t] \quad for\ t = 0, 1, \ldots, M-1; \tag{2.89}$$

and an upsampling operator $U : \ell^2(\mathbb{Z}_M) \to \ell^2(\mathbb{Z}_N)$ *where*

$$U(w)[t] = \begin{cases} w[t/2]\,, if\ t\ is\ even \\ 0 \qquad\ \ , if\ t\ is\ odd \end{cases} \quad for\ t = 0, 1, \ldots, N-1. \tag{2.90}$$

These operators allows to express $[z]_B$ as

$$[z]_B = [D(z * \tilde{\varphi}), D(z * \tilde{\psi})]. \tag{2.91}$$

Equation (2.91) illustrates a representation of the signal z by the vectors of approximations and details and it is also in accordance with the idea of the best approximation. To construct the wavelet bases for the whole analysis at the J^{th} level seems to be very complex. However, the wavelet basis has one important property, that it can be constructed only from generators at the 1^{st} level. This will be described in the following proposition.

Proposition 2. *Let* N *be divisible by* 2^J, $J \in \mathbb{N}$ *and let* $\varphi_1, \psi_1 \in \ell^2(\mathbb{Z}_N)$ *be a pair generators of wavelet basis at the* 1^{st} *level. Construct the sequence of pairs of vectors* φ_j, ψ_j *for* $j = 2, \ldots, p$ *as follows:*

$$\varphi_j[n] = \sum_{k=0}^{2^{j-1}-1} \varphi_1\left[n + \frac{kN}{2^{j-1}}\right], \psi_j[n] = \sum_{k=0}^{2^{j-1}-1} \psi_1\left[n + \frac{kN}{2^{j-1}}\right],\ for\ n = 0, \ldots, \frac{N}{2^{j-1}} - 1. \tag{2.92}$$

Then the set $\{\varphi_j, \psi_j\}_{j=1}^{J}$ *is a sequence of wavelet basis generators for analysis up to the* J^{th} *level.*

The wavelet analysis is now performed in accordance to the MRA through a *pyramidal algorithm*. Firstly, the representation of $z \in \ell^2(\mathbb{Z}_N)$ as in equation (2.91) allows to obtain vectors of approximation $a_1 = D(z * \tilde{\varphi}_1)$ and detail $d_1 = D(z * \tilde{\psi}_1)$ on the 1^{st} level, both from $\ell^2(\mathbb{Z}_{N/2})$. Then continue with analysis of the vector a_1 by repeating the procedure with φ_2 and ψ_2, so vectors $a_2 = D(a_1 * \tilde{\varphi}_2)$ and $d_2 = D(a_1 * \tilde{\psi}_2)$ are obtained, both from $\ell^2(\mathbb{Z}_{N/2^2})$. The procedure continues

up to the J^{th} level, where the final approximation a_J and final detail d_J are found, both from $\ell^2(\mathbb{Z}_{N/2^J})$. The $N-1$ coefficients that are associated with changes on various scales, $[d_1, \ldots, d_J]$,are called *wavelet coefficients* and a_J is called a *scaling coefficient*. Furthermore, d_j are coefficients associated with changes in z at scale $\tau_j = 2^{j-1}$ for $j = 1, \ldots, J$, and a_J is the coefficient associated with an average at scale 2^J.

Definition 20. *Let N be divisible by 2^J, $J \in \mathbb{N}$ and suppose $\varphi_j, \psi_j \in \ell^2(\mathbb{Z}_{N/2^{j-1}})$ is a pair of generators of a wavelet basis of $\ell^2(\mathbb{Z}_{N/2^{j-1}}), j = 1, 2, \ldots, J$. Then the vector $[d_1, d_2, \ldots, d_J, a_J]$, obtained as described above, is called the Discrete Wavelet Transform (DWT) of the vector $z \in \ell^2(\mathbb{Z}_N)$ at the J^{th} level.*

So far the theory of DWT for discrete time series was developed. Now, a brief look at the reconstruction of a discrete time series from its DWT is shown below.

Proposition 3. *Let $M \in \mathbb{N}, N = 2M$ and let $\varphi_1, \psi_1 \in \ell^2(\mathbb{Z}_N)$ be the generators of wavelet basis of $\ell^2(\mathbb{Z}_N)$. Let $D(z * \tilde{\varphi}_1) = x_1 \in \ell^2(\mathbb{Z}_M)$, $D(z * \tilde{\psi}_1) = y_1 \in \ell^2(\mathbb{Z}_M)$. Then the following equation holds*

$$\varphi_1 * U(x_1) + \psi_1 * U(y_1) = z, \ \forall z \in \ell^2(\mathbb{Z}_N). \tag{2.93}$$

Proposition 3 gives a equation for a perfect reconstruction of the original discrete time series $z \in \ell^2(\mathbb{Z}_N)$. If the DWT of z at the J^{th} level, $[d_1, d_2, \ldots, d_J, a_J]$, is given, then the original time series z is found through a backward procedure:

$$\begin{gathered}
\varphi_J * U(a_J) + \psi_J * U(d_J) = a_{J-1}, \\
\vdots \\
\varphi_j * U(a_j) + \psi_j * U(d_j) = a_{j-1}, \\
\vdots \\
\varphi_1 * U(a_1) + \psi_1 * U(d_1) = z.
\end{gathered}$$

Next subsection will deal with more details about how to find the wavelet coefficients from an algorithmic approach.

2.6.4 Pyramid Algorithm: the filtering approach

Previous subsection made a treatment of the formal DWT theory and a formal process to obtain the DWT of a discrete time series was studied. However, in practice the DWT needs a more suitable way to obtain its parameters. From an algorithmic perspective, vectors of wavelet (detail) and scale (approximation) coefficients are computed using a pyramid algorithm that makes use of a *filtering approach*.

Definition 21. *A filter* $\{h_l : l = 0, 1, \ldots, L-1\}$ *of even width* L *is called a wavelet filter if*

$$\sum_{l=0}^{L-1} h_l = 0 \tag{2.94}$$

$$\sum_{l=0}^{L-1} h_l h_{l+2n} = \begin{cases} 1\,, if\ n = 0 \\ 0\,, if\ n \neq 0 \end{cases} \tag{2.95}$$

In words, a wavelet filter must have mean zero, in order to preserve the addmisibility condition (2.74); and must be orthonormal respect to its even shifts. Equation (2.95) is referred as the *orthonormality property* of wavelet filters.

Definition 22. *The scaling filter* $\{g_l : l = 0, 1, \ldots, L-1\}$ *is defined in terms of the wavelet filter via the* ***quadrature mirror relationship (QMR)*** *as*

$$g_l = (-1)^{l+1} h_{L-1-l} \tag{2.96}$$

A wavelet filter is also known as a *low-pass filter* because it removes all frequencies that are above half of the highest frequency in the discrete time series. On the other hand, a scaling filter is known as a *high-pass filter* because it retains all frequencies that are above half of the highest frequency in the discrete time series. Due to each filter has a nominal pass-band covering half the full band of frequencies, both $\{h_l\}$ and $\{g_l\}$ are usually called *half-band filters*.

Intuitively, the pyramid algorithm using the filtering approach works as follows. Suppose a discrete time $z \in \ell^2(\mathbb{Z}_N)$ where N is divisible by 2^J, $J \in \mathbb{N}$ and whose maximum frequency is p. After passing z through a half-band lowpass filter, half of the samples can be eliminated, since the

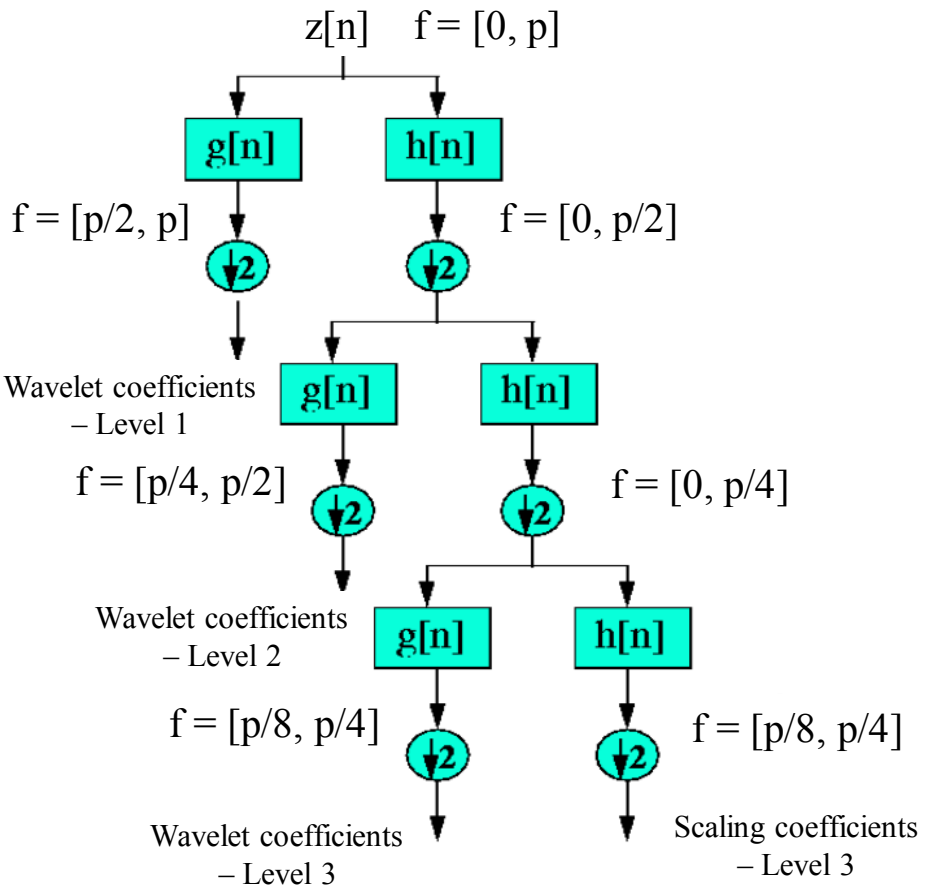

Figure 2.15: DWT process following a pyramid algorithm of a discrete time series z with three levels of decomposition. The maximum frequency of z is p.

signal now has a highest frequency of $p/2$ instead of p. Simply discarding every other sample will subsample the signal by two, and the signal will then have half the number of points. The scale of the signal is now doubled. Note that the lowpass filter removes the high frequency information, but leaves the scale unchanged. Only the subsampling process changes the scale. Resolution, on the other hand, is related to the amount of information in the signal, and therefore, it is affected by the filtering operations. The low-pass filter removes half of the frequencies, which can be interpreted as losing half of the information. Therefore, the resolution is halved after the filtering operation. Note, however, the subsampling operation after filtering does not affect the resolution, since removing half

of the spectral components from the signal makes half the number of samples redundant anyway. Half the samples can be discarded without any loss of information. In summary, the low-pass filter halves the resolution, but leaves the scale unchanged. The signal is then subsampled by 2 since half of the number of samples are redundant. This doubles the scale. Figure 2.15 shows an example of this process.

The wavelet and scale coefficients are obtained explicitly in the following way. Suppose $\{h_l\}$ and $\{g_l\}$ are the wavelet and scale filters respectively. Let d_j and a_j be the vectors of wavelet and scaling coefficients at the j^{th} level, respectively. For $j = 1, \ldots, J$, let $\delta_{j,n}$ be the n^{th} component of vector d_j for $n = 0, 1, \ldots, N/2^j - 1$, and $\alpha_{j,n}$ as the n^{th} component vector a_j, for $n = 0, 1, \ldots, N/2^J - 1$. Suppose $\alpha_{0,n} = z[n]$, then the wavelet and scale coefficients at the j^{th} level are found through the formulas (Percival and Walden, 2000):

$$\delta_{j,n} = \sum_{k=0}^{L-1} h_k \alpha_{j-1,\,(2n+1-k) \mod N/2^{j-1}}; \tag{2.97}$$

$$\alpha_{j,n} = \sum_{k=0}^{L-1} g_k \alpha_{j-1,\,(2n+1-k) \mod N/2^{j-1}}; \tag{2.98}$$

where $p \mod q$ represents the remainder of p/q. Therefore, the DWT coefficients are composed by the vectors $d_1, \ldots, d_J$ and a_J.

An alternative way to construct the DWT coefficients is through the formulas

$$\begin{aligned} \delta_{j,n} &= \sum_{k=0}^{L_j-1} h_{j,k} \cdot z[(2^j(n+1) - 1 - k) \mod N]; \\ \alpha_{j,n} &= \sum_{k=0}^{L_j-1} g_{j,k} \cdot z[(2^j(n+1) - 1 - k) \mod N]; \end{aligned} \tag{2.99}$$

where $\{h_{j,l}\}$ and $\{g_{j,l}\}$ are the j^{th} level wavelet and scaling filters, each having width $L_j = (2^j - 1)(L - 1) + 1$ and built through the relation (Percival and Walden, 2000):

$$h_{j,l} = \sum_{k=0}^{L_{j-1}-1} g_{l-2k} h_{j-1,k} \;\; and \;\; g_{j,l} = \sum_{k=0}^{L_{j-1}-1} g_{l-2k} g_{j-1,k}; \tag{2.100}$$

for $l = 0, 1, \ldots, L_j - 1$.

Equation (2.99) gives an alternative way of representing the DWT through matrices. Indeed, equation (2.99) permits to represent the approximation and detail coefficients as linear transformations of the original signal z. Explicitly, considering d_j and a_J as column vectors, then

$$d_j = \mathcal{D}_j z, \quad for\ j = 1, \ldots, J; \tag{2.101}$$

$$a_J = \mathcal{A} z; \tag{2.102}$$

where $\mathcal{D}_j$ are matrices of order $N/2^j \times N$ and $\mathcal{A}$ is a matrix of order $N/2^J \times N$. Following this nomenclature, the vector of detail and approximation coefficients, $W = [d_1', d_2', \ldots, d_J', a_J']'$, can be expressed as

$$W = \mathcal{W} z; \tag{2.103}$$

where $\mathcal{W} = [\mathcal{D}_1' : \mathcal{D}_2' : \cdots : \mathcal{D}_J' : \mathcal{A}']'$ is an orthogonal matrix of order $N \times N$.

Moreover, equation (2.99) enables to reconstruct the original time series given its DWT coefficients. It suffices to use equation (2.103) and the orthogonality of $\mathcal{W}$ to get:

$$z = \mathcal{W}' W = [\mathcal{D}_1' : \mathcal{D}_2' : \cdots : \mathcal{D}_J' : \mathcal{A}'] \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_J \\ a_J \end{bmatrix}$$

$$z = \sum_{i=1}^{J} \mathcal{D}_i' d_i + \mathcal{A}' a_J$$

$$z = \sum_{i=1}^{J} \mathbf{D}_i + \mathbf{A}; \tag{2.104}$$

where $\mathbf{D}_i = \mathcal{D}_i' d_i$ is an N dimensional column vector whose elements are associated with changes

in z at scale $\tau_i = 2^{i-1}$; i.e., $\mathbf{D}_i$ is the portion of the synthesis $z = \mathcal{W}'W$ attributable to scale τ_i; and $\mathbf{A} = \mathcal{A}'a_J$ is an N dimensional vector whose elements are associated with an average at scale 2^J.

2.6.5 Maximal Overlap Discrete Wavelet Transform (MODWT)

As it was seen, the DWT maps a discrete time series from its original representation in the time domain into an alternative representation in the time-scale domain by recursively applying two filters: the low-pass filter and the high-pass filter. The first reconstructs the smooth and low frequency parts of a discrete signal, whereas the latter describes the detailed and high-frequency parts of a signal. However, although its popularity due to its intuitive approach, the classic DWT suffers from an important drawback: the sample size must be divisible by 2^J, $J \in \mathbb{N}$. In order to address this drawback, the *Maximal Overlap Discrete Wavelet Transform* (MODWT) gives up the orthogonality property of the DWT to gain other features such as (Percival and Walden, 2000):

- the ability to handle any sample size of a discrete time series regardless of whether dyadic or not;
- increased resolution at coarser scales as the MODWT oversamples the data;
- invariant to circularly shifting the original time series; i.e., shifting the time series by an integer unit will shift the MODWT wavelet and scaling coefficients the same amount.

To build the MODWT coefficients a rescaling of the defining half-band filters are required to conserve energy, that is, if $\{h_l\}$ and $\{g_l\}$ are the high-pass and low-pass filter associated to a DWT (see definition 21 and 22), then the associated filters to the MODWT are defined as

$$\tilde{h}_l = h_l/\sqrt{2} \;\; and \;\; \tilde{g}_l = g_l/\sqrt{2} \; for \; l = 0, 1, \ldots, L-1. \tag{2.105}$$

These MODWT filters are still quadrature mirror filters since the QMR holds (see equation (2.96)). Furthermore,

$$\sum_{l=0}^{L-1} \tilde{h}_l = 0 \tag{2.106}$$

$$\sum_{l=0}^{L-1} \tilde{h}_l \tilde{h}_{l+2n} = \begin{cases} 1/2\,, if\ n = 0 \\ 0 \quad\ , if\ n \neq 0. \end{cases} \tag{2.107}$$

The scaling MODWT filter $\{\tilde{g}_l\}$ is also required to satisfy equation (2.107) and $\sum_{l=0}^{L-1} \tilde{g}_l = 1$. Then, analogous to the construction of DWT coefficients, the MODWT coefficients are built using the pyramid algorithm. So, given a discrete time series $z \in \ell^2(\mathbb{Z}_N)$, let $\tilde{d}_j$ and $\tilde{a}_j$ be the vectors of wavelet and scaling MODWT coefficients at the j^{th} level, respectively. For $j = 1, \ldots, J$, let $\tilde{\delta}_{j,n}$ and $\tilde{\alpha}_{j,n}$ be the n^{th} component of vector $\tilde{d}_j$ and $\tilde{a}_j$ respectively, for $n = 0, 1, \ldots, N-1$. Suppose $\tilde{\alpha}_{0,n} = z[n]$, then the wavelet and scaling MODWT coefficients at the j^{th} level are built through the formulas (Percival and Walden, 2000):

$$\tilde{\delta}_{j,n} = \sum_{k=0}^{L-1} \tilde{h}_k \tilde{\alpha}_{j-1,\,(n+2^{j-1}k) \mod N}; \tag{2.108}$$

$$\tilde{\alpha}_{j,n} = \sum_{k=0}^{L-1} \tilde{g}_k \tilde{\alpha}_{j-1,\,(n+2^{j-1}k) \mod N}; \tag{2.109}$$

for $n = 0, 1, \ldots, N-1$. Equation (2.108) can also be formulated as circular filter operations of the original time series z using the filters $\{\tilde{h}_{j,l} = h_{j,l}/2^{j/2}\}$ and $\{\tilde{g}_{j,l} = g_{j,l}/2^{j/2}\}$ (see equation (2.100)), namely,

$$\begin{aligned} \tilde{\delta}_{j,n} &= \sum_{k=0}^{L_j-1} \tilde{h}_{j,k} \cdot z[(n-k) \mod N]; \\ \tilde{\alpha}_{j,n} &= \sum_{k=0}^{L_j-1} \tilde{g}_{j,k} \cdot z[(n-k) \mod N]; \end{aligned} \tag{2.110}$$

where $L_j = (2^j - 1)(L-1) + 1$ for $j = 1, \ldots, J$.

Equation (2.110) allows to build a matrix representation of MODWT coefficients. Indeed, denote $\tilde{d}_j = (\tilde{\delta}_{j,n})_{n=0,1,\ldots,N-1}$ as the vector of MODWT wavelet coefficients at the j^{th} level (associated with changes on scale $\tau_j = 2^{j-1}$) and $\tilde{a}_J = (\tilde{\alpha}_{J,n})_{n=0,1,\ldots,N-1}$ as the vector of MODWT scaling

coefficients (associated with averages on scale 2^J). Then, equation (2.110) gives

$$\tilde{d}_j = \tilde{\mathcal{D}}_j z \; for \; j = 1, \ldots, J; \tag{2.111}$$

$$\tilde{a}_J = \tilde{\mathcal{A}} z; \tag{2.112}$$

where $\tilde{\mathcal{D}}_j$ and $\tilde{\mathcal{A}}_J$ are $N \times N$ matrices.

Definition 23. *Let $z \in \ell^2(\mathbb{Z}_N)$, where the sample size N is any positive integer. Given the filters $\{\tilde{h}_l\}_{l=0}^{L-1}$ and $\{\tilde{g}_l\}_{l=0}^{L-1}$ where L is an even number, then the vector $[\tilde{d}_1', \ldots, \tilde{d}_J', \tilde{a}_J']'$, obtained as described above, is called the Maximal Overlap Discrete Wavelet Transform (MODWT) of z at the J^{th} level.*

The original signal can be recovered by using equations (2.111) and (2.112). In fact, Percival and Walden (2000) showed that

$$z = \sum_{j=1}^{J} \tilde{\mathcal{D}}_j' \tilde{d}_j + \tilde{\mathcal{A}}' \tilde{a}_J$$

$$z = \sum_{j=1}^{J} \tilde{\mathbf{D}}_j + \tilde{\mathbf{A}} \tag{2.113}$$

Chapter 3

Proposed methods for spatio-temporal modeling of raster datasets

3.1 Introduction

The objective of this research is to demonstrate an effective method to forecast spatio-temporal raster datasets. To achieve this objective, three methods based on the use of PCA, MODWT and an AR-NN model are proposed. The general idea behind the proposed models is to summarize the essential spatio-temporal variability and then apply a prediction model over it. For this purpose, a PCA summarizes the essential spatio-temporal variability considering a dimensionality reduction on the raster dataset. Then a prediction model, using an AR-NN model over the decomposed series by a MODWT, is used to get forecasts of the significant information (PCs or eigenvectors). Finally the forecast maps are obtained through a spectral reverse reconstruction and a recursive algorithm.

The next sections describe the three proposed models. Sections 3.2 and 3.3 develop the temporal and spatial principal component analysis model respectively. Finally, section 3.4 presents the spatio-temporal principal component analysis model, which behaves as an ensembled model of the two previous models since it ponderates the forecast results of both according to its forecast performance.

3.2 The Temporal Principal Component Analysis model

This model analyzes the dataset $\mathbf{X}_{n\times p}$ (see figure 2.9) that summarizes the temporal and spatial information in rows and columns respectively. As the temporal structure is the dimension whose variability must be summarized by a PCA, this model is called the *TPCA model*. In fact, $\mathbf{X}$ follows an S mode structure according to Richman (see table 2.1) and as a result of a PCA application, the PCs are time-varying and they can be modeled using an AR-NN model.

Suppose that $\mathbf{X}$ induces the sample correlation matrix $\mathbf{R}$. According to PCA theory, the principal components associated to $\mathbf{X}$, the matrix $\mathbf{Z}$, are obtained through the orthonormalized eigenvectors of $\mathbf{R}$, such that

$$\mathbf{Z} = \mathbf{XP};$$

where $\mathbf{P}$ is the $p \times p$ matrix of orthonormalized eigenvectors of $\mathbf{R}$. Since $\mathbf{P}$ is an orthogonal matrix, $\mathbf{X}$ can be reconstructed (spectral reverse reconstruction) as

$$\mathbf{X} = \mathbf{ZP}'. \tag{3.1}$$

If the scree test optimal coordinate (see subsection 2.4.4) is applied and gives $k(< p)$ as the number of retained components, then $\mathbf{Z}$ and $\mathbf{P}$ are partitioned as

$$\mathbf{Z} = [\mathbf{Z}_1 : \mathbf{Z}_2] \quad and \quad \mathbf{P} = [\mathbf{P}_1 : \mathbf{P}_2]\,; \tag{3.2}$$

where $\mathbf{Z}_1, \mathbf{Z}_2, \mathbf{P}_1$ and $\mathbf{P}_2$ are of orders $n \times k$, $n \times (p-k)$, $p \times k$ and $p \times (p-k)$ respectively. Replacing equation (3.2) in equation (3.1), then

$$\mathbf{X} = \mathbf{Z}_1\mathbf{P}_1' + \mathbf{Z}_2\mathbf{P}_2'. \tag{3.3}$$

Since the significant variability of $\mathbf{X}$ is concentrated into the first right term of equation (3.3), it is reasonable to suppose that the second right term is related to noise signals and is, in terms of

variability, "negligible"[1]. Therefore, $\mathbf{Z}_2\mathbf{P}_2' \approx \mathbf{0}$ and equation (3.3) can result in

$$\mathbf{X} \approx \mathbf{Z}_1\mathbf{P}_1'. \tag{3.4}$$

The next step is to forecast the time-varying significant PCs. In the ideal case, the forecast of the k first PCs would be enough; however, if the rule underestimates the number of significant components there may be hidden essential information in the last $p-k$ PCs that would be important to model and; conversely, if the rule overestimates the number, there would be innecesary PCs to forecast in the first k PCs. In order to cope with this, the forecast of non significant PCs are also considered using the historical mean. Explicitly, forecasts considering significant and non significant components are obtained as follows:

1. *Forecasted values of the significant components*: Due to the uncorrelatedness of the k time series, the forecasted values are gathered by forecasting each of the k time series separately. For this purpose, the MODWT-AR-NN model is used. The MODWT-AR-NN applies a MODWT decomposition as a preprocessing method before to prepare an AR-NN model. It is, a MODWT decomposition using a Coifflets (6) wavelet filter (Percival and Walden, 2000) decomposes each PC into a series of approximation and two of details so that a different AR-NN model is applied to each time series component. Figure 3.1 explains this process.

2. *Forecasted values of the non significant components*: The respective mean of the $p-k$ time series is taken as a simple summary forecast of the last $p-k$ uncorrelated components. The forecast of "negligible" components should be considered as a preventive measure in the case that some significant information is not considered by the scree test optimal coordinate.

Denote $\hat{\mathbf{Z}}_1$ and F_1 as the predicted time series of the first k PCs using the MODWT-AR-NN model and the forecast values of the first k PCs respectively. Then, let $\mathbf{Z}_1^{(1)}$ be

$$\mathbf{Z}_1^{(1)} = \left[\hat{\mathbf{Z}}_1' : F_1\right]'. \tag{3.5}$$

[1]Negligible with respect to the test to select components.

