Politecnico di Milano
School of Industrial and Information Engineering
M.Sc. in Mathematical Engineering, MST Statistical Learning

Conformal Prediction Bands for Two-Dimensional Functional Time Series

Supervisor:
Simone Vantini

Assistant Supervisors:
Jacopo Diquigiovanni
Matteo Fontana

Master Thesis of:
Niccolò Ajroldi

ID:
928375

Academic Year 2020/2021

Abstract

Conformal Prediction (CP) is a versatile nonparametric technique used to quantify uncertainty in prediction problems. In this work, we propose an extension of such method to the case of time series of functions defined on a bivariate domain. Given the complex nature of data and the non-trivial dependence structure, we adapt the CP procedure, eventually deriving distribution-free prediction bands and providing performance bounds in terms of unconditional coverage and asymptotic exactness. The advantages of the CP method over the traditional Bootstrap approach are explored on synthetic data in a proper Appendix. Moreover, we extend the theory of autoregressive processes in Hilbert space in order to allow for functions with a bivariate domain. Given the novelty of the subject, we present estimation techniques for the Functional Autoregressive model (FAR) and for principal components analysis (PCA) for two-dimensional functional data. An *ad hoc* simulation study is implemented in order to investigate finite sample performances of estimators of the functional autoregressive model, comparing them with benchmark forecasting methods. Finally, we explore benefits and limits of the proposed approach on a real case study, employing a dataset from Copernicus Climate Change Service, collecting daily observations of Sea Level Anomalies of the Black Sea in the last twenty years.

Keywords: Conformal Prediction; Functional Autoregressive Model; Forecasting; Functional Time Series; Prediction band; Two-dimensional Functional data.

Sommario

La Conformal Prediction (CP) è una versatile tecnica nonparametrica utilizzata per la quantificazione dell'incertezza nei problemi di predizione. In questo lavoro, proporremo un estensione di tale metodo al caso di serie temporali di funzioni definite su un dominio bivariato. Per via dell'alta complessità dei dati e dell'intrinseca dipendenza temporale, adatteremo la classica procedura CP, derivando bande di previsione e quantificando l'efficienza predittiva in termini di copertura incondizionata ed esattezza asintotica. I vantaggi del metodo CP rispetto alla tradizionale tecnica Bootstrap verranno esplorati su dati sintetici in un'apposita appendice. Adatteremo inoltre la teoria dei processi autoregressivi in spazi di Hilbert a funzioni con un dominio bivariato. Data l'innovatiità dell'argomento, proporremo tecniche di stima per il modello Autoregressivo Funzionale (FAR) e due diversi metodi per l'analisi delle componenti principali (PCA) per dati funzionali bidimensionali. Attraverso l'implementazione di uno studio di simulazione, valuteremo le prestazioni a campione finito degli stimatori del modello autoregressivo funzionale, confrontandoli con metodi di previsione di riferimento. Infine, approfondiremo i benefici e i limiti dell'approccio proposto su un reale caso di studio, utilizzando un set di dati del Copernicus Climate Change Service, il quale raccoglie osservazioni giornaliere delle anomalie del livello del mare del Mar Nero negli ultimi venti anni.

Parole chiave: Bande di predizione; Conformal Prediction; Dati funzionali bidimensionali; Modello funzionale autoregressivo; Predizione; Serie temporali funzionali.

Contents

List of Figures

List of Tables

1 Introduction

Functional data arise naturally across several disciplines, motivating an increasing demand for dedicated analysis techniques. One interesting and expanding subfield of functional data analysis (FDA) regards functional time series (FTS). Informally, a functional time series consists of an ordered sequence of functional objects recorded over a time period and characterized by some sort of temporal dependency.

Uncertainty quantification in the context of FTS forecasting has received increasing attention in the statistical community in recent decades. Among the different publications tackling the problem of distributional forecasting, Hyndman and Ullah (2007) proposed a forecasting approach based on a preliminary functional principal component analysis (FPCA) and a subsequent univariate time series modelling of each of the FPC scores. Additionally, relying on the assumption of Gaussian errors, the authors proposed confidence bands for the predicted function. Building up on the just mentioned work, Hyndman and Shang (2009) enriched the forecasting algorithm with decreasing weights, introducing also functional partial least square regression along with a bootstrap technique to obtain prediction bands. Zhu and Politis (2017) presented a residual-based bootstrap procedure to construct prediction regions, exploiting a functional version of a kernel estimator for the autoregressive operator. Rossini and Canale (2018) proposed an autoregressive functional modelling framework for dependent functional data with curve constraints, relying on a bootstrap approach in order to quantify uncertainty around the forecasted curve. Recently, Hernández et al. (2021) also tackled the problem of constructing simultaneous predictive confidence bands for a stationary functional time series, introducing an entropy measure for stochastic processes and deriving themselves prediction sets through a functional bootstrap procedure.

As is now evident, the bootstrap approach is by far the most developed technique for distributional forecasting in the FTS context. However, the aforementioned procedure is only asymptotically valid, doesn't provide any theoretical guarantee, and is usually very computationally intensive, especially in the infinite-dimensional context of functional data. In this work, we will instead focus on Conformal Prediction, another nonparametric approach which has proved itself to be very useful and