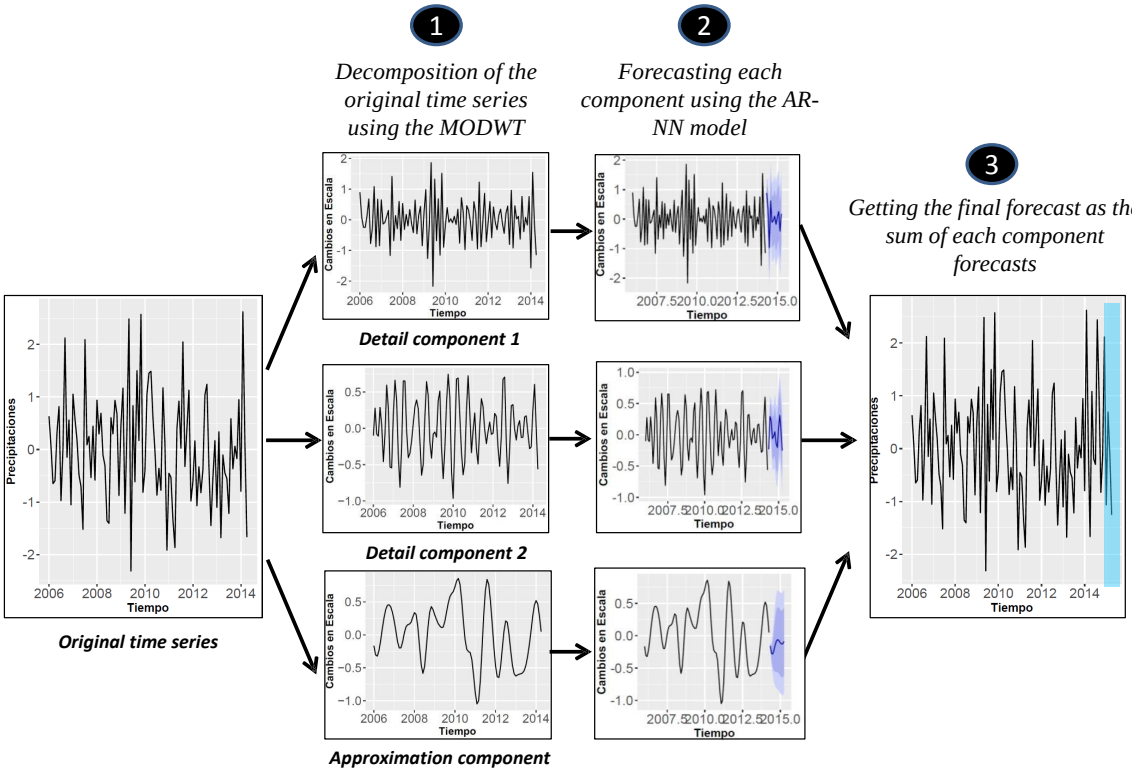

Figure 3.1: Operating scheme of the MODWT-AR-NN model.

Furthermore, let F_2 be the forecast of the non significant PCs. Then, define

$$\mathbf{Z}_2^{(1)} = \left[\mathbf{Z}_2' : F_2\right]'. \tag{3.6}$$

Thereafter the forecast map corresponding to the period $n+1$ is obtained by using the spectral reverse reconstruction of principal components (equation (3.3)), namely

$$\mathbf{X}^{(1)} = \mathbf{Z}_1^{(1)}\mathbf{P}_1' + \mathbf{Z}_2^{(1)}\mathbf{P}_2'; \tag{3.7}$$

where the first n rows of $\mathbf{X}^{(1)}$, given by

$$\hat{\mathbf{X}} =: \hat{\mathbf{Z}}_1\mathbf{P}_1' + \mathbf{Z}_2\mathbf{P}_2' \tag{3.8}$$

compose the predicted values of $\mathbf{X}$, and the $(n+1)^{th}$ row of $\mathbf{X}^{(1)}$, given by

$$\hat{\mathbf{y}}_1 =: F_1'\mathbf{P}_1' + F_2'\mathbf{P}_2' \tag{3.9}$$

is the forecast map corresponding to the period $n+1$.

For a general case, let $h > 1$ be the forecast horizon. Then the predicted maps in-sample and the h forecast maps out-sample are gathered through a recursive algorithm as follows:

1. Calculate the sample correlation matrix $\mathbf{R}$ corresponding to the matrix $\mathbf{X}$.
2. Calculate the matrix of orthonormalized eigenvectors, $\mathbf{P}$, and PCs, $\mathbf{Z}$, corresponding to the matrix $\mathbf{R}$.
3. Select the k significant components and to partition

   $$\mathbf{Z} = [\mathbf{Z}_1 : \mathbf{Z}_2] \quad and \quad \mathbf{P} = [\mathbf{P}_1 : \mathbf{P}_2]$$

   where $\mathbf{Z}_1$ and $\mathbf{P}_1$ are of orders $n \times k$ and $p \times k$ respectively.
4. Get the k predicted time series corresponding to the significant PCs using the MODWT-AR-NN model and save them in the matrix $\hat{\mathbf{Z}}_1$
5. Get $\hat{\mathbf{X}} = \hat{\mathbf{Z}}_1\mathbf{P}_1' + \mathbf{Z}_2\mathbf{P}_2'$, the n predicted maps in-sample.
6. Get the forecasts, with an horizon h, of the k significant components using the MODWT-AR-NN model and save them as vectors

   $$F_1^{(1)}, F_1^{(2)}, \ldots, F_1^{(h)};$$

   where $F_1^{(t)}$ represents the forecast of the first k components at time $n+t$, for $t = 1, \ldots, h$.
7. Get the forecasts, with an horizon 1, of the $p-k$ non significant components calculating the historical mean of these PCs and save it as vector $F_2^{(1)}$.

8. Create matrices $\mathbf{Z}_1^{(0)} = \left[\mathbf{Z}_1' : F_1^{(1)}\right]'$ and $\mathbf{Z}_2^{(0)} = \left[\mathbf{Z}_2' : F_2^{(1)}\right]'$ to get the matrix

$$\mathbf{X}^{(1)} = \mathbf{Z}_1^{(0)}\mathbf{P}_1' + \mathbf{Z}_2^{(0)}\mathbf{P}_2'.$$

9. Save the $(n+1)^{th}$ row of $\mathbf{X}^{(1)}$ as $\mathbf{y}_1$, the forecast map at time $n+1$.

10. For j from 2 to h.

    10.1. Calculate the sample correlation matrix $\mathbf{R}^{(j-1)}$ corresponding to the matrix $\mathbf{X}^{(j-1)}$.

    10.2. Calculate the matrix of orthonormalized eigenvectors, $\mathbf{P}^{(j-1)}$, and PCs, $\mathbf{Z}^{(j-1)}$, corresponding to the matrix $\mathbf{R}^{(j-1)}$.

    10.3. Partition $\mathbf{P}^{(j-1)}$ and $\mathbf{Z}^{(j-1)}$ as

$$\mathbf{Z}^{(j-1)} = \left[\mathbf{Z}_1^{(j-1)} : \mathbf{Z}_2^{(j-1)}\right] \ \textit{and} \ \mathbf{P}^{(j-1)} = \left[\mathbf{P}_1^{(j-1)} : \mathbf{P}_2^{(j-1)}\right]$$

    where $\mathbf{Z}_1^{(j-1)}$ and $\mathbf{P}_1^{(j-1)}$ are of orders $(n+j-1)\times k$ and $p \times k$ respectively.

    10.4. Calculate the average of the last $p-k$ components, and save it in vector $F_2^{(j)}$.

    10.5. Create matrices $\mathbf{Z}_1^{(j)} = \left[\mathbf{Z}_1^{(j-1)'} : F_1^{(j)}\right]'$ and $\mathbf{Z}_2^{(j)} = \left[\mathbf{Z}_2^{(j-1)'} : F_2^{(j)}\right]'$ to get the matrix

$$\mathbf{X}^{(j)} = \mathbf{Z}_1^{(j)}\mathbf{P}_1^{(j-1)'} + \mathbf{Z}_2^{(j)}\mathbf{P}_2^{(j-1)'}.$$

    10.6. Save the $(n+j)^{th}$ row of $\mathbf{X}^{(j)}$ as $\hat{\mathbf{y}}_j$, the forecast map at time $n+j$.

    End for.

3.3 The Spatial Principal Component Analysis model

This model functions analogous to the previous one but it works with the matrix $\mathbf{X}'$ instead. This structure takes the raster locations as data elements and the sampling times as variables. Since the

spatial structure is the dimension whose variability must be summarized by a PCA, this model is called the spatial principal component analysis (SPCA) model. Different from the TPCA model, the SPCA model works with a space-time data with a T mode structure according to Richman (see table 2.1) and as a consequence of a PCA application, the loadings (eigenvectors) are time-varying. This induces that the AR-NN model is applied over the time-varying loadings (eigenvectos) and not over the PCs. Next paragraphs exhibit the process behind the SPCA model.

The SPCA model works with the matrix $\mathbf{X}'$ that induces the sample correlation matrix $\mathbf{H}$. The principal components corresponding to $\mathbf{X}'$, the matrix $\mathbf{N}$, are related with the orthonormalized eigenvectors of $\mathbf{H}$ through

$$\mathbf{N} = \mathbf{X}'\mathbf{Q}; \tag{3.10}$$

where $\mathbf{Q}$ is the $n \times n$ matrix of eigenvectors of $\mathbf{H}$. Since $\mathbf{Q}$ is an orthogonal matrix, $\mathbf{X}'$ may be reconstructed as

$$\mathbf{X}' = \mathbf{N}\mathbf{Q}'. \tag{3.11}$$

A simple transpose operation in both sides of equation (3.11) gives the reconstruction of $\mathbf{X}$,

$$\mathbf{X} = \mathbf{Q}\mathbf{N}'. \tag{3.12}$$

If the scree test optimal coordinate gives m ($< n$) as the number of retained components, then $\mathbf{Q}$ and $\mathbf{N}$ are partitioned as

$$\mathbf{N} = [\mathbf{N}_1 : \mathbf{N}_2] \quad \textit{and} \quad \mathbf{Q} = [\mathbf{Q}_1 : \mathbf{Q}_2]\,; \tag{3.13}$$

where $\mathbf{N}_1, \mathbf{N}_2, \mathbf{Q}_1$ and $\mathbf{Q}_2$ are of orders $p \times m$, $p \times (n-m)$, $n \times m$ and $n \times (n-m)$ respectively. If these partitions are replaced in equation (3.12), then

$$\mathbf{X} = \mathbf{Q}_1\mathbf{N}_1' + \mathbf{Q}_2\mathbf{N}_2'. \tag{3.14}$$

The next step is to forecast the summarized information of the significant loadings. Similar to the TPCA model, the first m eigenvectors are forecasted separately using the MODWT-AR-NN model and for the last $n - m$ non significant eigenvectors the historical average is calculated. For the

predicted maps in-sample, let $\hat{\mathbf{Q}}_1$ be the predicted time series of the first m eigenvectors using the MODWT-AR-NN model.

Let G_1 and G_2 be vectors with the forecasts of the first m and the last $n-m$ eigenvectors respectively. Define $\mathbf{Q}^{(1)}$ and $\mathbf{Q}^{(2)}$ as

$$\mathbf{Q}^{(1)} = \left[\hat{\mathbf{Q}}_1' : G_1\right]' \quad and \quad \mathbf{Q}^{(2)} = \left[\mathbf{Q}_2' : G_2\right]'. \tag{3.15}$$

Replacing $\mathbf{Q}^{(1)}$ and $\mathbf{Q}^{(2)}$ in equation (3.14) then

$$\mathbf{X}^{(1)} = \mathbf{Q}^{(1)}\mathbf{N}_1' + \mathbf{Q}^{(2)}\mathbf{N}_2'; \tag{3.16}$$

where the first n rows of $\mathbf{X}^{(1)}$, given by

$$\hat{\mathbf{X}} =: \hat{\mathbf{Q}}_1\mathbf{N}_1' + \mathbf{Q}_2\mathbf{N}_2', \tag{3.17}$$

compose the n predicted maps in-sample for $\mathbf{X}$, and the $(n+1)^{th}$ row of $\mathbf{X}^{(1)}$,

$$\hat{\mathbf{y}}_1 =: G_1'\mathbf{N}_1' + G_2'\mathbf{N}_2', \tag{3.18}$$

corresponds to the forecast map at time $n+1$.

For a general case, if $h > 1$ is the forecast horizon, the predicted maps in-sample and the h forecast maps out-sample are found through a recursive algorithm analogous to the TPCA model:

1. `Calculate the sample correlation matrix` $\mathbf{H}$ `corresponding to the matrix` $\mathbf{X}'$.
2. `Calculate the matrix of orthonormalized eigenvectors,` $\mathbf{Q}$`, and PCs,` $\mathbf{N}$`, corresponding to the matrix` $\mathbf{H}$.
3. `Select the` m `significant eigenvectors and to partition`

$$\mathbf{Q} = [\mathbf{Q}_1 : \mathbf{Q}_2] \quad and \quad \mathbf{N} = [\mathbf{N}_1 : \mathbf{N}_2]\,;$$

where $\mathbf{Q}_1$ and $\mathbf{N}_1$ are of orders $n \times m$ and $p \times m$ respectively.

4. Get the m predicted time series corresponding to the significant eigenvectors using the MODWT-AR-NN model and save them in the matrix $\hat{\mathbf{Q}}_1$.

5. Calculate $\hat{\mathbf{X}} = \hat{\mathbf{Q}}_1 \mathbf{N}_1' + \mathbf{Q}_2 \mathbf{N}_2'$, the n predicted maps in-sample.

6. Get the forecast, with an horizon h, of the m significant eigenvectors using the MODWT-AR-NN model. Save them as vectors

$$G_1^{(1)}, G_1^{(2)}, \ldots, G_1^{(h)};$$

where $G_1^{(t)}$ represents the forecast of the first m eigenvectors at time $n+t$, for $t = 1, \ldots, h$.

7. Get the forecast, with an horizon of 1, of the $n-m$ non significant eigenvectors using the average of each eigenvector. Save it as vector $G_2^{(1)}$.

8. Create matrices $\mathbf{Q}_1^{(0)} = \left[\mathbf{Q}_1' : G_1^{(1)}\right]'$ and $\mathbf{Q}_2^{(0)} = \left[\mathbf{Q}_2' : G_2^{(1)}\right]'$ to get the matrix

$$\mathbf{X}^{(1)} = \mathbf{Q}_1^{(0)} \mathbf{N}_1' + \mathbf{Q}_2^{(0)} \mathbf{N}_2'.$$

9. Save the $(n+1)^{th}$ row as $\mathbf{y}_1$, the forecast map at time $n+1$.

10. For j from 2 to h.

    10.1. Calculate the sample correlation matrix $\mathbf{H}^{(j-1)}$ corresponding to the matrix $\mathbf{X}^{(j-1)'}$.

    10.2. Get the matrix of orthonormalized eigenvectors, $\mathbf{Q}^{(j-1)}$, and PCs, $\mathbf{N}^{(j-1)}$, corresponding to the matrix $\mathbf{H}^{(j-1)}$.

10.3. Partition $\mathbf{N}^{(j-1)}$ and $\mathbf{Q}^{(j-1)}$ as

$$\mathbf{N}^{(j-1)} = \left[\mathbf{N}_1^{(j-1)} : \mathbf{N}_2^{(j-1)}\right] \ \textit{and} \ \mathbf{Q}^{(j-1)} = \left[\mathbf{Q}_1^{(j-1)} : \mathbf{Q}_2^{(j-1)}\right]$$

where $\mathbf{N}_1^{(j-1)}$ and $\mathbf{Q}_1^{(j-1)}$ are of orders $(p+j-1)\times m$ and $n\times m$ respectively.

10.4. Calculate the average of the last $n-m$ eigenvectors and save it in a vector $G_2^{(j)}$.

10.5. Create matrices $\mathbf{Q}_1^{(j)} = \left[\mathbf{Q}_1^{(j-1)'} : G_1^{(j)}\right]'$ and $\mathbf{Q}_2^{(j)} = \left[\mathbf{Q}_2^{(j-1)'} : G_2^{(j)}\right]'$ to get the matrix

$$\mathbf{X}^{(j)} = \mathbf{Q}_1^{(j)}\mathbf{N}_1^{(j-1)'} + \mathbf{Q}_2^{(j)}\mathbf{N}_2^{(j-1)'}.$$

10.6. Save the $(n+j)^{th}$ row as $\hat{\mathbf{y}}_j$, the forecast map at time $n+j$.

End for.

3.4 The Spatio - Temporal Principal Component Analysis model

This model consists of an hybridization of the TPCA and SPCA model. In fact, the spatio-temporal PCA model (in short STPCA model) tries to capture the spatial and temporal variability in a proportional way according to its importance. It means that if it is supposed that total variability is composed, separately, by spatial and temporal variability, then one kind of variability can be considered more important than the other for modeling purposes. This is developed mathematically by using a convex combination of results in the TPCA and SPCA model. Indeed, a convex combination of equations (3.1) and (3.12) leads to

$$\mathbf{X} = \alpha \cdot \mathbf{ZP'} + (1-\alpha)\cdot \mathbf{QN'}; \tag{3.19}$$

where $\alpha \in [0,1]$.

Let F and G be the corresponding forecast maps at time $n+1$ by using the TPCA and SPCA

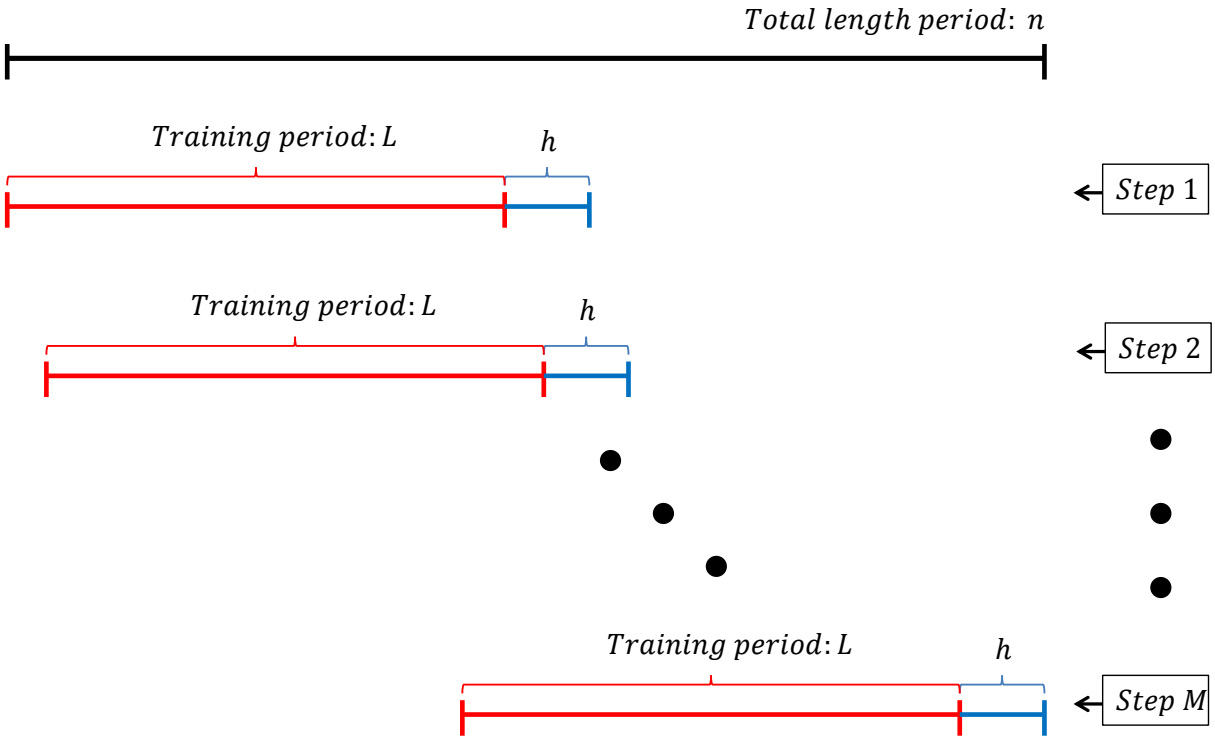

Figure 3.2: An sketch of how to use the training period to calculate the "importance" weights. n, L and h represent the lengths of total period, training period and forecast horizon respectively.

model respectively, thus define

$$T = \left[\mathbf{PZ}' : F\right]' \quad and \quad S = \left[\mathbf{NQ}' : G\right]'; \tag{3.20}$$

then with equation (3.19) the first forecast map is obtained through

$$\mathbf{X}^* = \alpha \cdot T + (1-\alpha) \cdot S \quad with \quad \alpha \in [0,1]; \tag{3.21}$$

where the first n rows compose the original dataset $\mathbf{X}$ and the $(n+1)^{th}$ row is the forecast map at time $n+1$ belonging to the STPCA model. Since the terms T and S correspond to the TPCA and SPCA model respectively, then α (and $1-\alpha$) represents a weight measuring the "importance" of

TPCA (and SPCA) model forecast results. The calculation of this "importance" measure depends on the forecast performance of each model. For this purpose, let h be the forecast horizon and a training period of length $L < n$ is chosen. Figure 3.2 shows an sketch of how the training period helps to find an estimation for α. The idea is to use the training period L as a mobile window that moves through the complete length period, at each movement by dropping the first map and adding the one following its last current one. At each step, get the h forecast maps with a TPCA and a SPCA model and save the respective performance measures. If E_{ij}^T (E_{ij}^S) denotes a global forecast error measure with the TPCA (SPCA) model at horizon j and step i, for $i = 1, \ldots, M = n - L - h$ and $j = 1, \ldots, h$. Then, the training "importance" weights are defined as

$$\alpha_{ij}^T = \frac{1/E_{ij}^T}{1/E_{ij}^T + 1/E_{ij}^S} \;\; and \;\; \alpha_{ij}^S = 1 - \alpha_{ij}^T; \tag{3.22}$$

for $i = 1, \ldots, M$ and $j = 1, \ldots, h$. Note that since E_{ij}^T and E_{ij}^S represent measures of forecast error and "importance" weights are related with the good forecast performance, then the "importance" weights must have an inverse relation with the forecast errors. Furthermore, the "importance" weight is not constant in all the forecast horizon, it means that the weights change according to the forecast horizon position. In order to handle the operations, the training "importance" weights are put into the matrices:

$$A_T = \begin{pmatrix} \alpha_{11}^T & \alpha_{12}^T & \ldots & \alpha_{1h}^T \\ \alpha_{21}^T & \alpha_{22}^T & \ldots & \alpha_{2h}^T \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{M1}^T & \alpha_{M2}^T & \ldots & \alpha_{Mh}^T \end{pmatrix} \;\; and \;\; A_S = \begin{pmatrix} \alpha_{11}^S & \alpha_{12}^S & \ldots & \alpha_{1h}^S \\ \alpha_{21}^S & \alpha_{22}^S & \ldots & \alpha_{2h}^S \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{M1}^S & \alpha_{M2}^S & \ldots & \alpha_{Mh}^S \end{pmatrix}.$$

Matrix A_T (A_S) saves the training "importance" weights of the TPCA (SPCA) model. This notation allows to estimate the forecast importance "weights" of both models and for each forecast horizon time. To achieve this, note that since $\alpha_{ij}^T + \alpha_{ij}^S = 1, \forall i, j$, then

$$(\overline{\alpha}_1^T, \overline{\alpha}_2^T, \ldots, \overline{\alpha}_h^T) + (\overline{\alpha}_1^S, \overline{\alpha}_2^S, \ldots, \overline{\alpha}_h^S) = (1, 1, \ldots, 1); \tag{3.23}$$

where $\overline{\alpha}_i^T$ ($\overline{\alpha}_i^S$) represents the median of the M training weights at forecast horizon i for the TPCA (SPCA) model, for $i = 1, \ldots, h$.

Once the forecast "importance" weights of each model and for each horizon position are found, the h forecast maps are found making the corresponding ponderation of TPCA and SPCA forecasts according to its "importance". The next algorithm explains how:

1. Define L, h and n.

2. For i from 1 to $M = n - L - h$.

   2.1. Calculate the forecast ''importance'' weights $\overline{\alpha}_i^T$ and $\overline{\alpha}_i^S$.

   End for

3. For j from 1 to h.

   3.1. Get the forecast maps $\hat{\mathbf{y}}_j^T$ and $\hat{\mathbf{y}}_j^S$ using the TPCA and SPCA model respectively.

   3.2. Calculate

$$\hat{\mathbf{y}}_j^{ST} = \overline{\alpha}_j^T \hat{\mathbf{y}}_j^T + \overline{\alpha}_j^S \hat{\mathbf{y}}_j^S;$$

   where $\hat{\mathbf{y}}_j$ is the forecast map at time $n + j$.

   End for

The process to obtain the predicted maps in-sample through a STPCA model is simpler and makes use of prediction "importance" weights. Indeed, let $\hat{\mathbf{X}}_S$ and $\hat{\mathbf{X}}_T$ be the predicted maps in-sample for $\mathbf{X}$ using the SPCA and TPCA model respectively. If ε_i^T (ε_i^S) represents a global prediction error measure corresponding to the TPCA (SPCA) model at time position i, for $i = 1, \ldots, n$, then the prediction "importance" weights at each time position is determined by

$$\beta_i^T = \frac{1/\varepsilon_i^T}{1/\varepsilon_i^T + 1/\varepsilon_i^S} \;\; and \;\; \beta_i^S = 1 - \beta_i^T \;\; for\, i = 1, \ldots, n. \tag{3.24}$$

In order to get a compact formula for the predicted maps, define the matrices of prediction "importance" weights as

$$M_T = \begin{pmatrix} \beta_1^T & 0 & \dots & 0 \\ 0 & \beta_2^T & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \beta_n^T \end{pmatrix} \quad and \quad M_S = \begin{pmatrix} \beta_1^S & 0 & \dots & 0 \\ 0 & \beta_2^S & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \beta_n^S \end{pmatrix};$$

where M_T and M_S represent the matrix of prediction "importance" weights of the TPCA and SPCA model respectively. The predicted maps in-sample for $\mathbf{X}$ using the STPCA model are found through the formula

$$\hat{\mathbf{X}}_{ST} = M_T \hat{\mathbf{X}}_T + M_S \hat{\mathbf{X}}_S; \tag{3.25}$$

where the i^{th} row of $\hat{\mathbf{X}}_{ST}$ is the predicted map at time position i, for $i = 1, \dots, n$.

Chapter 4

Simulation study

In order to verify the forecast performance of the three proposed models, a simulation study is carried out. For this purpose, three type of processes are simulated. These simulations consist in a regular spatial grid of 30×30 scattered into a square of $1u^2$ (where u is any unit of length), represented by D, and a temporal period of length 250 m (where m is any unit of time). The processes are divided according to the type of variability they possess. Subsection 4.1 describes the experiments with a pure spatial variability process, subsection 4.2 the pure temporal variability and the experiment using a spatio-temporal process is described in Subsection 4.2. Finally, subsection 4.4 shows the results of the simulation study.

4.1 The pure spatial variability process

The pure spatial variability process (PSV process, in short) consists in a simulated process of the form

$$y(s,t) = \mu + \varepsilon(s) \quad for\ s \in D\ and\ t = 1, 2, \ldots, 250; \tag{4.1}$$

where $y(s,t)$ denotes the spatio-temporal observation, μ represents the structured spatial mean and $\varepsilon(s)$ denotes the space residual field which saves the spatial covariance structure. Observe that this process does not have a temporal variability component and only the spatial variability is present. Furthermore, spatial covariance models based on isotropic processes are chosen due to its simplicity

and interpretability (Banerjee et al., 2014).

For the simulated PSV processes, the parameters are fixed in $\mu = 10$, $E = [\varepsilon(s)]_{s \in D} \sim \mathcal{N}(\mathbf{0}, \Sigma_s)$, where Σ_s is determined by using the most common spatial auto-covariance functions based on isotropic structures:

- The exponential auto-covariance function:

$$C(s_1, s_2) = \begin{cases} \tau^2 e^{-\phi \|s_2 - s_1\|}, & \text{if } \|s_2 - s_1\| > 0 \\ \tau^2 + \sigma^2, & \text{otherwise} \end{cases};$$

  where σ^2 and τ^2 are known as the nugget and partial sill parameters and ϕ is called the decay parameter. The decay parameter is related with the degree of smoothness of the auto-covariance function. The nugget and partial sill parameters are characteristic of every well-defined spatial auto-covariance function. The nugget parameter is related to the nugget effect, which is attributed to measurement errors or spatial sources of variation at distances smaller than the sampling interval or both. For this reason the nugget parameter is viewed as a "nonspatial effect variance". On the other hand, the partial sill parameter is a measure of the observed auto-covariance at an infinitesimally small separation distance and is viewed as a "spatial effect variance" (for more ditails see Banerjee et al., 2014). For purposes of this simulation, the values $\tau^2 = 0.25$, $\sigma^2 = 0.1$ and $\phi = 1$ are taken. Hereafter, this simulated process is referred as the *PSV-Exponential process*.

- The Mátern auto-covariance function:

$$C(s_1, s_2) = \begin{cases} \frac{\tau^2}{2^{v-1}\Gamma(v)} (2\sqrt{v}\, \|s_2 - s_1\|\, \phi)^v K_v(2\sqrt{v}\, \|s_2 - s_1\|\, \phi), & \text{if } \|s_2 - s_1\| > 0 \\ \tau^2 + \sigma^2, & \text{otherwise} \end{cases};$$

  where $\Gamma(.)$ is the standard gamma function and $K_v(.)$ represents the modified Bessel function of second kind with order v (for more details see Banerjee et al., 2014). The parameter $\phi > 0$ controls the rate of decay of the auto-covariance according to the distance $\|s_2 - s_1\|$, and the parameter v controls smoothness of the spatial process (Banerjee et al., 2014). For this

simulation study, the values $v = 1$, $\tau^2 = 0.25$ and $\sigma^2 = 0.1$ are used. ϕ takes the solution of the equation $C(s_1, s_2) = 0$ when $\|s_2 - s_1\| = 0.15$. From now on, this simulated process is called the *PSV-Mátern process.*

- The spherical auto-covariance function:

$$C(s_1, s_2) = \begin{cases} 0, & \text{if } \|s_2 - s_1\| \geq 1/\phi \\ \tau^2 \left(1 - \frac{3}{2} \|s_2 - s_1\| \phi + \frac{1}{2}(\|s_2 - s_1\| \phi)^3\right), & \text{if } 0 < \|s_2 - s_2\| \leq 1/\phi \cdot \\ \tau^2 + \sigma^2, & \text{otherwise} \end{cases}$$

  For this simulated process, the next values are used: $\tau^2 = 0.25$, $\sigma^2 = 0.1$ and ϕ as the solution of the equation $C(s_1, s_2) = 0$ when $\|s_2 - s_2\| = 0.15$. Onwards, this simulated process is referred as the *PSV-Spherical process.*

4.2 The pure temporal variability process

The pure temporal variability process (PTV process, in short) consists in a simulated autoregressive of order 1 process of the form

$$y(s,t) = \mu + \rho \cdot y(s, t-1) + \varepsilon(t) \;\; for\; s \in D \; and \; t = 1, 2, \ldots, 250; \tag{4.2}$$

where μ represents the structured temporal mean, ρ is the $AR(1)$ coefficient and $\varepsilon(s)$ denotes the time residual field. This process does not have a spatial variability component and only the temporal variability is present.

For the simulated PTV processes, the parameters are fixed in $\mu = 10$, $\varepsilon(t) \overset{\text{iid}}{\sim} \mathcal{N}(0, 0.01)$ and, in order to study the influence in the forecast performance of the $AR(1)$ coefficients, the following values are used:

- $\rho = 0.50$, this simulated scenario is called in short the PTV - $\rho = 0.50$ *process.*
- $\rho = 0.75$, this simulated scenario is called in short the PTV - $\rho = 0.75$ *process.*

- $\rho = 0.95$, this simulated scenario is called in short the PTV - $\rho = 0.95$ *process.*

4.3 The spatio - temporal variability process

The spatio - temporal variability process (STV process, in short) consists in a simulated process based on a dinamyc structure of the form

$$y(s,t) = O(s,t) + \varepsilon(t)$$

$$O(s,t) = \mu + \rho \cdot O(s,t-1) + \eta(s);$$

for $s \in D$ and $t = 1, 2, \ldots, 250$. This hierarchical AR model has μ as the structured spatio-temporal mean, ρ as the unknown temporal correlation parameter, $\varepsilon(t)$ as the time residual field and $\eta(s)$ as the space redisual field. Note that the temporal and spatial variability are merged in one process through the hierarchical structure.

For the simulated STV processes, the parameters are fixed in $\mu = 10$, $\rho = 0.95$, $E_T = [\varepsilon(t)]_{t=1,\ldots,250} \overset{\text{iid}}{\sim} \mathcal{N}(\mathbf{0}, 0.1 \cdot \mathbb{I}_{250})$ and $E_S = [\varepsilon(s)]_{s \in D} \sim \mathcal{N}(\mathbf{0}, \Sigma_s)$ where Σ_s has a structure according to the spatial auto-covariance functions used in section 4.1:

- The exponential auto-covariance function, hereafter this process is called the STV - *Exponential process.*
- The Mátern auto-covariance function, hereafter this process is called the STV - *Mátern process.*
- The spherical auto-covariance function, hereafter this process is called the STV - *Spherical process.*

The simulated processes are summarized in table 4.1. In total, there are nine simulated scenarios. With the goal of obtaining statistically reliable estimates of the results, each scenario is replicated 50 times. Next section shows the most significant results.

Table 4.1: Table with the nine simulated processes used in the simulation study.

	PSV processes	PTV processes	STV processes
Simulated processes	PSV - Exponential	PTV - $\rho = 0.50$	STV - Exponential
	PSV - Mátern	PTV - $\rho = 0.75$	STV - Mátern
	PSV - Spherical	PTV - $\rho = 0.95$	STV - Spherical

4.4 Simulation results

This section shows the results to verify the forecast performance of the three proposed models in simulated data. For this purpose, nine scenarios were simulated with 50 replications each one. To get a simple summary of these data, simple statistics as the average and standard deviation are calculated. Figure 4.1 shows the averaged maps, with the standard deviation contoured, of one replication in the nine simulated scenarios. The first three maps (figures 4.1a, 4.1b and 4.1c) correspond to PSV processes; figures 4.1d, 4.1e and 4.1f correspond to PTV processes and the last three (figures 4.1g, 4.1h and 4.1i) belong to STV processes. The PSV processes have an average value ranging from 9.97 to 10.03 and they have deviations lower than 0.2. For its part, the PTV processes exhibit different range of means and deviations. For the $PTV - \rho = 0.50$ process, the mean value ranges from 19.95 and 20.05 with deviations lower than 0.12; the $PTV - \rho = 0.75$ process presents mean values ranging from 39.9 to 40.1 with deviation lower than 0.18; and the $PTV - \rho = 0.95$ process shows a spatial mean varying from 199.4 to 200.5 with deviations less than 0.45. Finally, the STV processes show similar ranges of mean, fluctuating between 199.5 and 200.3, but they present greater deviations than the PSV and PTV processes (varying from 0.45 to 0.6).

In order to evaluate the prediction performance in-sample of the three proposed models, an estimation of the global median MAPE in-sample is calculated in the nine simulated processes. To achieve this, given a replication of any simulated process, the MAPE in-sample is calculated period by period, by comparing the predicted map with the real map, over the 250 units of time. Then, the median of these MAPE values is calculated and represents a global measure of prediction performance in-sample for the replication. Finally the median of MAPEs is found for the 50 replications process and the median of these value is considered as the global median MAPE

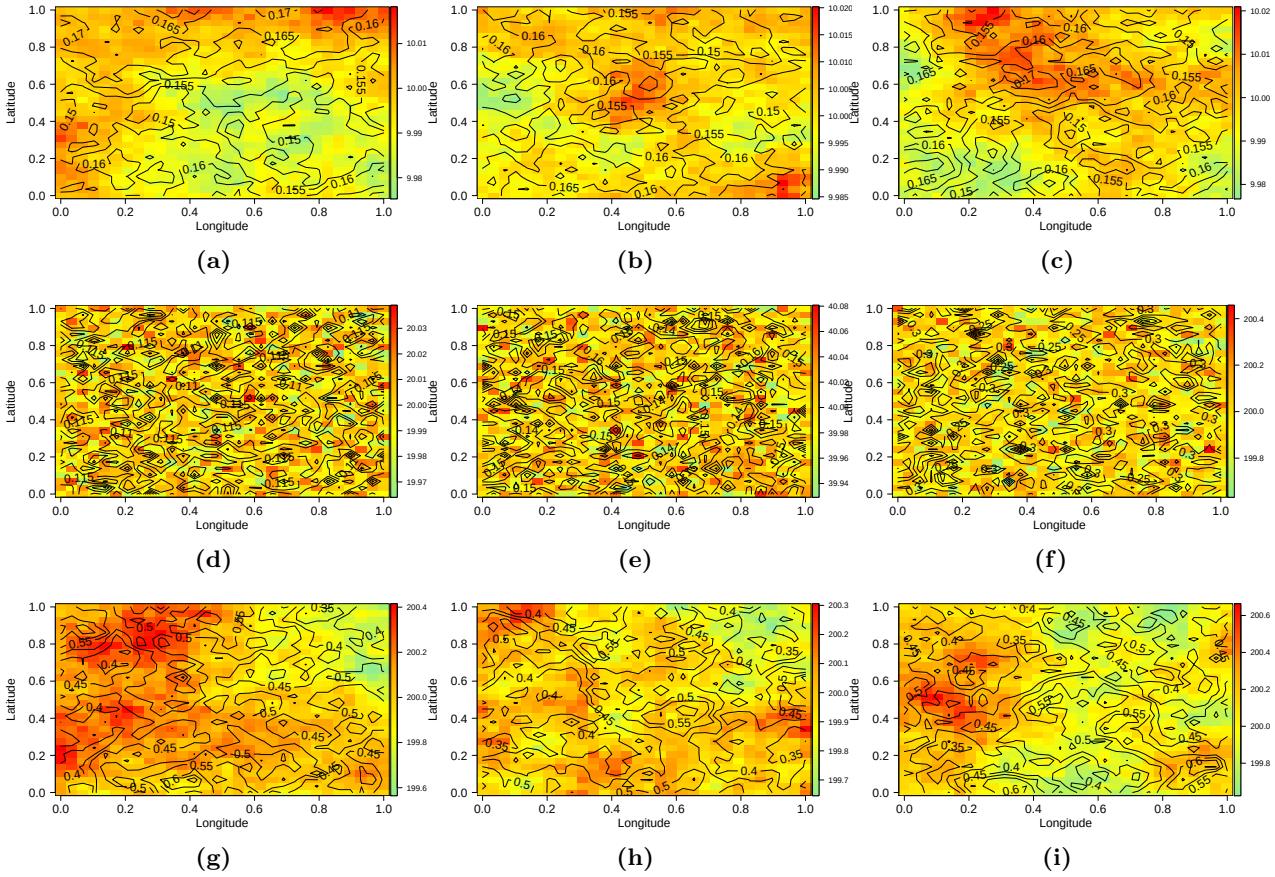

Figure 4.1: Spatial distribution of averaged maps of one replication in scenario: (a) PSV - Exponential, (b) PSV - Matérn, (c) PSV - Spherical, (d) PTV - $\rho = 0.5$, (e) PTV - $\rho = 0.75$, (f) PTV - $\rho = 0.95$, (g) STV - Exponential, (h) STV - Matérn, (i) STV - Spherical. The standard deviation is contoured in black.

in-sample for each process. Table 4.2 exhibits the global median MAPE in-sample for the nine simulated processes using the three proposed models. As it is observed, for the SPCA model, the values of the global median MAPE range from 0.6% to 1.3%. For the TPCA and STPCA model, the values are slightly lower, with global median MAPEs varying between 0.001% to 0.05%.

To evaluate the forecast performance of the three proposed models, an estimation of the evolution of the global median of the spatial MAPE (figure 4.2) in a forecast horizon of six time units is performed from the nine simulated processes. To do this, in the 50 replications of each process a training period of length 244 is chosen and the last six time units are taken as the forecast val-

Table 4.2: Global median MAPE in-sample of the nine simulated processes using the SPCA, TPCA and STPCA model.

Variability	Simulated process	SPCA model	TPCA model	STPCA model
Pure spatial variability	PSV - Exponential	0.729%	0.045%	0.051%
	PSV - Mátern	0.818%	0.041%	0.043%
	PSV - Spherical	0.767%	0.042%	0.046%
Pure temporal variability	PTV - $\rho = 0.50$	1.233%	0.005%	0.007%
	PTV - $\rho = 0.75$	0.875%	0.004%	0.004%
	PTV - $\rho = 0.95$	0.647%	0.001%	0.001%
Spatio temporal variability	STV - Exponential	0.751%	0.003%	0.004 %
	STV - Mátern	0.782%	0.004%	0.007%
	STV - Spherical	0.707%	0.004%	0.006%

idation period. Then, the spatial MAPE corresponding to the i^{th} replication and the j^{th} forecast horizon is denoted by $\overline{MAPE}_{i,j}$ (for $i = 1, \ldots, 50$; $j = 1, \ldots, 6$) and is calculated as the median of the MAPEs of all the grids in the forecast map. Finally, the global median of the spatial MAPE in the forecast horizon k, denoted by $\overline{\overline{MAPE}}_k$ (for $k = 1, \ldots, 6$), is found as

$$\overline{\overline{MAPE}}_k = Median_{i=1,\ldots,50}(\overline{MAPE}_{i,k})$$

and is interpreted as a global measure of forecast error at a specific forecast horizon position. Figure 4.2 schematizes this process.

Figure 4.3 exhibits the evolution of the global median of the spatial MAPE with a forecast horizon of length six in the nine simulated processes using the SPCA, the TPCA and STPCA model. It can be noted that forecast results belonging to the PSV processes (figures 4.3a, 4.3b and 4.3c) show that the SPCA ans STPCA model provide good global forecast errors (lower than 20% in all the forecast horizon) while the TPCA model displays greater global forecast errors, more specifically, beginning from the fourth forecast horizon position the $\overline{\overline{MAPE}}$ reachs values ranging from 20% to 160%. On the other side, in the PTV processes (figures 4.3d, 4.3e and 4.3f) the SPCA model has the worst forecast results for these processes, attaining $\overline{\overline{MAPE}}$s that exceed 80% in some

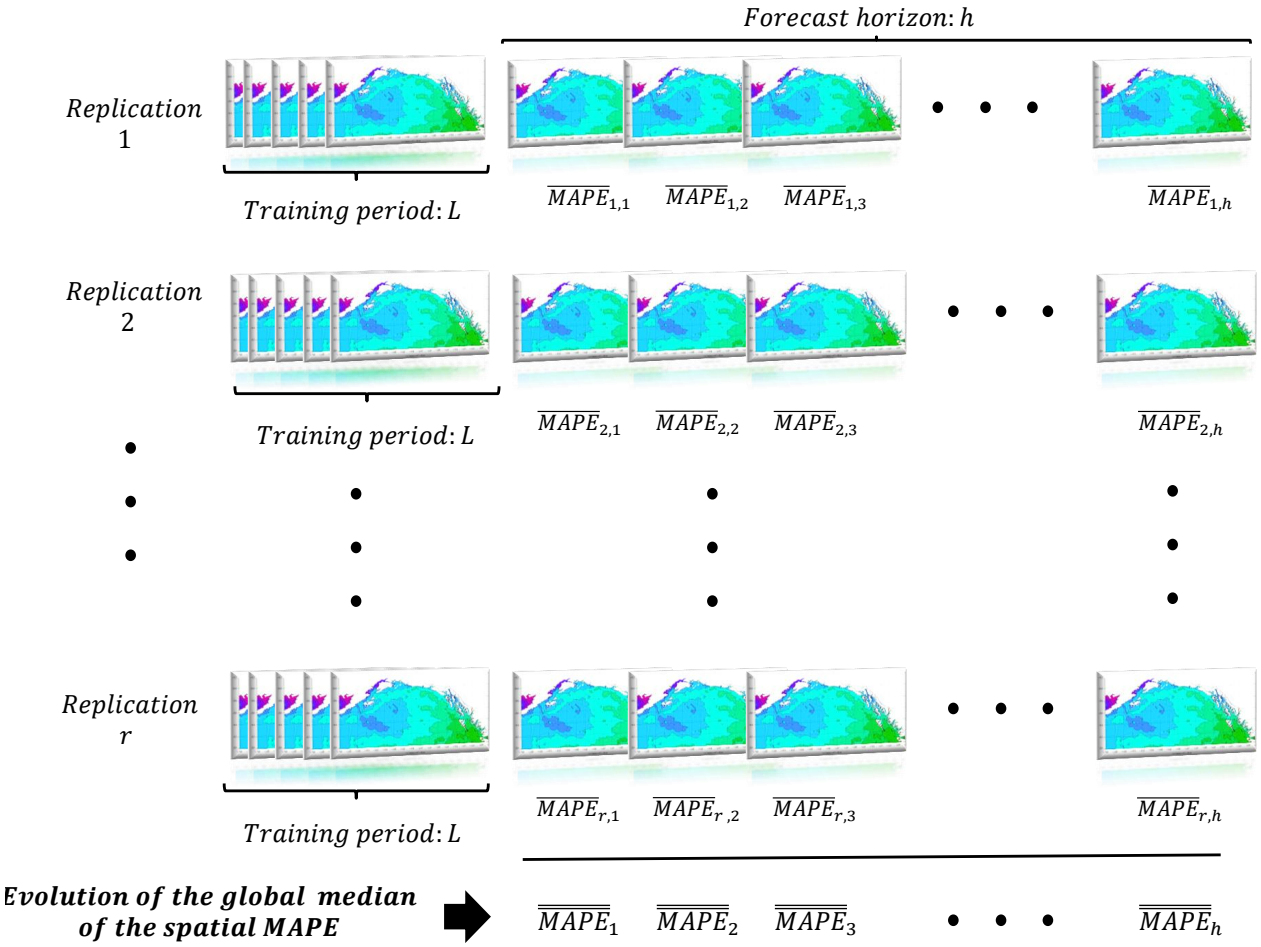

Figure 4.2: An sketch of how to calculate the evolution of the global median of the spatial MAPE with a forecast horizon of length six.

cases. The TPCA model, for its part, shows the lowest global forecast errors in the $PTV - \rho = 0.50$ and $PTV - \rho = 0.75$ processes, but not in the $PTV - \rho = 0.95$ where the STPCA model has the best forecast performance in all the forecast horizon. Finally, in the STV processes (figures 4.3g, 4.3h and 4.3i) it is observed that the SPCA and STPCA models obtain the best forecast results. In the STV processes, the TPCA model presents the worst forecast performance, displaying $\overline{\overline{MAPE}}$s from the third forecast horizon position ranging from 60% to 160%.

To analyze the forecast performance from a spatial perspective, the spatial distribution of the median MAPE in the six forecast periods is calculated for each simulated process and using the

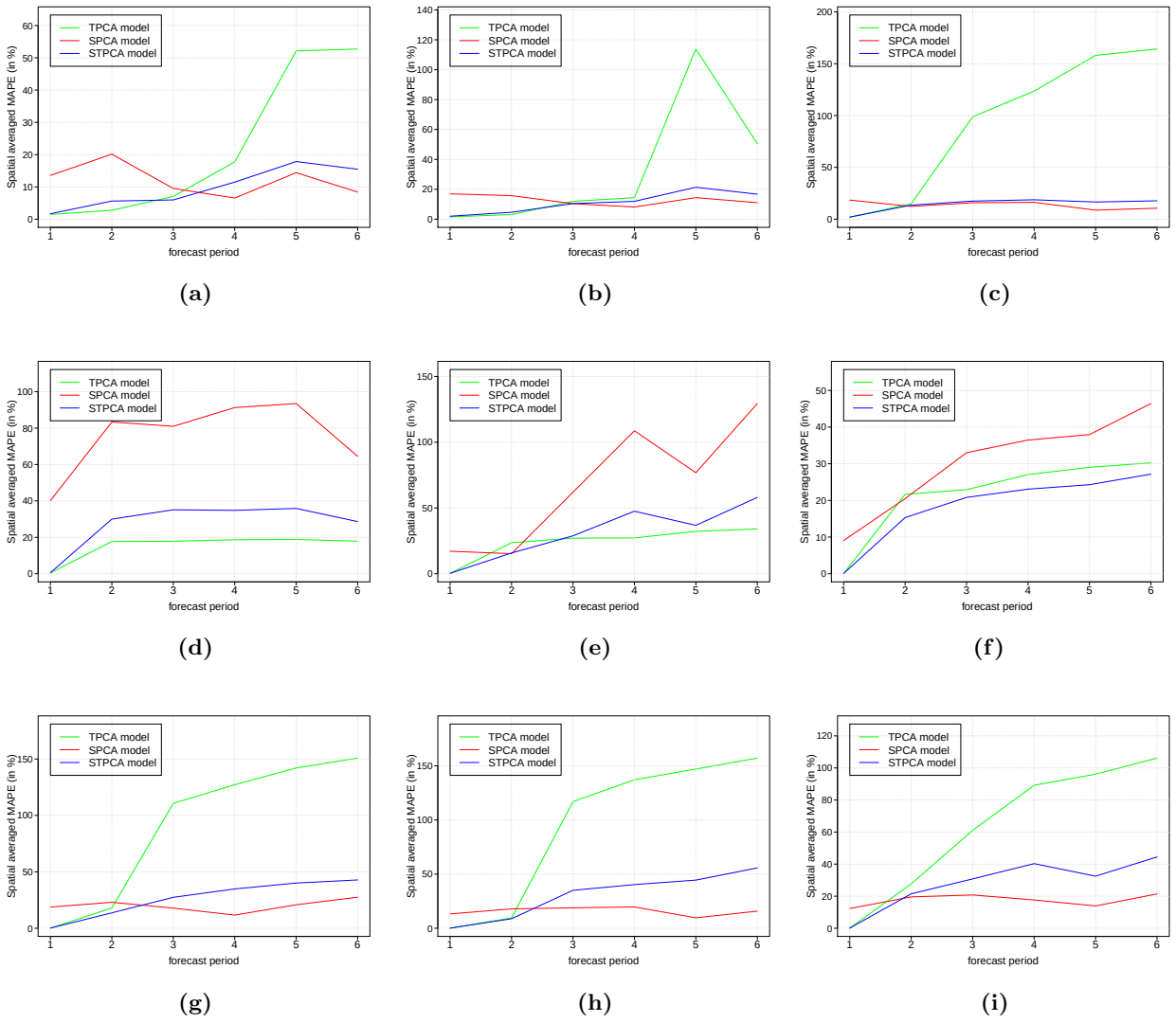

Figure 4.3: Evolution of the global median of the spatial MAPE using the SPCA, TPCA and STPCA model in the simulated processes: (a) PSV-Exponential, (b) PSV-Matérn, (c) PSV-Spherical, (d) PTV-$\rho = 0,5$, (e) PTV-$\rho = 0,75$, (f) PTV-$\rho = 0,95$, (g) STV-Exponential, (h) STV-Matérn, (i) STV-Spherical.

three proposed models. For each simulated process, the spatial distribution of the median MAPE is a map that results from calculating the median of the MAPE maps estimated in the 50 replications. In this case, the PSV - Spherical, the PTV - $\rho = 0.5$ and the STV - Mátern processes are chosen to

represent the general results found for the PSV, PTV and STV processes respectively. The spatial distribution of the averaged MAPE of the other processes can be found in appendix A.

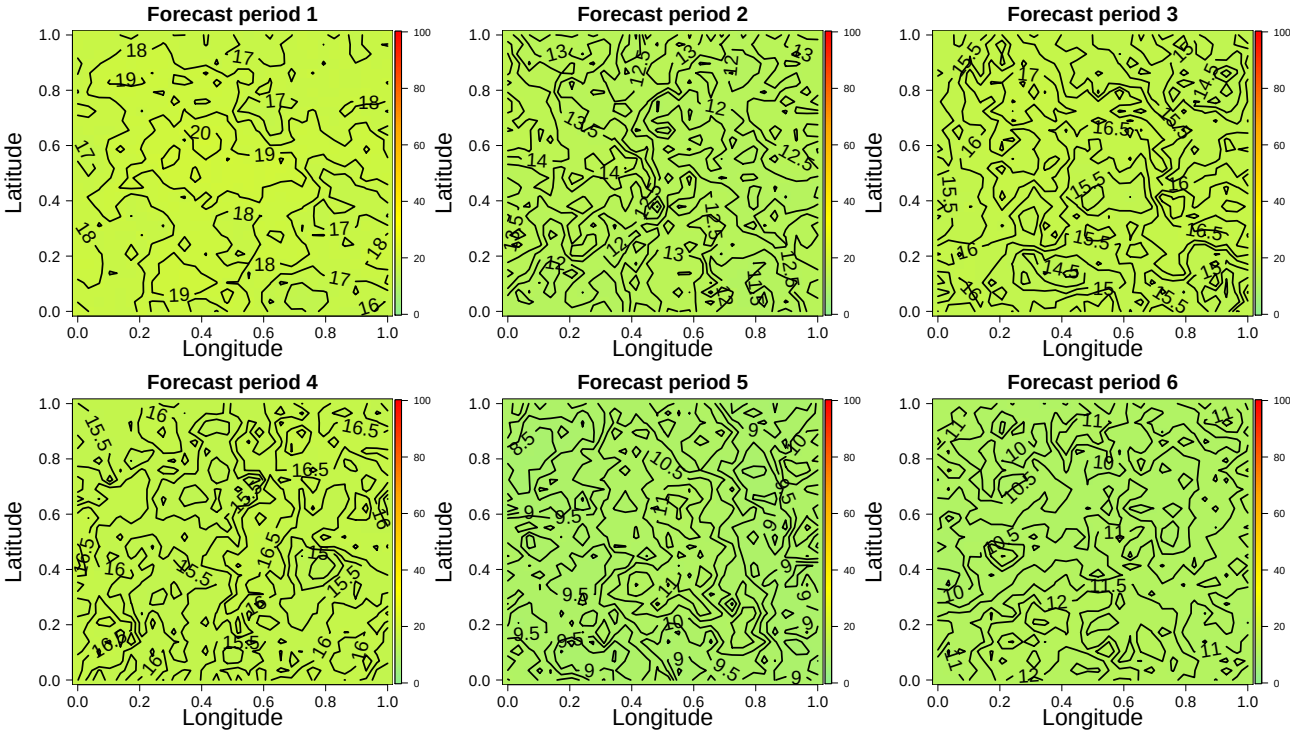

Figure 4.4: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the PSV- Spherical process using the SPCA model.

Figures 4.4 and 4.5 exhibit the spatial distribution of the median MAPE in all the forecast horizon for the PSV-Spherical process using the SPCA and TPCA model respectively. As it is observed, using a SPCA model gives spatially distributed MAPEs that do not surpass the 20% in all the forecast horizon. This result is consistent with figure 4.3c, where the evolution of the global median of the spatial MAPE shows values lower than 20%. On the other hand, using a TPCA model gives contrary results. As shown in figure 4.5, only in the first and second forecast periods the spatial distribution of the median MAPE presents low values (less than 20%) and from the third forecast period onwards, the values increase dramatically in different specific regions, with median MAPEs greater than 60%. The results of the PSV - Exponential and PSV - Mátern processes (figures A.1, A.2, A.4 and A.5) manifest identical results. The spatial distribution of the

median MAPE in the PSV processes using the STPCA model (figures A.3, A.6 and A.7) behave as a ponderation of the error maps observed in the SPCA and TPCA model.

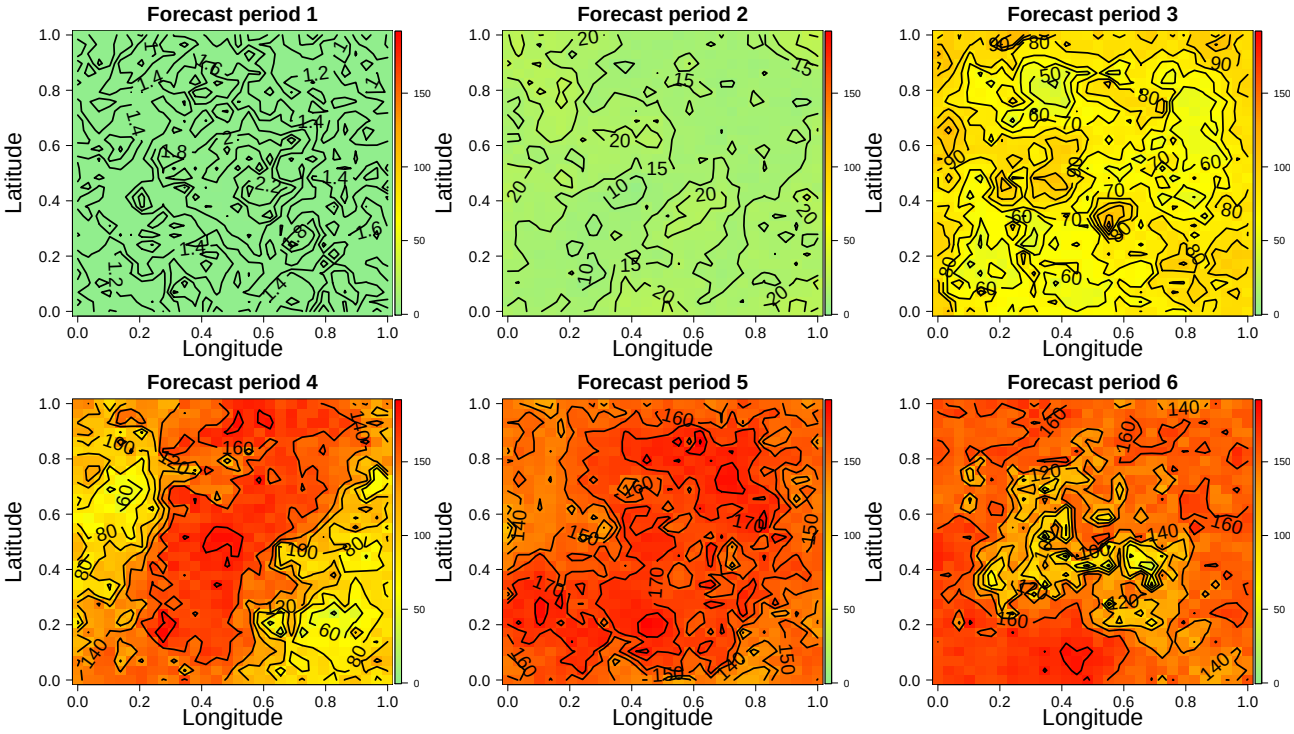

Figure 4.5: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the PSV- Spherical process using the TPCA model.

Figures 4.6 and 4.7 display the spatial distribution of the median MAPE in all the forecast horizon for the PTV - $\rho = 0.5$ process using the SPCA and the TPCA model respectively. Looking at the results, when a SPCA model is applied over the PTV - $\rho = 0.5$ process, it gives spatially distributed MAPEs that augment gradually as the forecast period increases and showing values greater than 40% since the first forecast period. However, when a TPCA model is used the results are totally different. Figure 4.7 exhibits a uniform spatial distribution of the averaged MAPE in all the forecast horizon and with values lower than 20%. The results corresponding to the PTV - $\rho = 0.75$ (figures A.9 and A.10) exhibit identical results. The results corresponding to the PTV- $\rho = 0.95$ process (figures A.12, A.13 and A.14), show that the STPCA model offers the best spatial performance forecast in all the forecast horizon with respect to the SPCA and TPCA models.

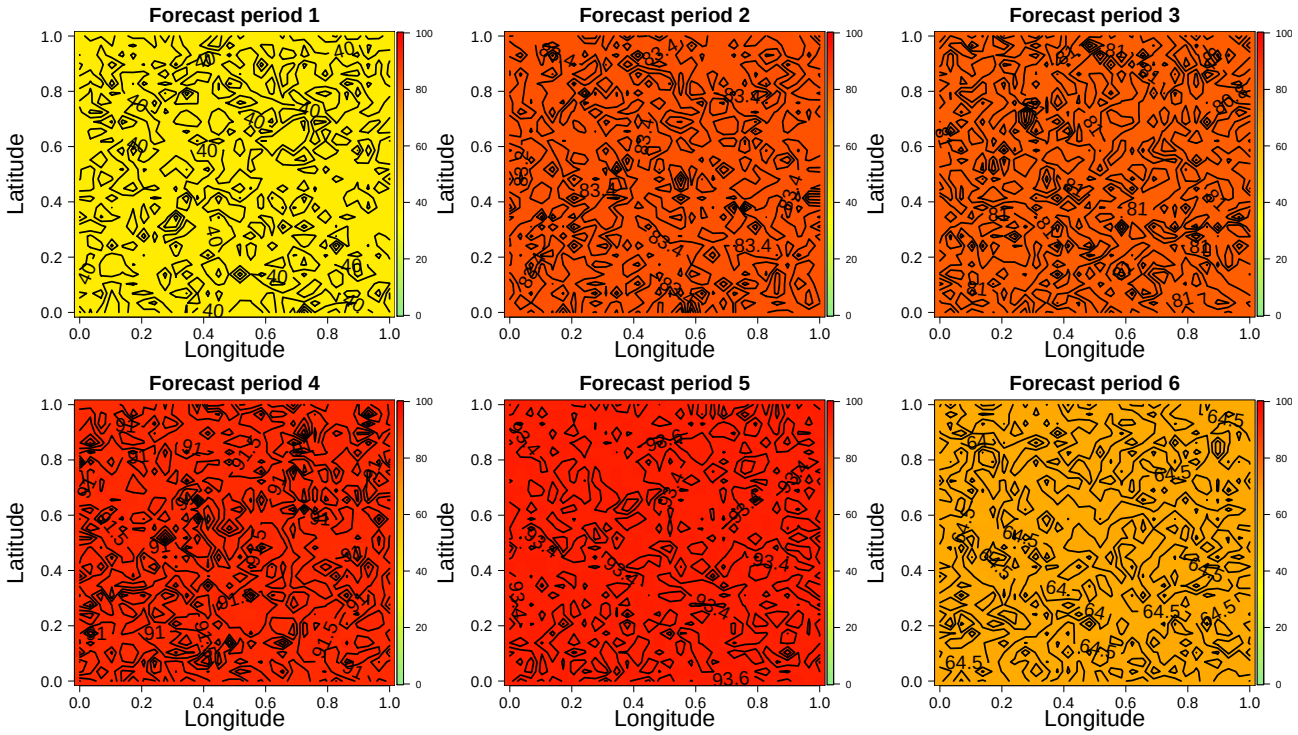

Figure 4.6: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the PTV - $\rho = 0.5$ process using the SPCA model.

With regard to the STV processes, figures 4.8, 4.9 and 4.10 show the spatial distribution of the median MAPE in all the forecast horizon for the STV-Mátern process using the SPCA, the TPCA and the STPCA model respectively. When the SPCA model is used, the spatial distribution of the median MAPE (figure 4.8) shows a uniform distributions and with MAPE values lower than 20%. For its side, when the TPCA model is applied the spatial distribution of the median MAPE (figure 4.9) presents values that augment gradually in specific regions, as the forecast period increases. The MAPE values begin with low quantities in the two first forecast periods (with values lower than 10%) and they increase gradually over all the map, reaching values of 200%. Finally, when one performs the STPCA model, the spatial distribution of the median MAPE (figure 4.10) has the similar pattern observed in the TPCA model, but with a decrease in the MAPE values.

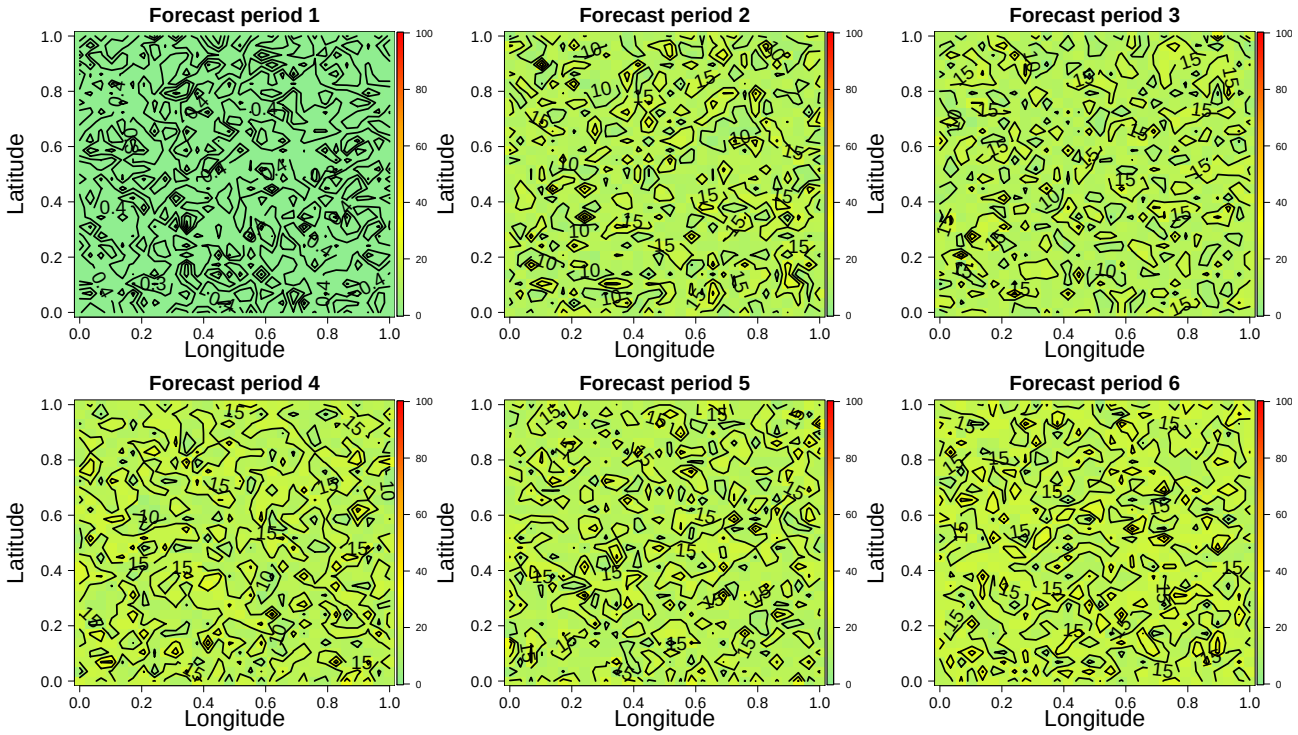

Figure 4.7: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the PTV - $\rho = 0.5$ process using the TPCA model.

4.5 Final comments

This chapter has described three new methodologies - the SPCA, TPCA and STPCA models - for modeling spatio-temporal raster datasets and has developed a simulation study to evaluate the prediction and forecast performance of the proposed models. The results for the prediction of raster maps in-sample (table 4.2) has shown that the three proposed models have an excellent prediction performance in-sample. Particularly, if predictions with greater accuracy are needed, one can use the TPCA or the STPCA model which, according to the simulated scenarios, present global median MAPE lower than 0.1%.

The results related to the forecast performance of the proposed models in the simulated scenarios have shown that, in general, the models provide good forecasts in a short-term forecast period. To analyze the global forecast performance of the proposed models, the median of the $\overline{\overline{MAPE}}$s are calculated in the nine simulated processes using each model. This measure can be seen as a global

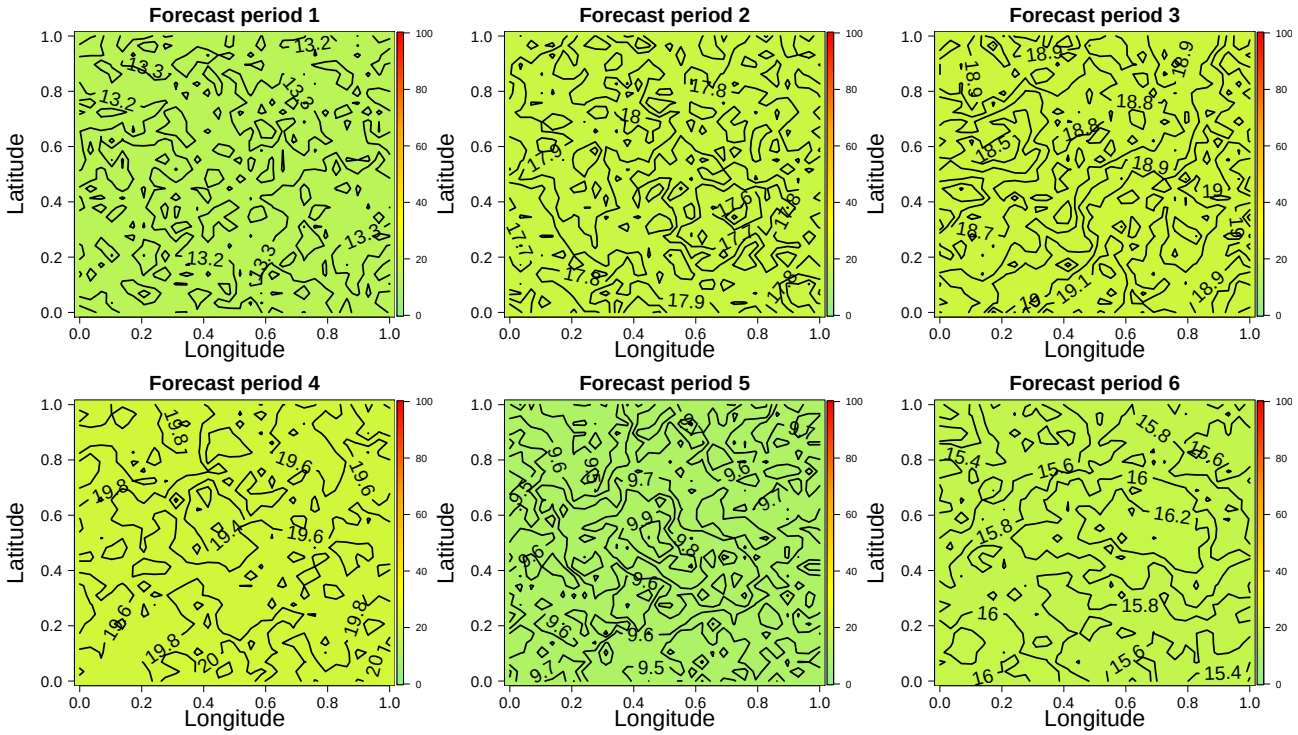

Figure 4.8: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the STV - Matérn process using the SPCA model.

measure of forecast performance in all the forecast horizon. Table 4.3 shows these values and an analysis separated by the type of process according to the type of predominant variability they posses is carried out:

1. *For the PSV - processes:* As it is observed, the STPCA model has the best performances for the PSV-Exponential and PSV-Mátern processes, except for the PSV-Spherical process, where the SPCA model has the best performance. For the PSV - Exponential and PSV - Spherical the TPCA model gives the worst global forecast results except for the PSV-Mátern process. Looking at the evolution of the global error maps using the TPCA model in PSV processes (figures A.2, A.5 and 4.5) it is noted that while the forecast horizon moves away, the error increases gradually in specific spatial regions. Thus, when the spatial variability is predominant, the TPCA model does not capture the necessary patterns corresponding to the spatial variation that allows to get a good forecast performance. Also, remark that when

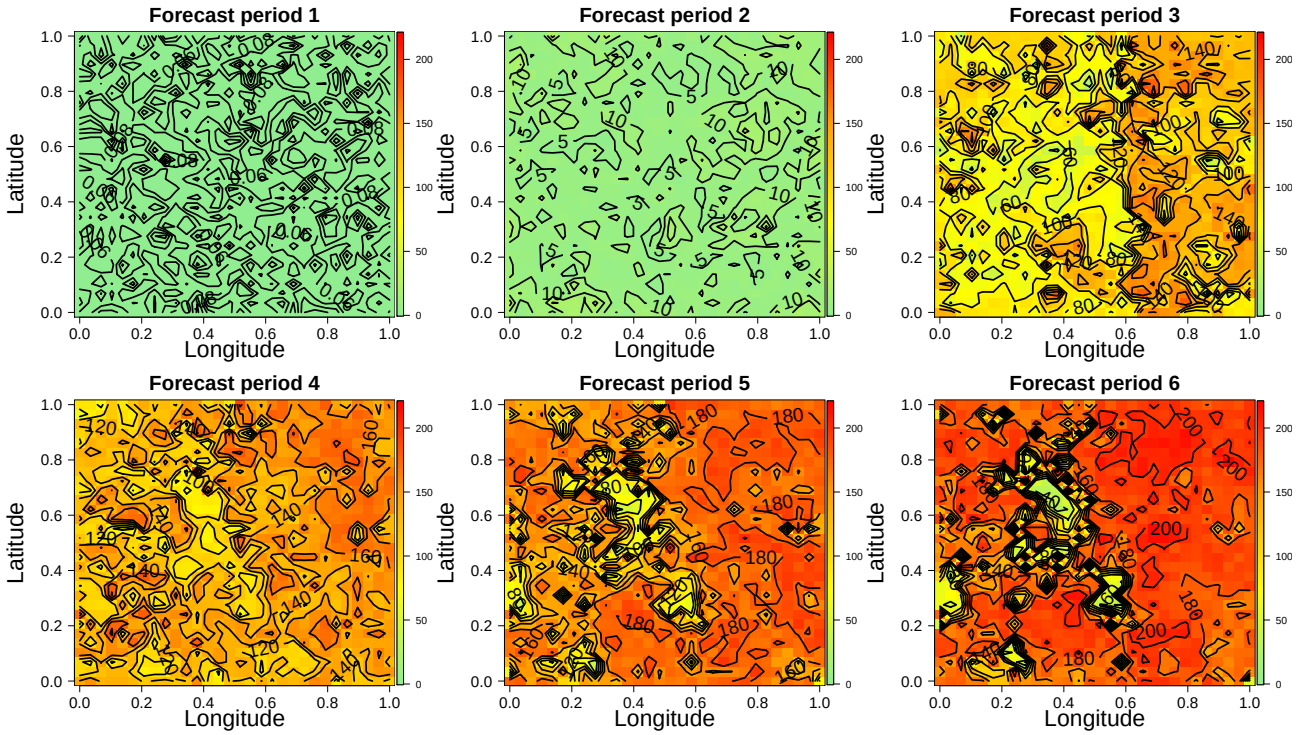

Figure 4.9: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the STV - Matérn process using the TPCA model.

the lines of evolution of $\overline{\overline{MAPE}}$ cross at the middle part of the forecast horizon, then the STPCA model gives an improvement of the global forecast results. Thereby, in these cases the STPCA model captures temporal and spatial patterns, in a balanced way, in order to improve the overall forecast.

2. *For the PTV - processes:* Table 4.3 displays that, for these processes, the TPCA model exhibits the best forecast performance except for the $PTV - \rho = 0.95$ process, where the STPCA model is the model with the best result. On the other hand, the SPCA model produce the worst global forecast results without exceptions. Moreover, if the evolution of the global error maps using the SPCA model in PTV processes (figures 4.6, A.9 and A.12) is seen, it is observed that after a short forecast horizon (no more than two units of time) the error is dramatically high over all the spatial region. So, when the temporal variability is predominant, the SPCA model does not capture the necessary patterns corresponding to the

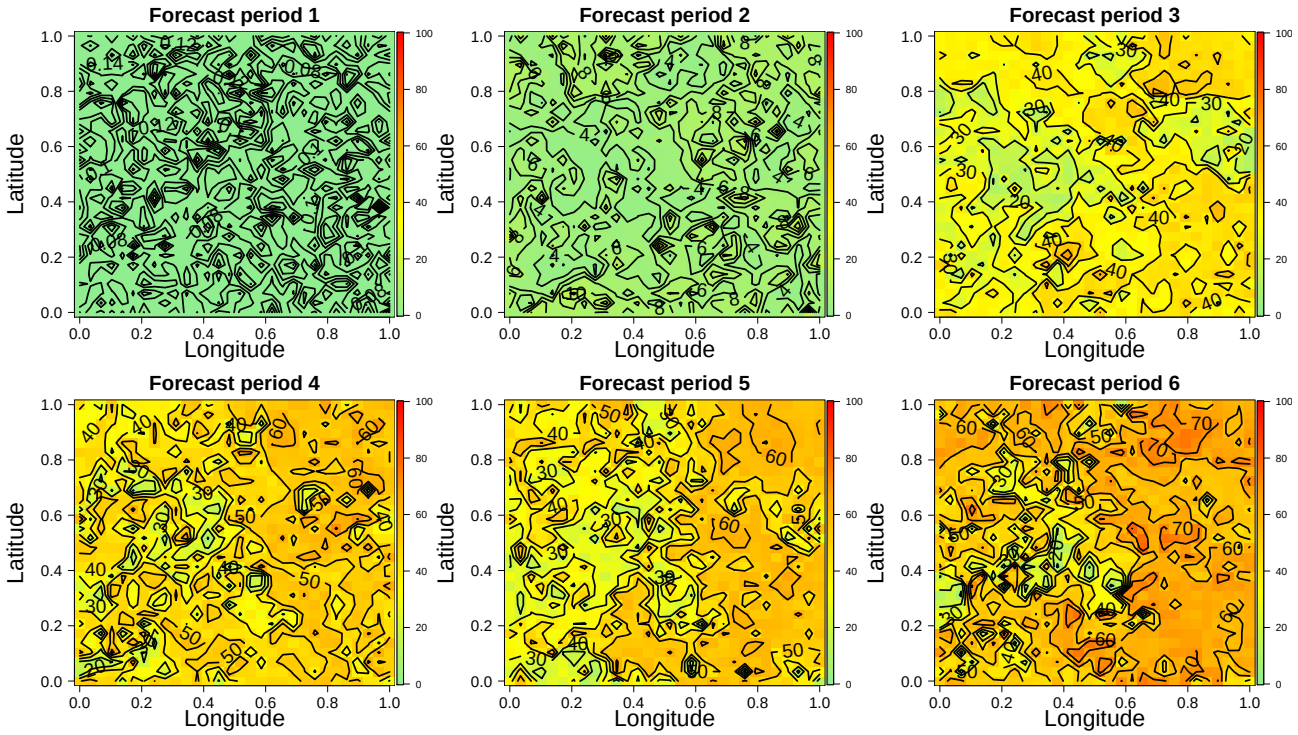

Figure 4.10: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the STV - Matérn process using the STPCA model.

temporal variation that allows to get good forecast results.

3. *For the STV - processes:* Looking at table 4.3, it is remarkable that for theses processes the SPCA model produces the best global forecast performances and, on the other hand, the TPCA model provides the worst. The STPCA model shows an intermediate performance. Taking a look at the the evolution of the global error maps using the STPCA model in STV processes (figures A.17, 4.10 and A.20), it is noted that the error increases in certain spatial regions as the forecast horizon is extended. This happens since the STPCA model gives a forecast taking a convex combination of the forecast maps given by the SPCA and TPCA models. As the TPCA model produces maps with great forecast errors, this feature is inherited to the STPCA model in a certain way.

Based on the previous discussion about the global forecast performance of each model, it is

Table 4.3: Median of the $\overline{\overline{MAPE}}$s in the nine simulated processes using the SPCA, TPCA and STPCA model.

Variability	Process	SPCA model	TPCA model	STPCA model
Pure spatial variability	PSV - Exponential	11.532%	12.406%	8.725%
	PSV - Mátern	15.577%	13.284%	11.611%
	PSV - Spherical	14.096%	111.194%	17.009%
Pure temporal variability	PTV - $\rho = 0.5$	82.164%	17.851%	32.400%
	PTV - $\rho = 0.75$	69.409%	27.145%	32.801%
	PTV - $\rho = 0.95$	34.732%	24.982%	21.939%
Spatio temporal variability	STV - Exponential	19.796%	118.940%	31.124%
	STV - Mátern	16.837%	127.007%	37.668%
	STV - Spherical	18.595%	75.129%	31.640%

important to determine under what statistical conditions one model would be preferable than the others for forecasting purposes. Following this path, table 4.4 summarizes the median "importance" weights of the SPCA and TPCA model for each simulated process. For each simulated process, the median "importance" weights are calculated as the median of all the "importance" weights of all the forecast horizon. Note that the median "importance" weights corresponding to both models are complementary since they always sum one. As it is shown in table 4.4, for the PSV processes the SPCA "importance" weights have values greater than 0.53. On the other hand, for the PTV processes the TPCA "importance" weights are predominant with weights greater than 0.55. For the STV processes the SPCA "importance" weights show greater values than 0.66. These results agree with the results described in the previous discussion. Indeed, relating the results, when the SPCA "importance" weight is predominant, with values greater than 0.65, the SPCA model would give better overall forecast results. However, when the TPCA "importance" weights present values greater than 0.6, a TPCA model would have a better overall forecast performance. Finally, in the rest of cases the chose would depend on the evolution of the $\overline{\overline{MAPE}}$s in both models. It is, if the lines of evolution of $\overline{\overline{MAPE}}$s of both models do not intersect (as in figure 4.3d), then the model with better "importance" weight could be the chosen one, but if the lines of evolution of $\overline{\overline{MAPE}}$s

intersect like in figures 4.3a and 4.3b, then the STPCA model would be preferable.

Table 4.4: Median"importance" weights of the SPCA and TPCA model in the nine simulated processes.

Variability	Process	SPCA importance	TPCA importance
Pure spatial variability	PSV - Exponential	0.594	0.406
	PSV - Mátern	0.538	0.462
	PSV - Spherical	0.864	0.136
Pure temporal variability	PTV - $\rho = 0.5$	0.385	0.615
	PTV - $\rho = 0.75$	0.379	0.621
	PTV - $\rho = 0.95$	0.414	0.586
Spatio temporal variability	STV - Exponential	0.712	0.288
	STV - Mátern	0.667	0.333
	STV - Spherical	0.68	0.320

Chapter 5

Empirical application

In this chapter an application of the proposed methodology introduced in Chapter 3 is considered. The monthly sea surface temperature anomalies corresponding to the Tropical Pacific Ocean (see figure 5.1) is analyzed with the objective to obtain accurate forecast maps of this climatological variable.

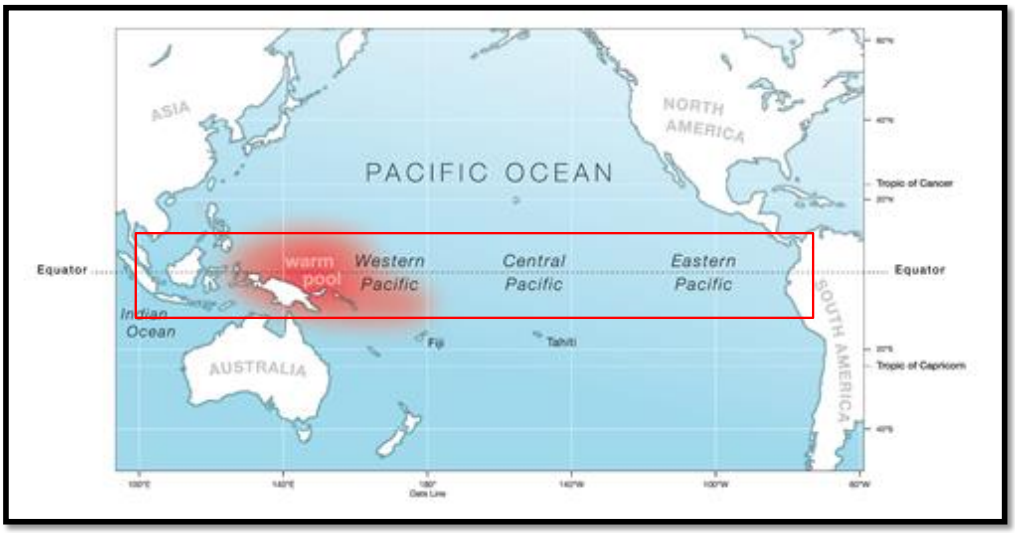

Figure 5.1: Map of the Pacific Ocean indicating, with red lines, the location of the Tropical Pacific Ocean.

The importance of skilful climate forecasts of climatological variables provide useful scientific information to planners and operational agencies to plan and develop contingency measures and strategies to deal with the adverse conditions. In particular, the sea surface temperature (SST) is considered the most influential climatological variable because it provides fundamental information on the global climate system. Even more important, SST data are especially useful for identifying the onset of El Niño and La Niña cycles. During El Niño, temperatures in the Pacific near the equator are warmer than normal. During La Niña, the same area experiences colder than normal ocean temperatures. These cycles are caused by multiyear shifts in pressure and wind speeds, and affect ocean circulation, global weather patterns, and marine ecosystems (Wang and Weisberg, 2000).

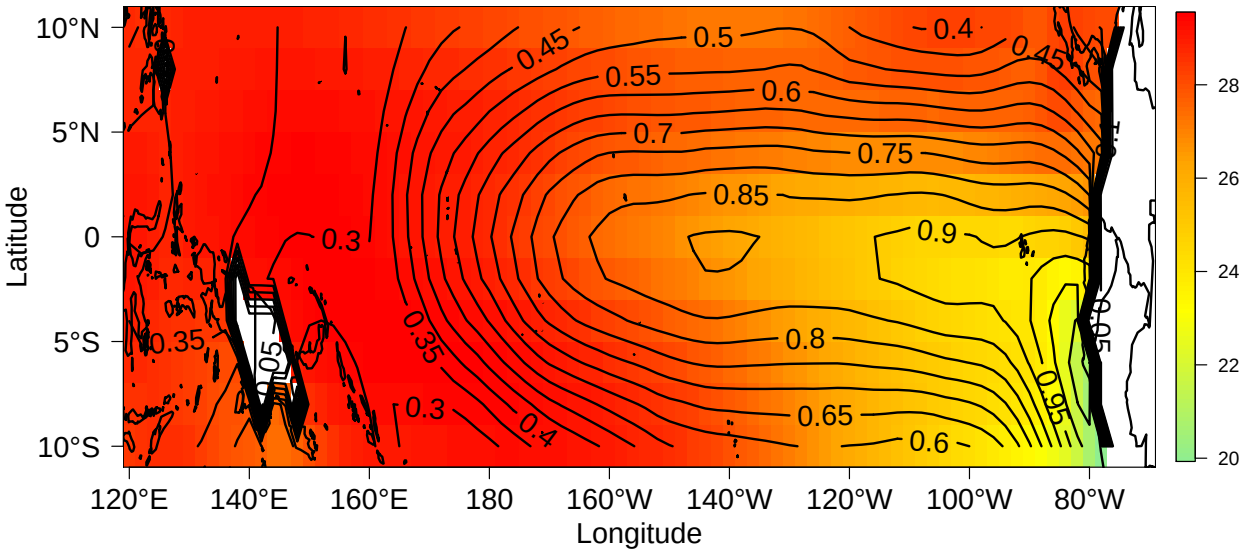

Figure 5.2: Spatial distribution of the averaged *SST* (in °C) of the Tropical Pacific Ocean. The value of the standard deviation of *SST* is contourned in black.

Statistical and dynamical models have been developed to forecasting sea surface temperature anomalies (SSTA) fields over the Tropical Pacific Ocean (for more details see Barnston and Ropelewski, 1992; Chen et al., 1997). In this applicated study the skill potential of the three proposed methods in forecasting SSTA maps is examined. Section 5.1 develops a brief description of the SST dataset and section 5.2 shows the principal results of the application.

5.1 Description of the sea surface temperature dataset

The empirical application is based on the data set estimated by the Extended Reconstructed Sea Surface Temperature (*ERSST*) (Smith et al., 2008). It is a global monthly SST dataset derived from the International Comprehensive Ocean-Atmosphere Dataset (*ICOADS*), published online by U.S. *National Oceanic and Atmospheric Administration* (NOAA[1]). It is produced on a $2° \times 2°$ grid with spatial completeness enhanced using statistical methods (for details see Smith et al., 2008). This monthly analysis begins in January 1854 continuing to the present and includes anomalies computed with respect to a 1971-2000 monthly-climatology.

For purposes of this application, the study is limited to the grid bounded by $10°S - 10°N$ and $120°E - 70°W$ (see figure 5.2), which corresponds to the Tropical Pacific Ocean. This dataset consists in 884 grid nodes in which a monthly SST was estimated, limited to the months from January 1950 and January 2015[2]. SSTA have been calculated by subtracting from each datum the average of the same month of the year along the corresponding time series limited to the period January 1980-December 2010, approximately the same considered by Ashok et al. (2007). Hereafter it is referred as the SSTA dataset.

To get some descriptive summaries of the SST dataset, figure 5.2 shows the averaged (over the complete period) map of SST with the standard deviation contoured. It is observed that the average values of SST fluctuates between 20°C and 32°C. The spatial distribution of the average SST shows the highest values on the west of the Tropical Pacific Ocean but it presents the lowest deviations (with values lower than 0.3°C). In fact, this spatial region is known by climatologists as the *warm pool* (Kug et al., 2009) and is characterized for producing the warmer temperatures, in normal conditions, of the Tropical Pacific Ocean. On the other hand, the region corresponding to the east of the Tropical Pacific Ocean, specifically in the northern coastal of Peru, the SST mean values are the lowest but the deviations are the highest in all the studied region (with values greater than 0.9°C). These highest deviations occur due to El Niño and La Niña events which cause an

[1] https://www.ncdc.noaa.gov/data-access/marineocean-data/extended-reconstructed-sea-surfacetemperature-ersst-v4. Accessed: 15 - 04 - 2017.

[2] This period was chosen to evaluate the forecast performance with the presence of El Niño 2015-2016. This will allow to evaluate the forecast habilities of the model with the presence of extreme values.

increase and decrease of the normal conditions of SST in the central and east-central equatorial Pacific Ocean (Wang and Weisberg, 2000).

5.2 Principal results of the application

The three reviewed methods, the SPCA, TPCA and STPCA model, are applied over the SSTA dataset considering the same parameters of the simulation study. For the forecast horizon, a period of six months is taken. More specifically, the validation forecast period belongs to the months from 2014 - August to 2015 - January. With respect to the error measures, it is used the MAE instead of the MAPE since the SSTA dataset has values very close to zero. Moreover, for the calculus of the "importance" weights, a training period of 768 months is chosen.

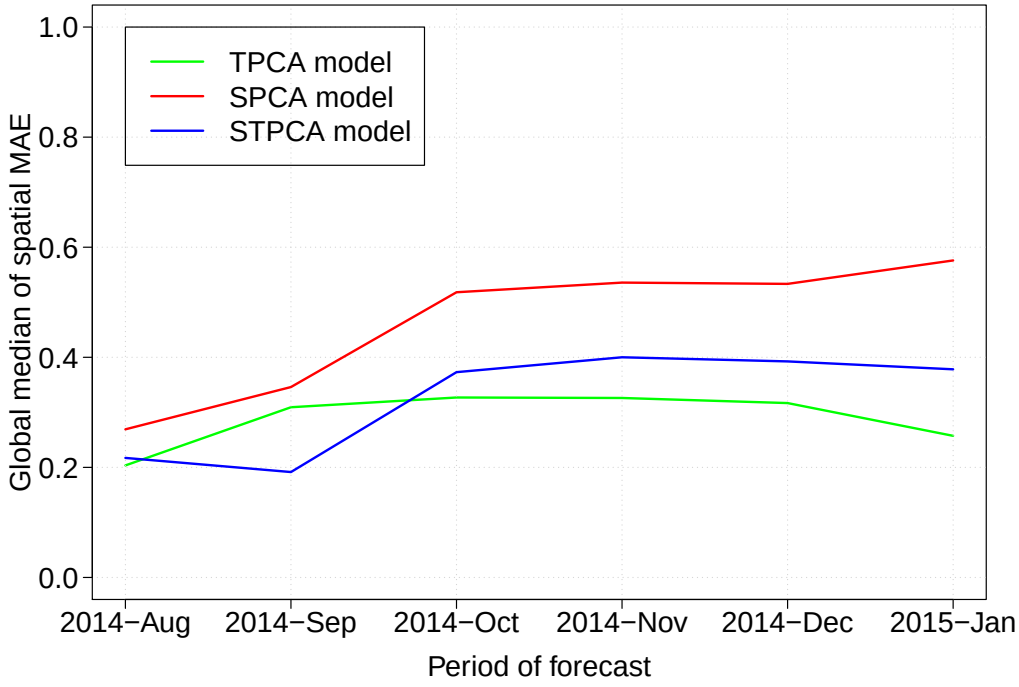

Figure 5.3: Evolution of the median of the spatial MAE using the TPCA, the SPCA and STPCA model over the SSTA dataset.

In order to evaluate the forecast performance, from a temporal perspective, of the proposed models over the SSTA dataset, figure 5.3 exhibits the evolution of the global median of the spatial

MAE in the six months of forecast horizon. As it is observed, the three models provide global absolute errors lower than 0.6°C. For the months of 2014 - August and 2014 - September, the STPCA model displays a better forecast performance than the others. However, since 2014 - October until 2015 - January, the TPCA model shows the best performance compared with the others.

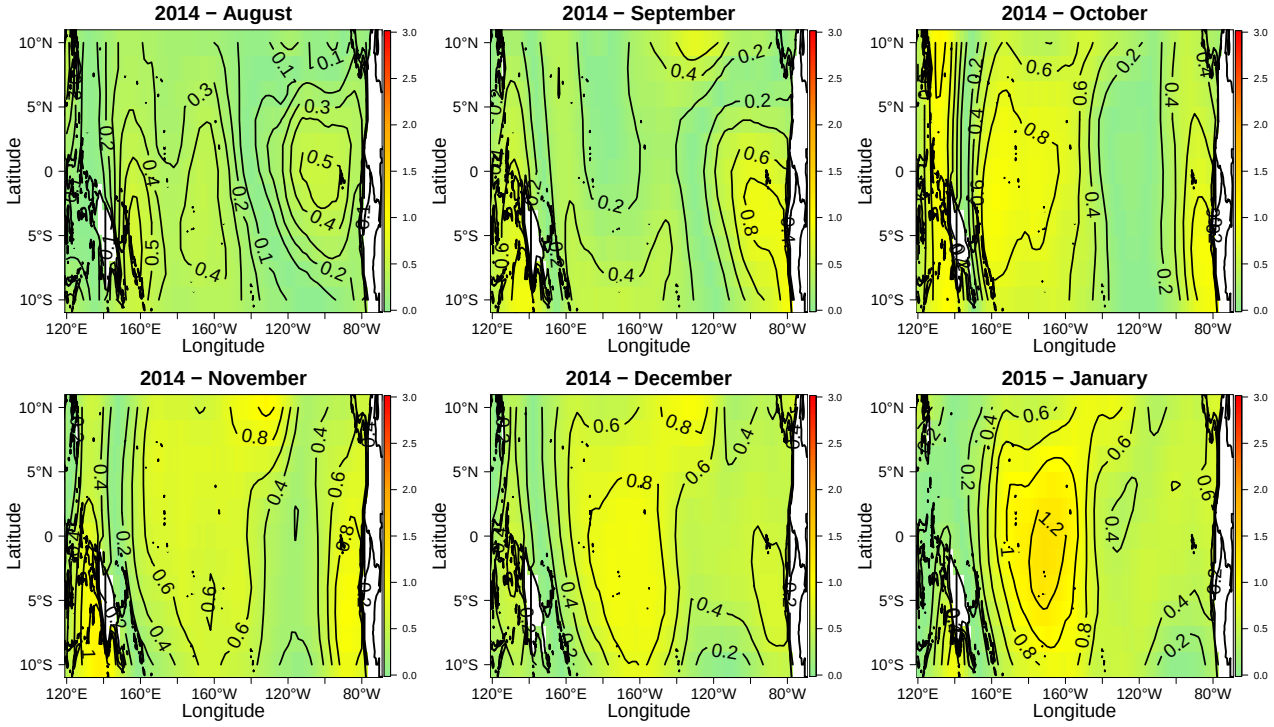

Figure 5.4: Spatial distribution of the median MAE when the SPCA model is used over the SSTA dataset.

Figures 5.4, 5.5 and 5.6 display the spatial distribution of the median MAE in the six evaluated months using the SPCA, TPCA and STPCA model respectively.As it is shown, when the SPCA model is applied over the SSTA dataset, the highest global absolute errors are concentrated over the eastern Tropical Pacific Ocean (with values greater than 0.4°C). This pattern is repeated from 2014-August to 2014 - December. For the months going from 2014-October and 2015-January, the highest global absolute errors are located in the central Tropical Pacific Ocean, with values ranging from 0.4°C to 1.2°C. On the other hand, when a TPCA model is used, the global absolute errors

produce high values over the *warm pool* region and the northeast of the Tropical Pacific Ocean for the months corresponding to 2014 - August, 2014 - September, 2014 - October and 2015 - January. For the months 2014 - November and 2014 - December the highest global absolute errors moves to the eastern Tropical Pacific Ocean, presenting values greater than 0.5°C. Finally, the results provided by the STPCA model shows that for the first two months of the forecast horizon, the global errors have lower values than 0.5°C uniformily over all the studied region. In the month 2014 - October, the errors increase on the west, reaching absolute error values of 1°C. Nevertheless, for the last three months, it is marked an increase of the errors in the eastern and central regions of the Tropical Pacific Ocean (with values greater than 0.5°C). This low forecast performance over these specific regions from 2014 - November to 2015 - January could be due to the occurrence of El Niño 2015 - 2016 event, which according to the NOAA began over those months.

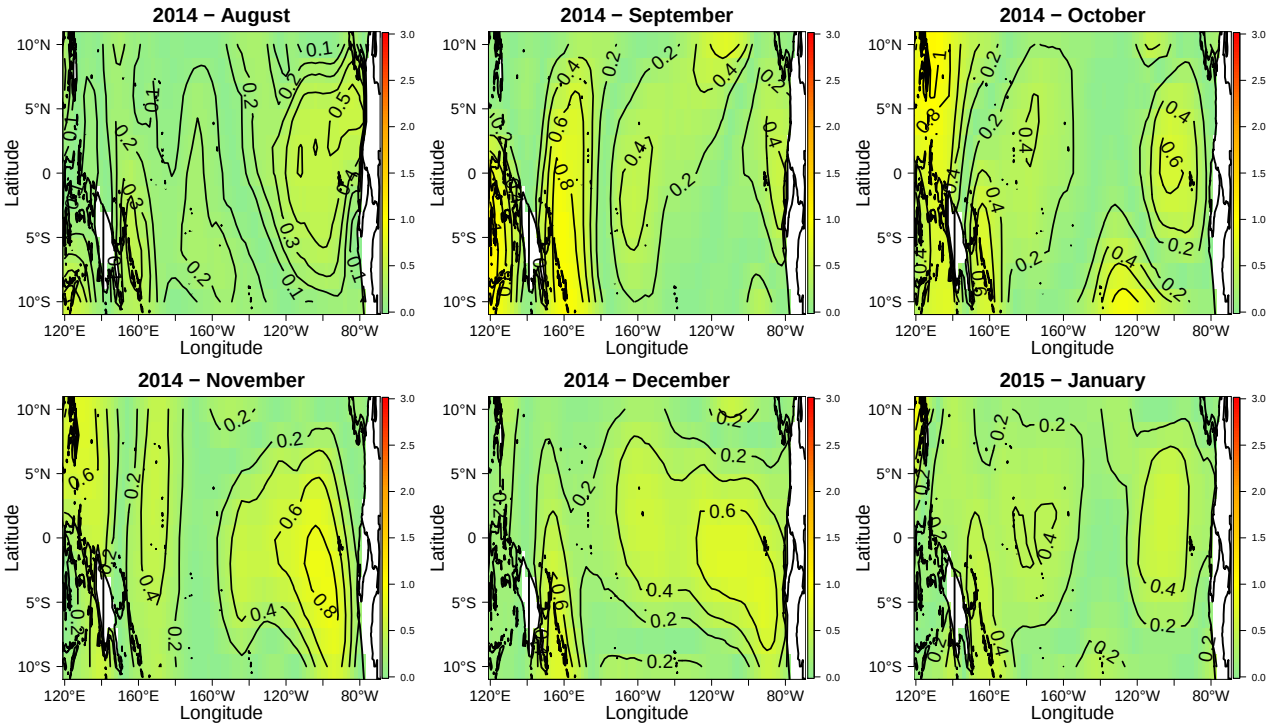

Figure 5.5: Spatial distribution of the median MAE when the TPCA model is used over the SSTA dataset.

In order to get extra information to decide what model to use, table 5.1 shows the global

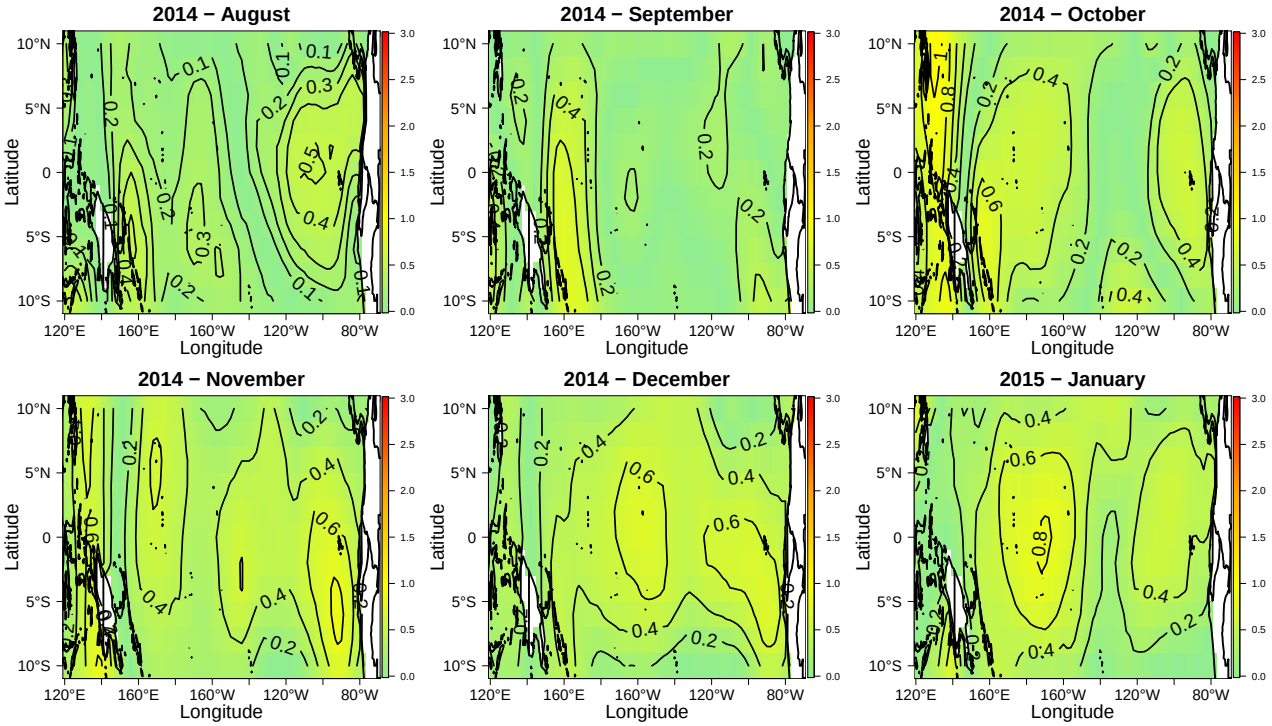

Figure 5.6: Spatial distribution of the median MAE when the STPCA model is used over the SSTA dataset.

prediction and forecast MAE[3] for the SPCA, TPCA and STPCA model. For prediction purposes, the three models exhibit a similar performance. However, if the global forecast performance of the three models is compared, then the TPCA model produce the best results. Furthermore, the TPCA "importance" has a value of 0.6, which gives more evidence in favor of selecting the TPCA model as the model with best performance.

Table 5.1: Global prediction and forecast results of the SPCA, TPCA and STPCA applied over the SSTA dataset.

Model	Global prediction MAE	Global forecast MAE
SPCA model	0.0127	0.526
TPCA model	0.0113	0.313
STPCA model	0.0115	0.376

[3]These values are obtained with the same process followed in the simulation study.

Chapter 6

Conclusions and future developments

The major goal of this thesis was to study the forecasting accuracy of a proposed methodology to model spatio-temporal raster datasets. The devised methodology consists in summarizing the essential spatio-temporal variability through a PCA and then applies a forecasting model over the significant PCs or eigenvectors, ending with a spectral reverse reconstruction and a recoursive algorithm to get the forecast maps. As a result, three different models have been proposed and each one presented diverse advantages and shortcomings:

- *The TPCA model*: It was discovered that in terms of global forecast performance, this model provides good results in datasets with a predominant temporal variability. As a drawback, this model does not achieve to capture the necessary spatial variability patterns to get the best overall forecast results. With respect to the prediction accuracy, this model produces the best prediction performance in-sample independent of the type of variability is predominant in the dataset.

- *The SPCA model*: It was found that when the dataset has a predominant spatial variability, then the model provides the best overall forecast performance. However, when the model has a predominant temporal variability, it does not achieve to capture the necessary temporal variability patterns to get the best forecasting accuracy. With regard to the prediction accuracy, this model provides a good prediction accuracy but it is not better than the produced

by the TPCA model.

- *The STPCA model:* This is an hybrid model that integrates the good prediction and forecasting results of the two previous models in order to improve the global prediction and forecasting performance. It was showed that datasets with similar spatial and temporal variability information, this model improves the forecasting accuracy substantially. Finally, for prediction purposes, the STPCA model produced a similar prediction accuracy than the TPCA model, which is the model with the best prediction accuracy.

It was also found a measure that allows to determine what type of variability (temporal or spatial) is predominant in raster datasets. The "importance" weights can be used as indicators for this purpose. Indeed, the results of the simulation study showed that the TPCA (SPCA) "importance" weights (table 4.4) could be interpreted as a measure of the amount of temporal (spatial) variability presented in the dataset. Thus, if the global SPCA (TPCA) "importance" weight is greater than 0.6, it is claimed that the spatial (temporal) variability is predominant in the dataset.

The empirical application carried out had as objective to check the prediction and forecast accuracy of the three devised models with a concrete dataset. The three models were applied over the SSTA of the Tropical Pacific Ocean from 1950 - January to 2015 - January. The results of the application lead to conclude that the three models provided an excellent prediction accuracy wherewith any of them can be chosen for prediction studies. For forecasting purposes, the TPCA model showed the best performance; however, no model was able to accurately capture the extreme values of SSTA produced by the onset of El Niño 2015 - 2016. This suggests an evaluation of the devised methodology with the presence of extreme values in the dataset.

6.1 Future research directions

This investigation has revealed a number of interesting results through the simulation study and the empirical application results. Now, next paragraphs give suggestions about future investigations that can be carried out as a result of these discoveries.

1. *To consider possible extensions of the method.* Extensions of the devised methodology can be made in different lines. One of them is to consider the modeling of aerial data irregularly spaced. Indeed, the results found in this thesis have shown that without an explicit specification of the spatial and temporal covariance structure, the models work flexibly and provide good prediction and forecast results for different raster dataset structures. Thus, an study of the modeling in aerial data using the three models would be interesting. Another important extension is to consider modeling raster datasets with categorical variables instead of continuous. To achieve this, slight changes can be made in the internal techniques used for the three models. For example, a categorical PCA (Niitsuma and Okada, 2005) could be used instead of the traditional PCA and forecasting models for categorical data (Agresti, 2013) would be more suitable than the MODWT-AR-NN model used in this thesis.

2. *Doing simulation studies by fine-tuning the internal settings of the model.* The developed models in this work have internally a lot of steps that involve the use of two main tecniques: the PCA and MODWT-AR-NN model. Both tecniques have some internal settings as the optimal number of components to retain for the PCA or the type of mother wavelet and the levels of decomposition to use in the MODWT-AR-NN model. However, this thesis does not prove whether these features influence or not in the forecast and prediction performance of the models. So, a future work proving the influence or not of these internal features in the global forecast accuracy would answer the question.

3. *Doing simulation studies evaluating the performance of the models with different dataset characteristics.* This thesis has performed a simulation study considering three types of scenarios: PSV, PTV and STV processes. Whithin the PSV processes, only isotropic auto-covariance structures were considered. Thus, a simulation study considering anisotropic processes to compare forecast performances would be appealing. For its side, whithin the PTV processes, only AR(1) stationary processes were simulated and other structures were not considered. So, one study considering nonstationary temporal processes could be carried out. For the STV processes, a hierarchical arrangement was developed varying only the spatial structure and making constant the temporal variation with an AR(1) structure. For these simulated

scenarios the interaction between the spatial and temporal component was not controlled sistematically. In that way, another simulation study controling the amount of interaction between the spatial and temporal component would help to expand the scope of the proposed models.

4. *Comparing the proposed methods with other existing spatio-temporal models.* An investigation displaying the differences in prediction and forecast accuracy of the proposed models against other existing statistical spatio-temporal models would make the devised methodology more reliable. Models such as the spatial autoregressive (SAR) model, the conditional autoregressive (CAR) model and its bayesian versions (Gamerman, 2010; Cressie and Wikle, 2015) may be used for this purpose.

5. *The use of more datasets for application purposes.* This thesis exhibited a simple application study over a SSTA dataset in order to verify the accuracy of the proposed models. Other application studies, that involve the spatio - temporal modeling of any raster variable, using the devised methods would enable to check the accuracy of the models in other branches.

Bibliography

Agresti, A. (2013), *Categorical data analysis* John Wiley & Sons.

Anders, U. (1997), *Statistische neuronale Netze* Vahlen.

Ashok, K., Behera, S. K., Rao, S. A., Weng, H., and Yamagata, T. (2007), "El Niño Modoki and its possible teleconnection," *Journal of Geophysical Research: Oceans*, 112(C11).

Banerjee, S., Carlin, B. P., and Gelfand, A. E. (2014), *Hierarchical modeling and analysis for spatial data* Crc Press.

Barnston, A. G., and Ropelewski, C. F. (1992), "Prediction of ENSO episodes using canonical correlation analysis," *Journal of climate*, 5(11), 1316–1345.

Bishop, C. M. (1995), *Neural networks for pattern recognition* Oxford university press.

Bottou, L. (2003), "Stochastic Learning in: Bousquet," *O., Luxburg, Uv and Rätsch, G., editors: Advanced Lectures on Machine Learning Springer, Berlin et al*, p. 146.

Box, G. E., and Jenkins, G. M. (1976), *Time series analysis: forecasting and control, revised ed* Holden-Day.

Cattell, R. B. (1966), "The scree test for the number of factors," *Multivariate behavioral research*, 1(2), 245–276.

Chen, D., Zebiak, S. E., Cane, M. A., and Busalacchi, A. J. (1997), "Initialization and predictability of a coupled ENSO forecast model," *Monthly Weather Review*, 125(5), 773–788.

Cressie, N., and Majure, J. J. (1997), "Spatio-temporal statistical modeling of livestock waste in streams," *Journal of Agricultural, Biological, and Environmental Statistics*, pp. 24–47.

Cressie, N., and Wikle, C. K. (2015), *Statistics for spatio-temporal data* John Wiley & Sons.

Daubechies, I. (1992), *Ten lectures on wavelets* SIAM.

Dutta, S., Ganguli, R., and Samanta, B. (2005), "Investigation of two neural network methods in an automatic mapping exercise," *Journal of Applied GIS*, 1(2), 1–19.

Ferré, L. (1995), "Selection of components in principal component analysis: a comparison of methods," *Computational Statistics & Data Analysis*, 19(6), 669–682.

Frazier, M. W. (2006), *An introduction to wavelets through linear algebra* Springer Science & Business Media.

Gamerman, D. (2010), "Dynamic spatial models including spatial time series," *Handbook of Spatial Statistics*, pp. 437–448.

Granger, C. W., Terasvirta, T. et al. (1993), "Modelling non-linear economic relationships," *OUP Catalogue*, .

Guttorp, P., Meiring, W., and Sampson, P. D. (1994), "A space-time analysis of ground-level ozone data," *Environmetrics*, 5(3), 241–254.

Haar, A. (1910), "Zur theorie der orthogonalen funktionensysteme," *Mathematische Annalen*, 69(3), 331–371.

Haykin, S. S., Haykin, S. S., Haykin, S. S., and Haykin, S. S. (2009), *Neural networks and learning machines*, Vol. 3 Pearson Upper Saddle River, NJ, USA.

Hernández, E., and Weiss, G. (1996), *A first course on wavelets*, 1 edn.

Hornik, K. (1993), "Some new results on neural network approximation," *Neural networks*, 6(8), 1069–1072.

Jackson, D. A. (1995), "Bootstrapped principal components analysis- Reply to Mehlman et al.," *Ecology*, 76(2), 644–645.

Jaffard, S., Meyer, Y., and Ryan, R. D. (2001), *Wavelets: tools for science and technology* SIAM.

Johnson, R. A., Wichern, D. W. et al. (2014), *Applied multivariate statistical analysis*, 4 edn, New Jersey: Prentice-Hall.

Jolliffe, I. T. (2002), *Principal component analysis and factor analysis*, 2 edn, New York: Springer.

Kaiser, H. F. (1960), "The application of electronic computers to factor analysis," *Educational and psychological measurement*, 20(1), 141–151.

Kölzow, D. (1994), "Wavelets; A tutorial and a bibliography," , .

Kug, J.-S., Jin, F.-F., and An, S.-I. (2009), "Two types of El Niño events: cold tongue El Niño and warm pool El Niño," *Journal of Climate*, 22(6), 1499–1515.

Mallat, S. G. (1989), "Multiresolution approximations and wavelet orthonormal bases of $L^2(R)$," *Transactions of the American mathematical society*, 315(1), 69–87.

Mardia, K. V., T., J., and Bibby, J. M. (1980), *Multivariate analysis (probability and mathematical statistics)*, 1 edn, London: Academic Press.

Migon, H. S., Gamerman, D., Lopes, H. F., and Ferreira, M. A. (2005), "Dynamic models," *Handbook of Statistics*, 25, 553–588.

Niitsuma, H., and Okada, T. (2005), Covariance and PCA for categorical variables,, in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, Springer, pp. 523–528.

Pace, R. K., Barry, R., Clapp, J. M., and Rodriquez, M. (1998), "Spatiotemporal autoregressive models of neighborhood effects," *The Journal of Real Estate Finance and Economics*, 17(1), 15–33.

Percival, D. B., and Walden, A. (2000), *Wavelet methods for time series analysis, vol. 4 of Cambridge Series in Statistical and Probabilistic Mathematics*, Cambridge, UK: Cambridge University Press.

Peres-Neto, P. R., Jackson, D. A., and Somers, K. M. (2005), "How many principal components? Stopping rules for determining the number of non-trivial axes revisited," *Computational Statistics & Data Analysis*, 49(4), 974–997.

Pfeifer, P. E., and Jay Deutsch, S. (1980), "Stationarity and invertibility regions for low order starma models: stationarity and invertibility regions," *Communications in Statistics-Simulation and Computation*, 9(5), 551–562.

Preisendorfer, R. W., and Mobley, C. D. (1988), *Principal component analysis in meteorology and oceanography*, 1 edn, Amsterdam: Elsevier.

Qi, M., and Zhang, G. P. (2001), "An investigation of model selection criteria for neural network time series forecasting," *European Journal of Operational Research*, 132(3), 666–680.

Raîche, G., Walls, T. A., Magis, D., Riopel, M., and Blais, J.-G. (2013), "Non-graphical solutions for Cattell's scree test," *Methodology: European Journal of Research Methods for the Behavioral and Social Sciences*, 9(1), 23.

Richman, M. B. (1986), "Rotation of principal components," *International Journal of Climatology*, 6, 293–335.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1985), Learning internal representations by error propagation,, Technical report, California Univ San Diego La Jolla Inst for Cognitive Science.

Sampson, P. D., and Guttorp, P. (1992), "Nonparametric estimation of nonstationary spatial covariance structure," *Journal of the American Statistical Association*, 87(417), 108–119.

Smith, T., Reynolds, R., Peterson, T., and Lawrimore, J. (2008), "Improvements to NOAA's

historical merged land-ocean temperature analysis (1880-2006)," *Journal of Climate*, 21, 2283–2296.

Stoffer, D. S. (1986), "Estimation and identification of space-time ARMAX models in the presence of missing data," *Journal of the American Statistical Association*, 81(395), 762–772.

Ver Hoef, J. M., and Cressie, N. (1993), "Multivariable spatial prediction," *Mathematical Geology*, 25(2), 219–240.

Wang, C., and Weisberg, R. H. (2000), "The 1997–98 El Niño evolution relative to previous El Niño events," *Journal of Climate*, 13(2), 488–501.

White, H. (1992), *Artificial neural networks: approximation and learning theory* Blackwell Publishers, Inc.

Widmann, G. (2000), "Künstliche Neuronale Netze und ihre Beziehung zur Statistik Doctoral Dissertation," *University of Tübingen*, .

Wikle, C. K., Berliner, L. M., and Cressie, N. (1998), "Hierarchical Bayesian space-time models," *Environmental and Ecological Statistics*, 5(2), 117–154.

Wold, H. (1938), A study in the analysis of stationary time series, PhD thesis, Almqvist & Wiksell.

Appendix A

Complementary results

This section shows some complementary results of the simulation study.

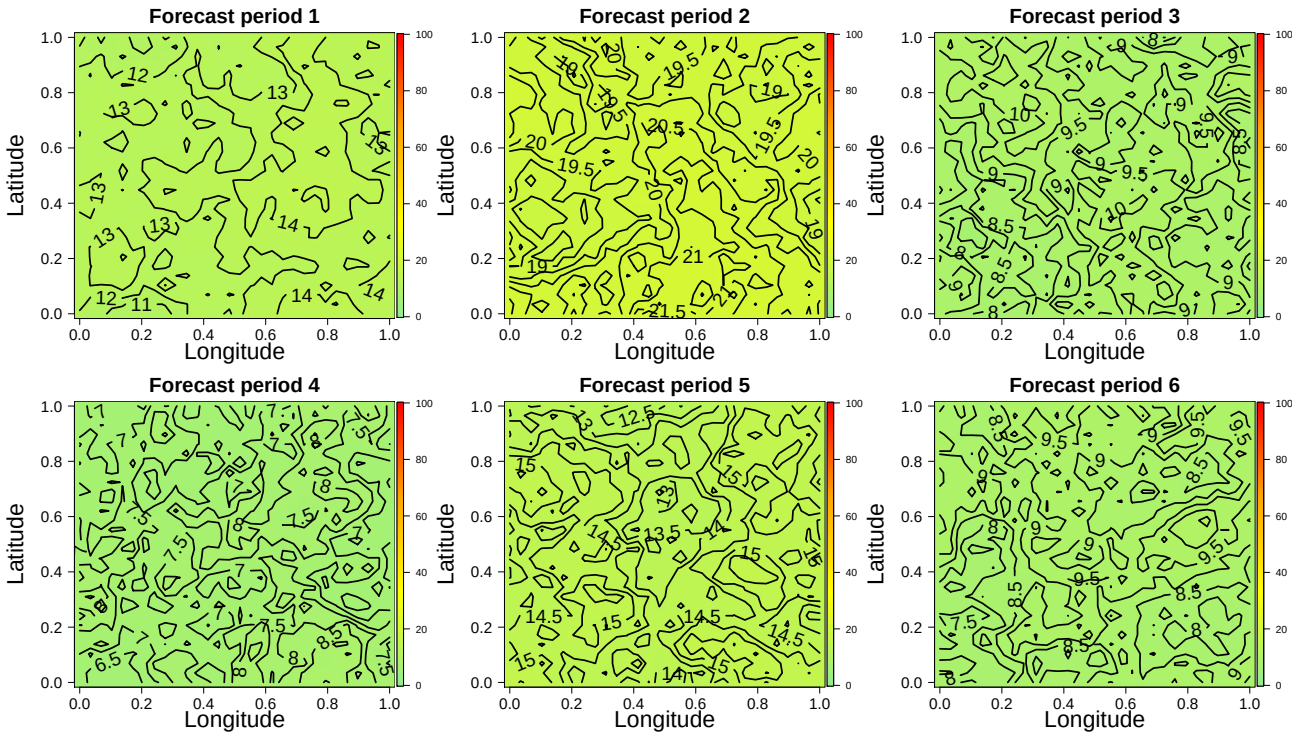

Figure A.1: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the PSV-Exponential process using the SPCA model.

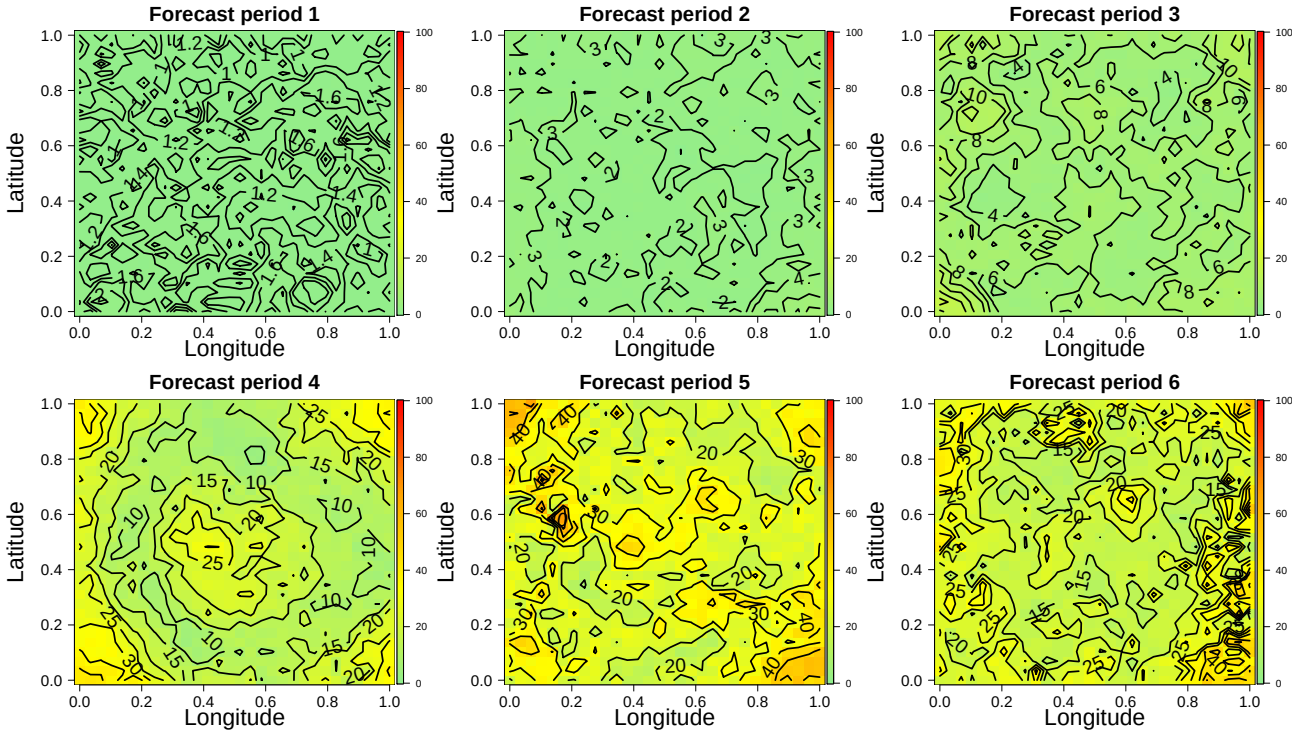

Figure A.2: Spatial distribution of the averaged MAPE (of the 50 replications) corresponding to the six forecasted periods of the PSV-Exponential process using the TPCA model.

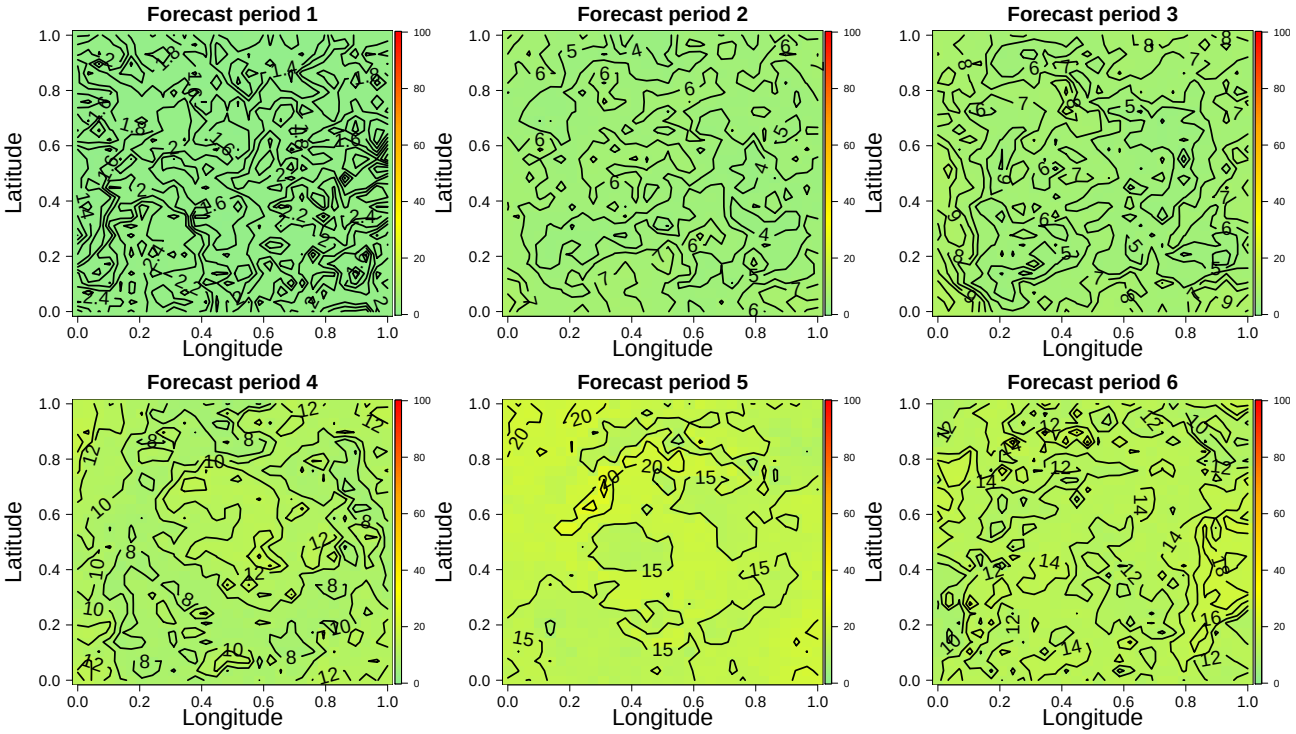

Figure A.3: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the PSV-Exponential process using the STPCA model.

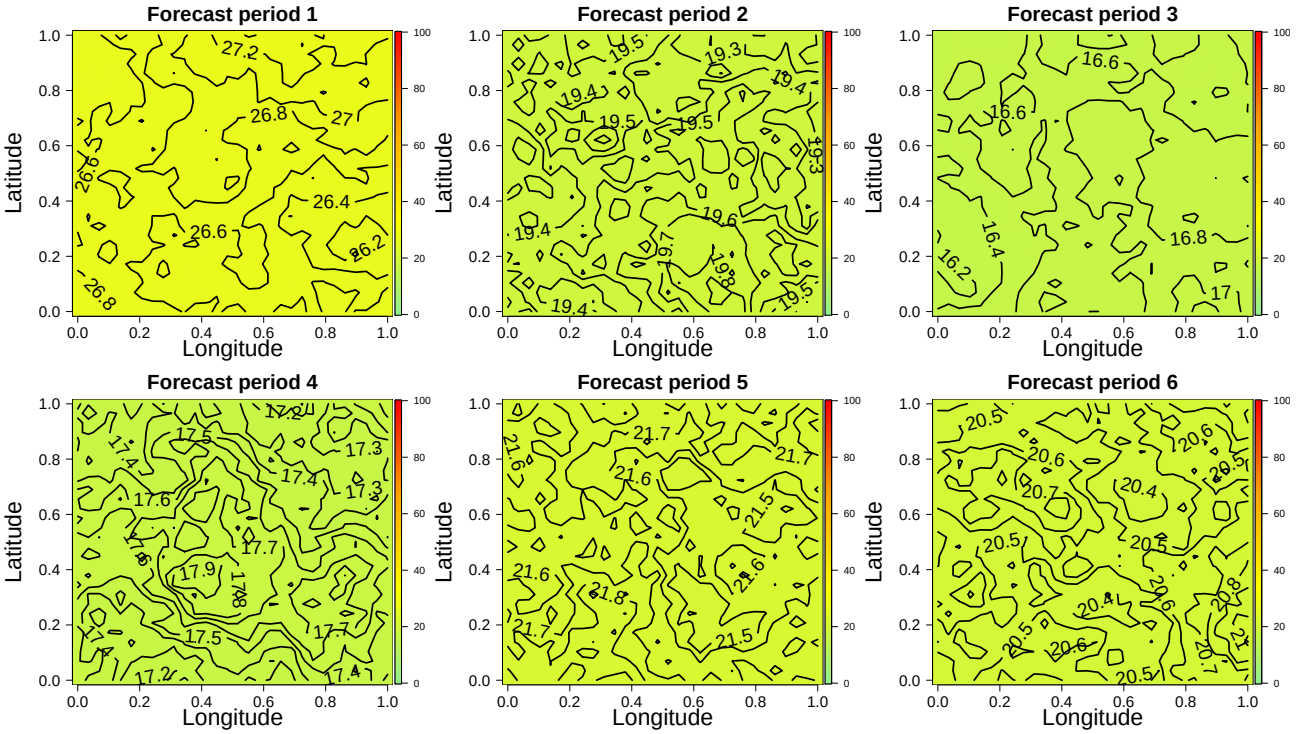

Figure A.4: Spatial distribution of the averaged MAPE (of the 50 replications) corresponding to the six forecasted periods of the PSV-Matérn process using the SPCA model.

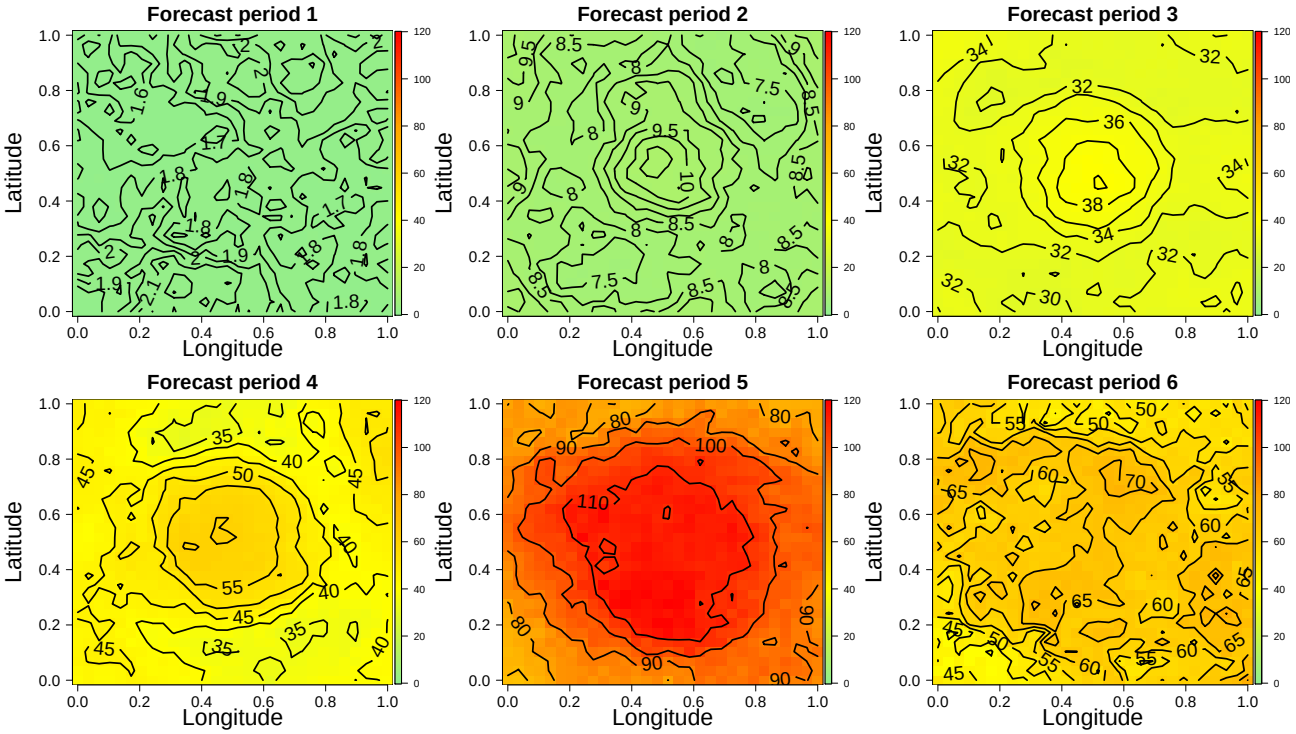

Figure A.5: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the PSV-Matérn process using the TPCA model.

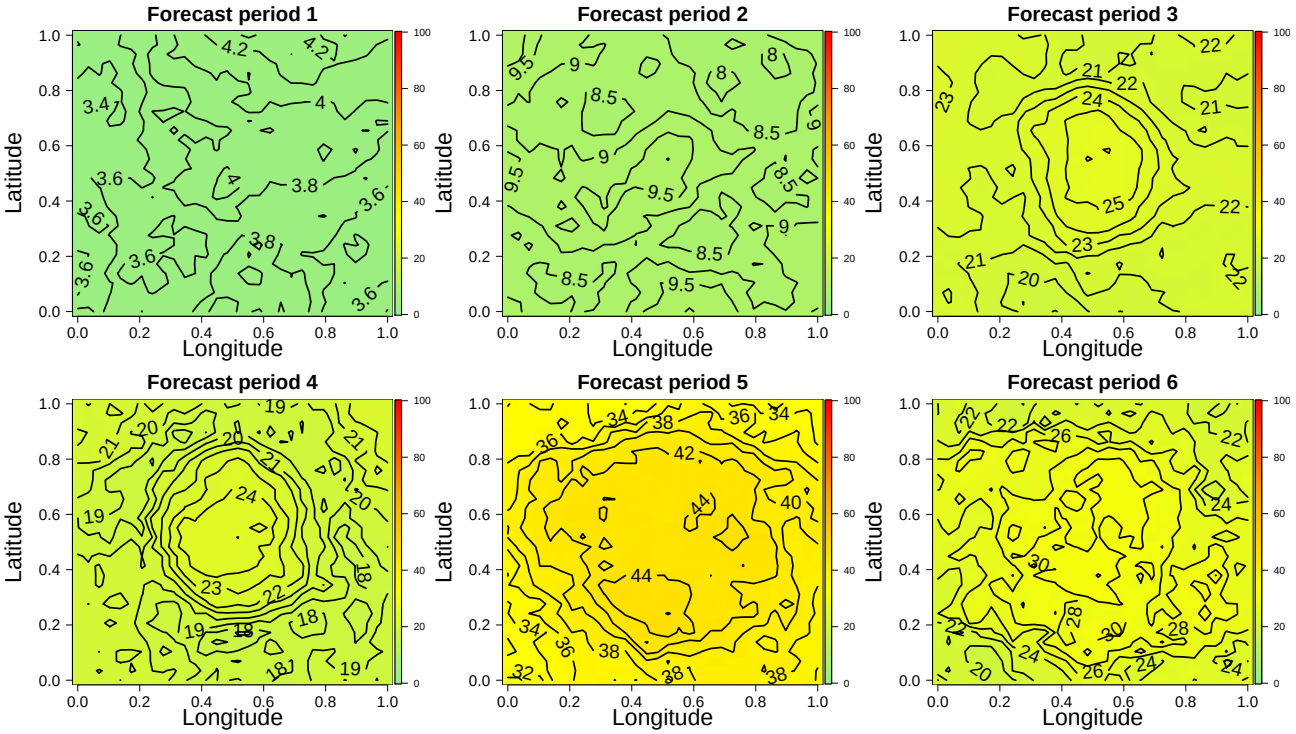

Figure A.6: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the PSV-Matérn process using the STPCA model.

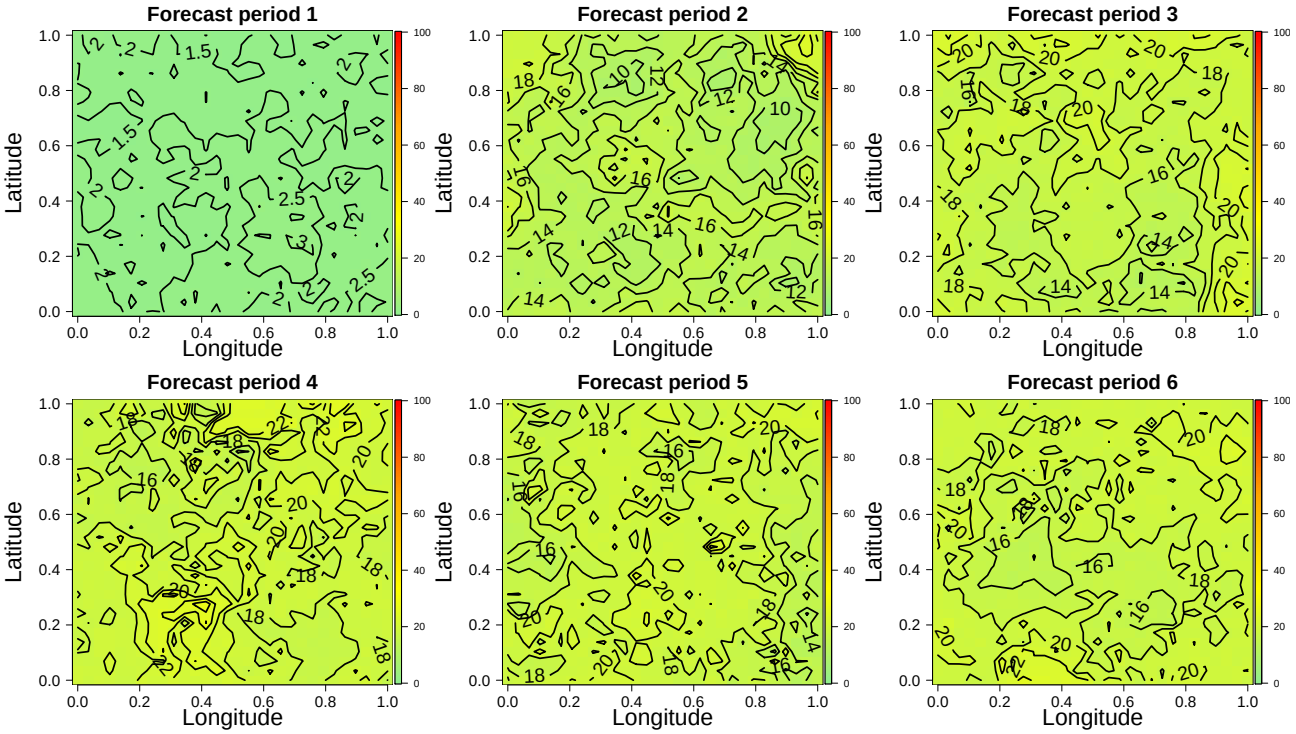

Figure A.7: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the PSV- Spherical process using the STPCA model.

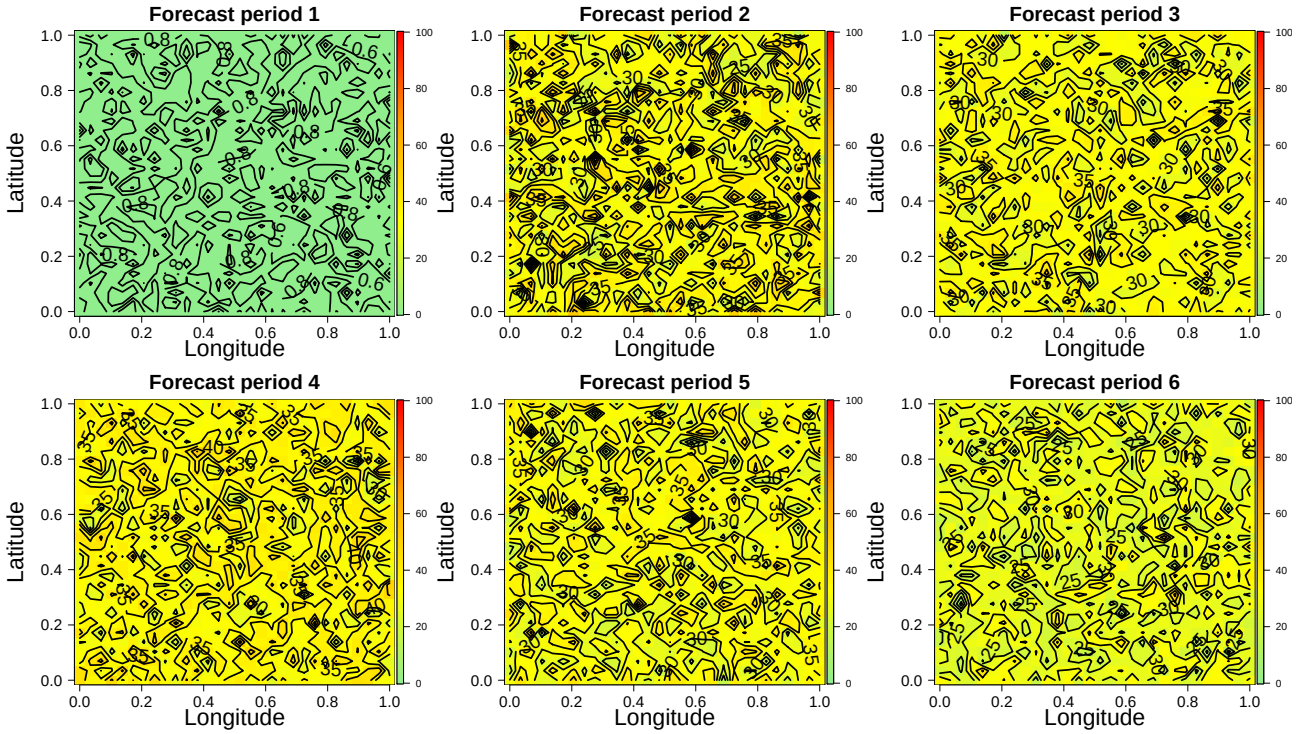

Figure A.8: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the PTV - $\rho = 0.50$ process using the STPCA model.

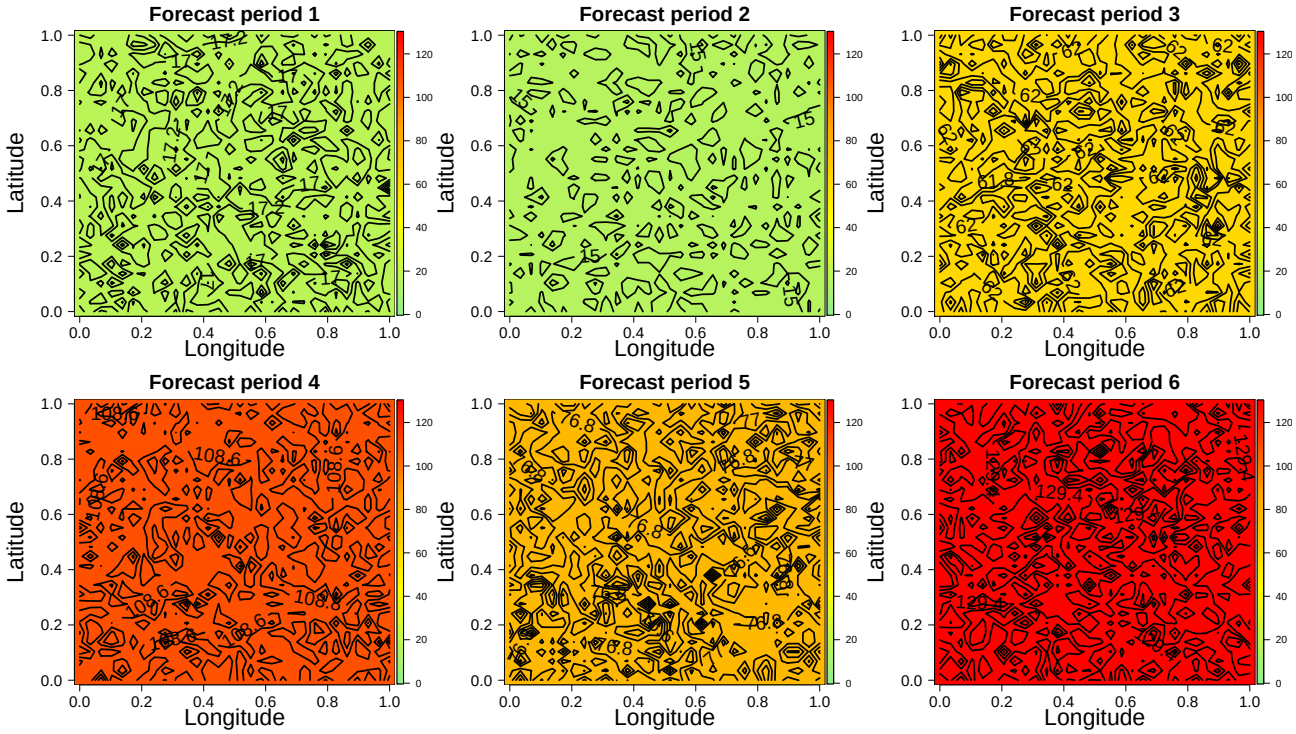

Figure A.9: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the PTV - $\rho = 0.75$ process using the SPCA model.

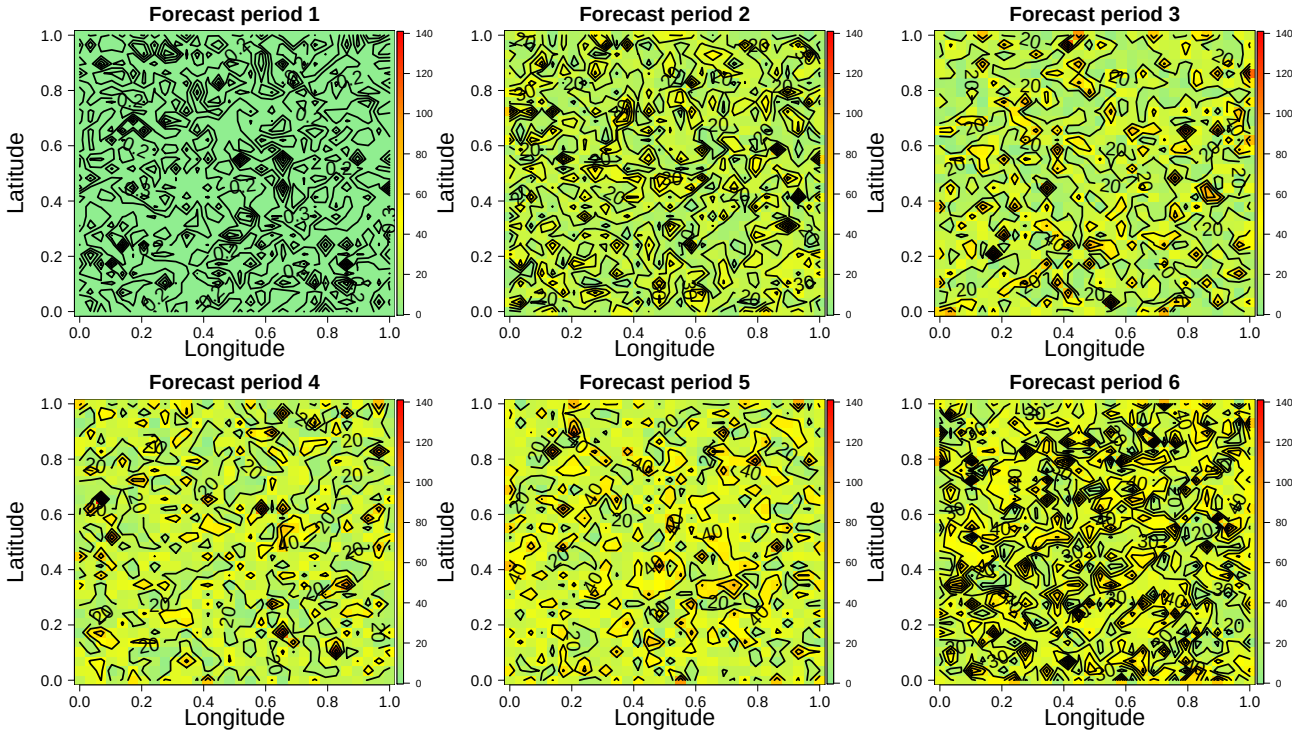

Figure A.10: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the PTV - $\rho = 0.75$ process using the TPCA model.

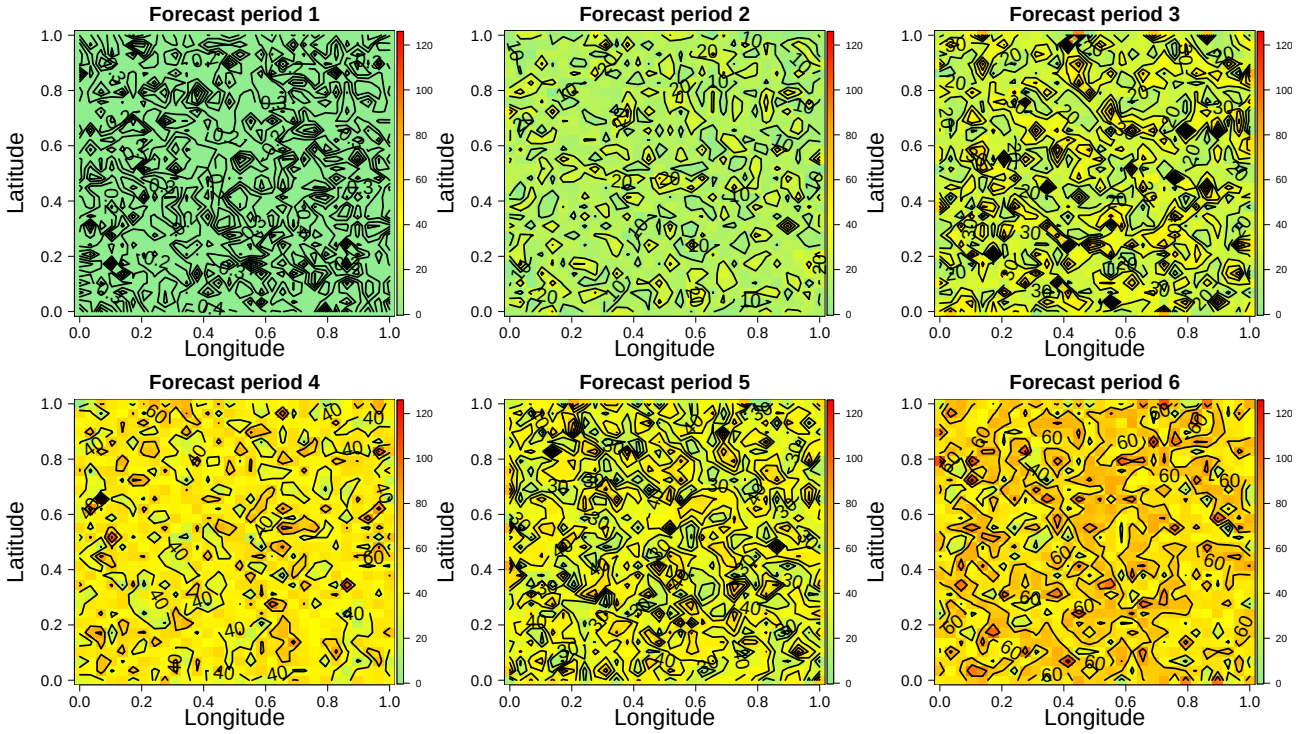

Figure A.11: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the PTV - $\rho = 0.75$ process using the STPCA model.

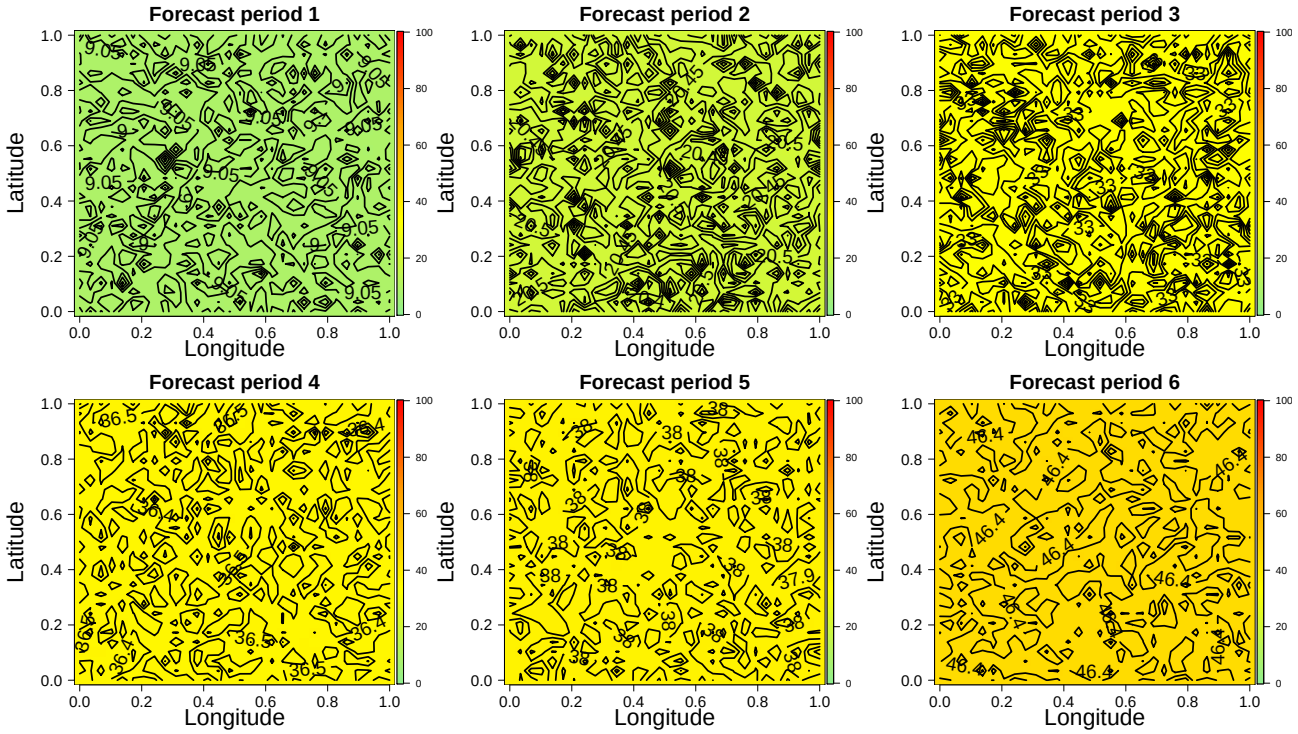

Figure A.12: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the PTV - $\rho = 0.95$ process using the SPCA model.

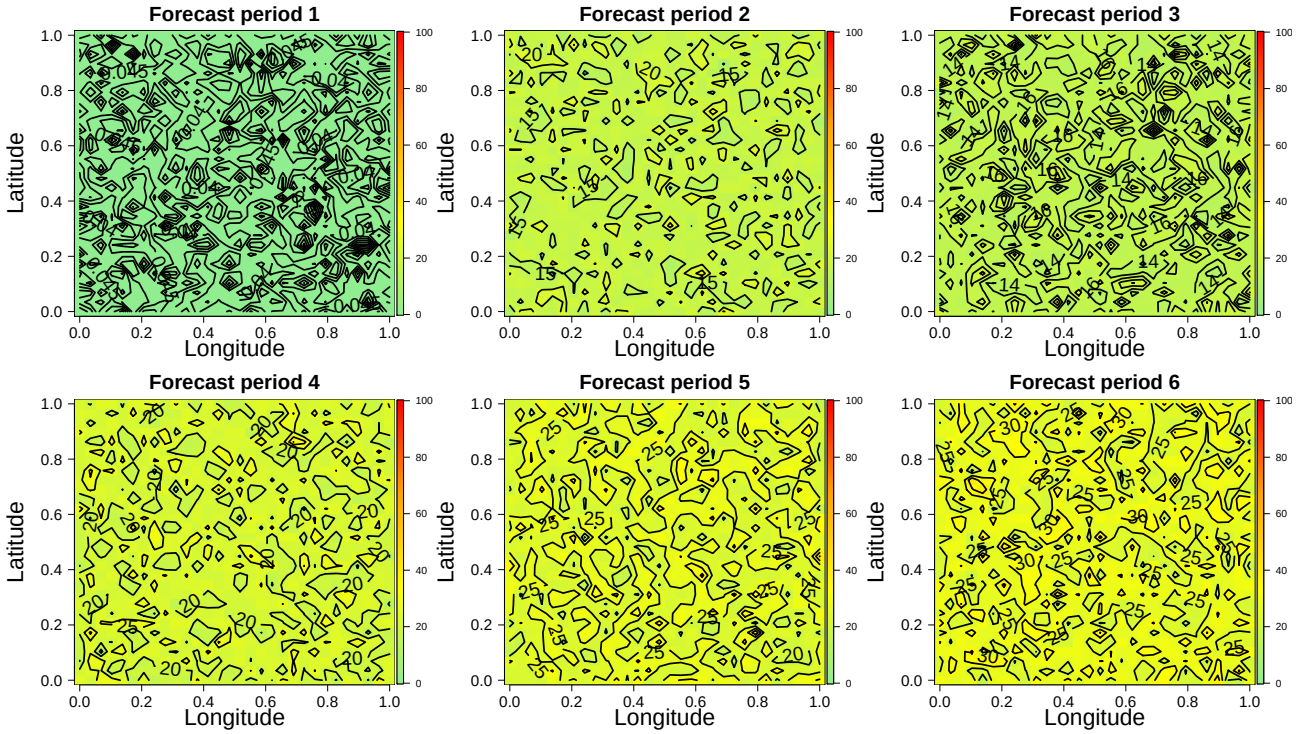

Figure A.13: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the PTV - $\rho = 0.95$ process using the TPCA model.

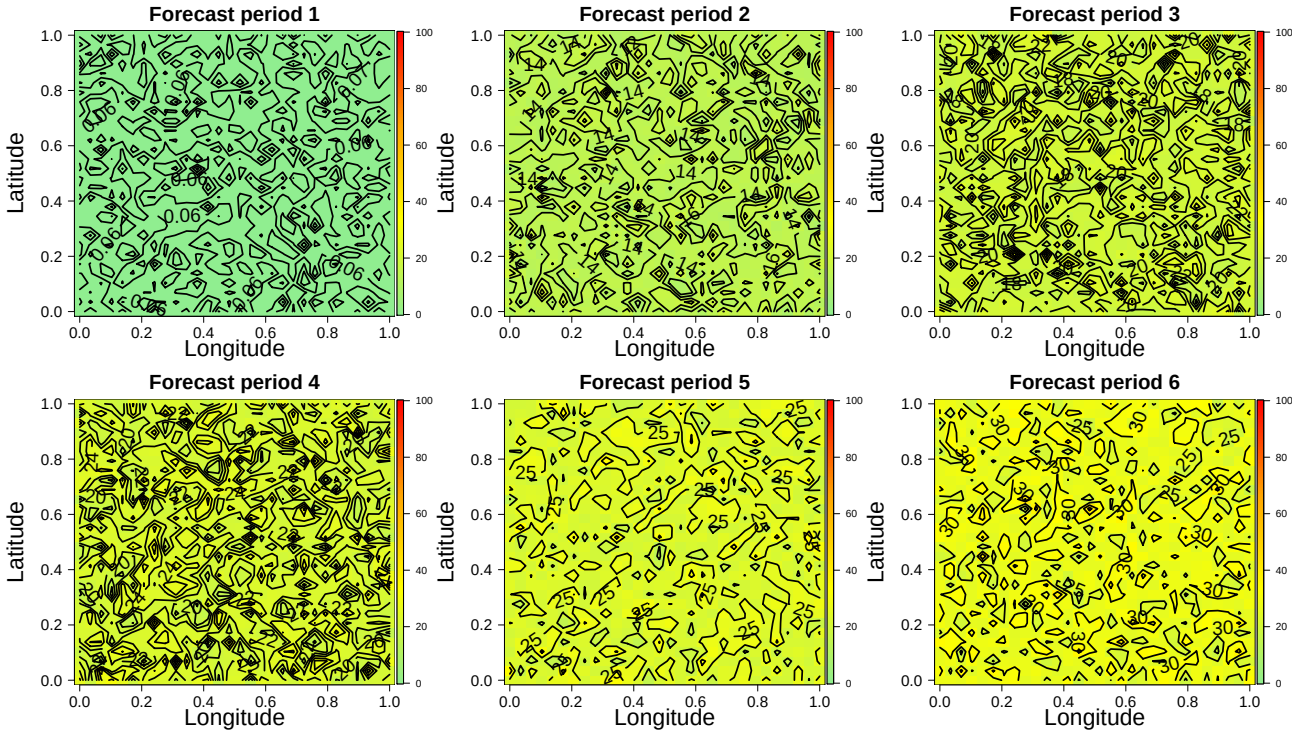

Figure A.14: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the PTV - $\rho = 0.95$ process using the STPCA model.

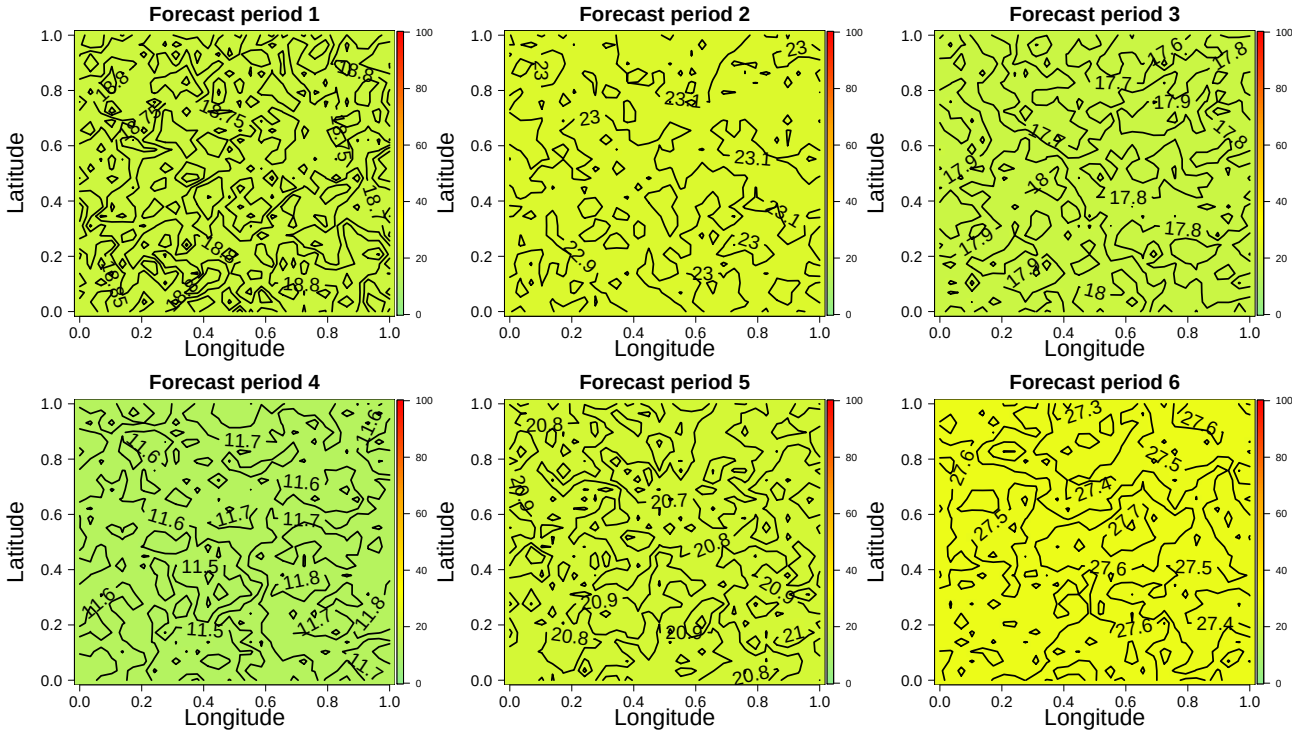

Figure A.15: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the STV - Exponential process using the SPCA model.

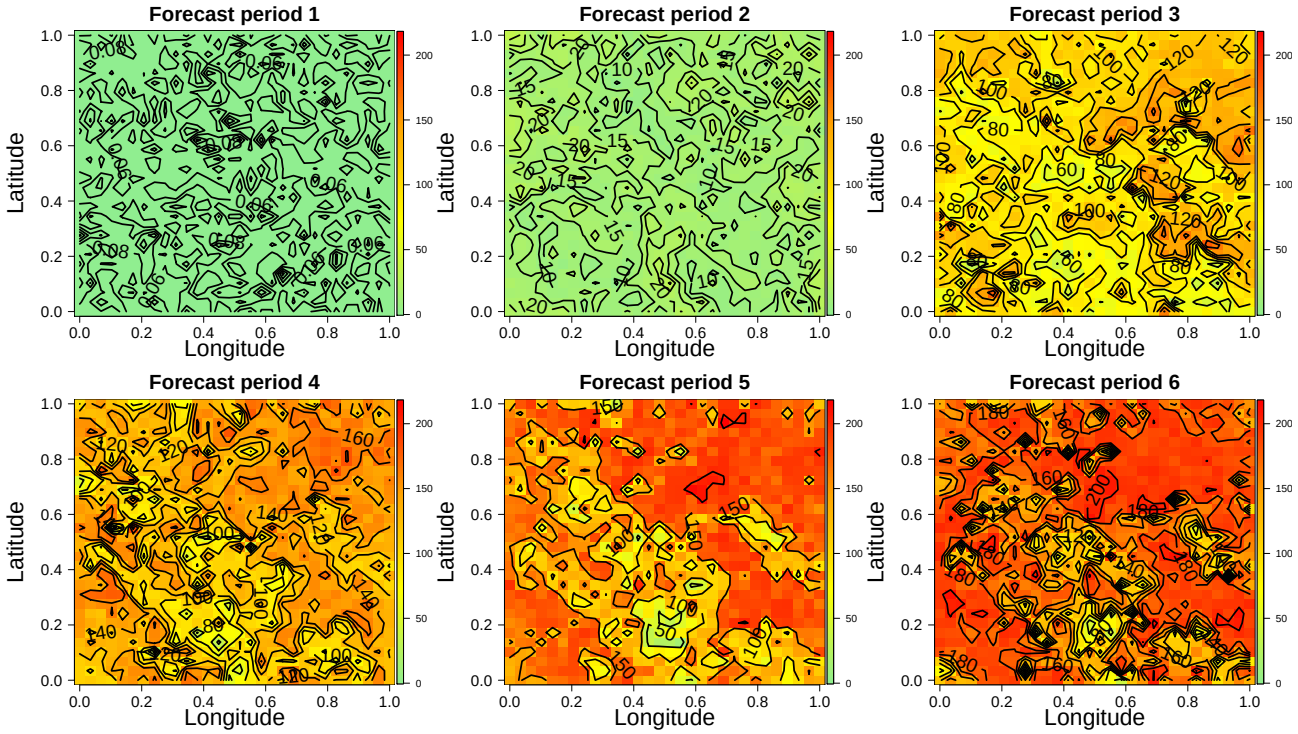

Figure A.16: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the STV - Exponential process using the TPCA model.

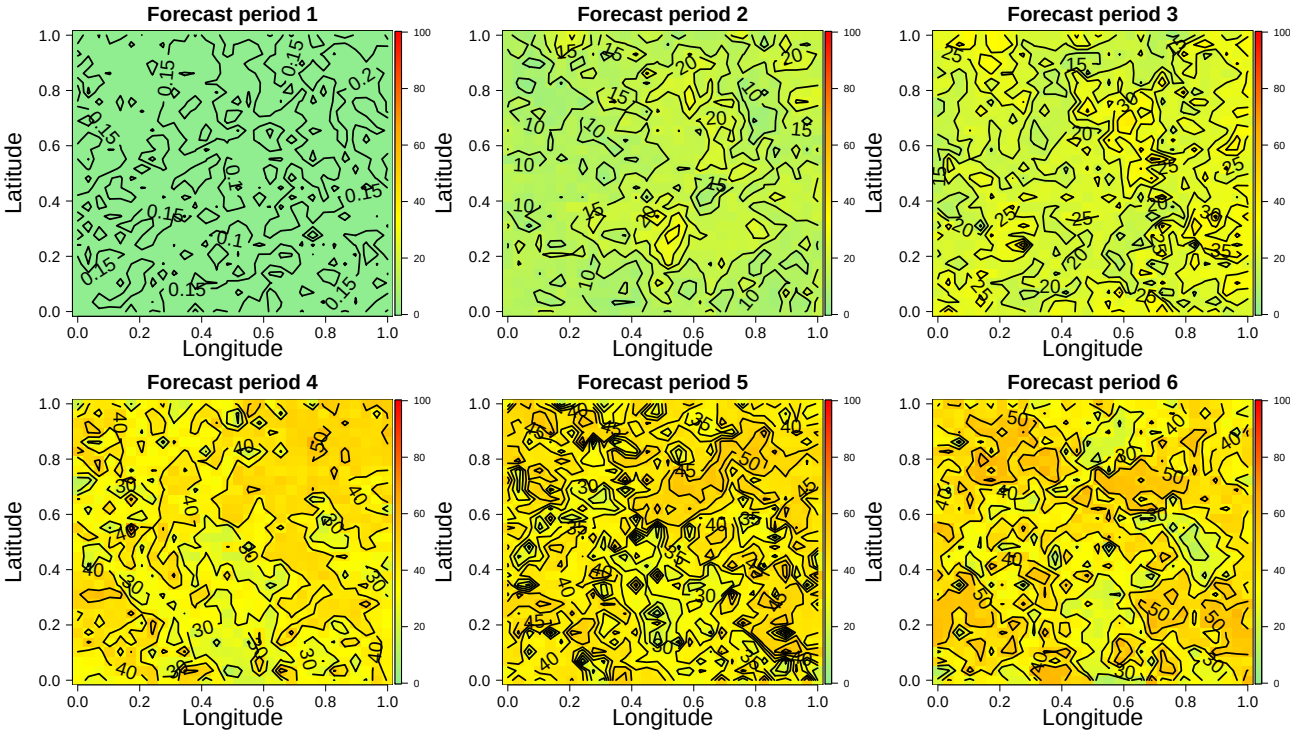

Figure A.17: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the STV - Exponential process using the STPCA model.

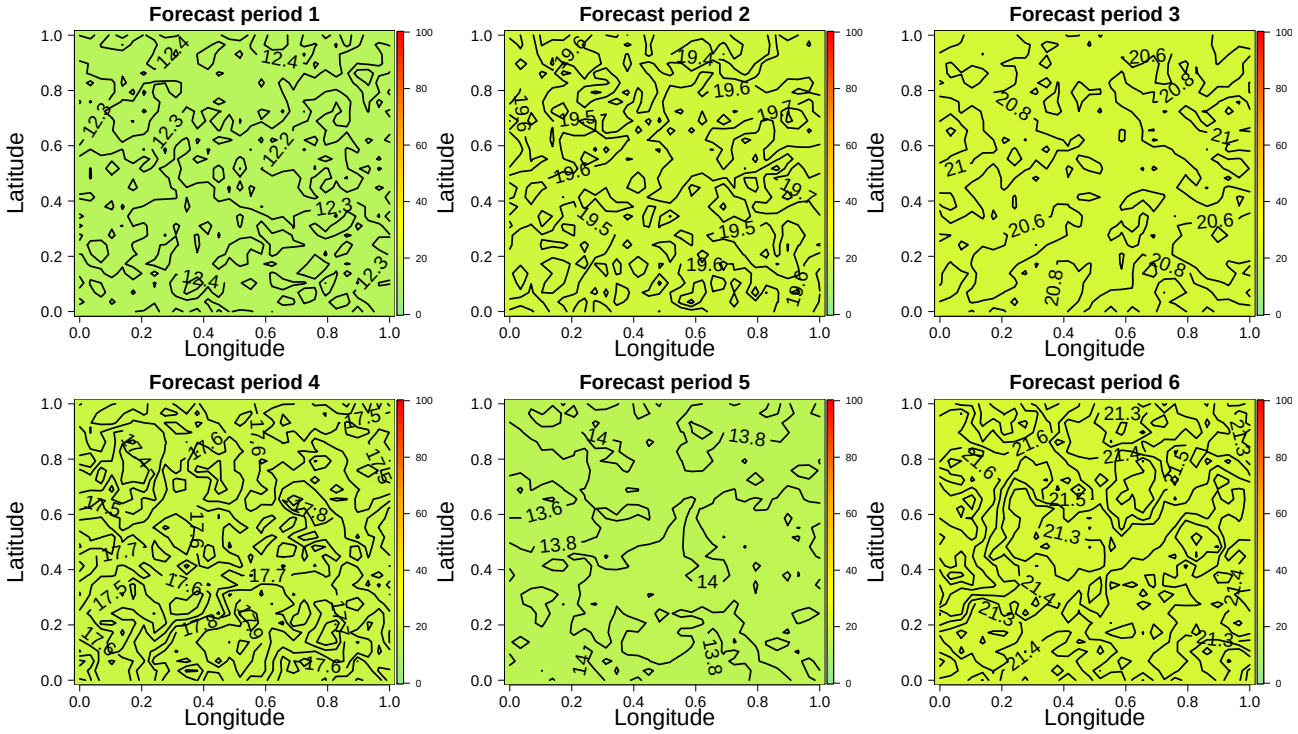

Figure A.18: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the STV - Spherical process using the SPCA model.

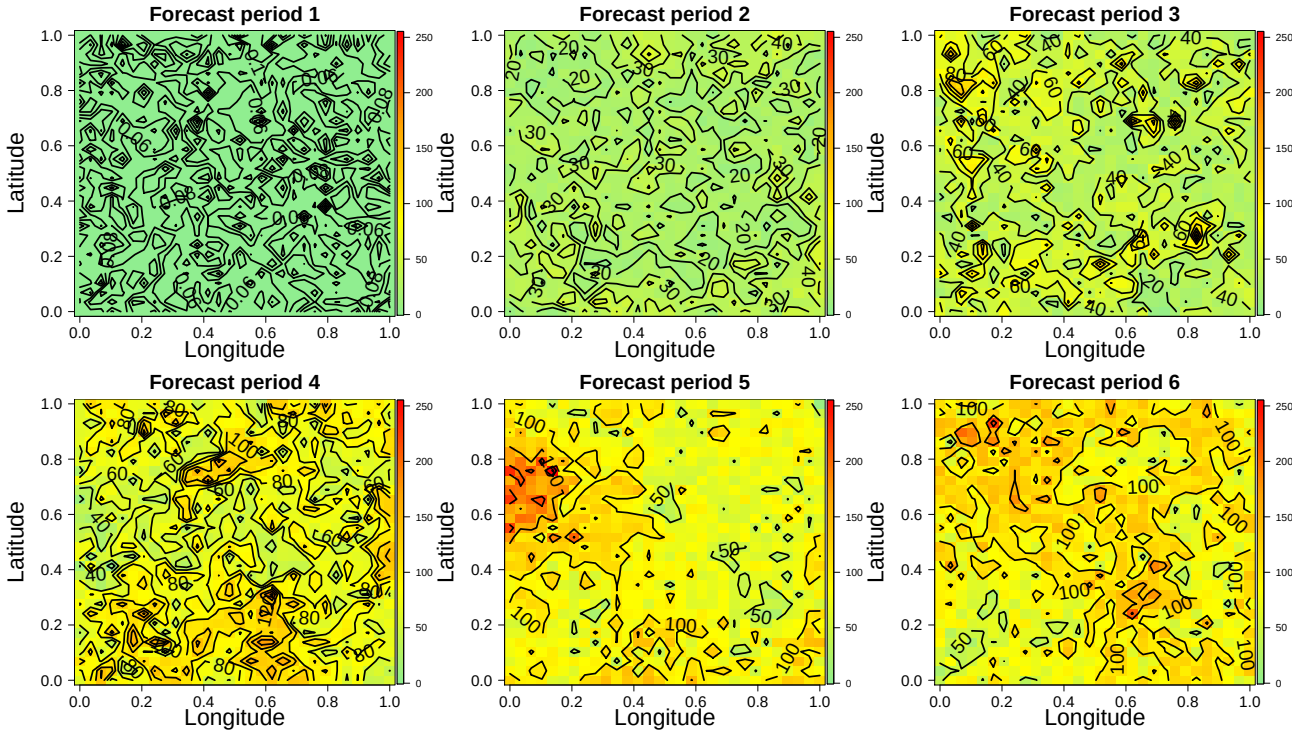

Figure A.19: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the STV - Spherical process using the TPCA model.

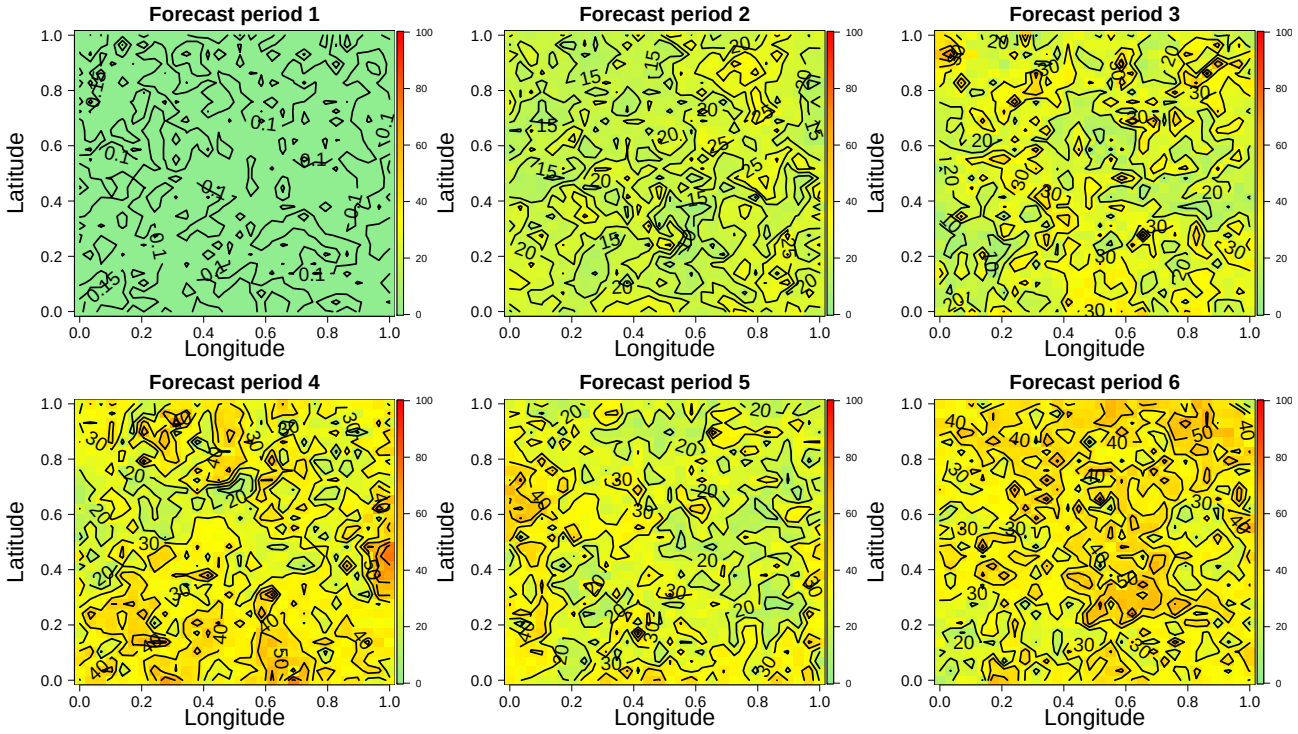

Figure A.20: Spatial distribution of the median MAPE (of the 50 replications) corresponding to the six forecasted periods of the STV - Spherical process using the STPCA model.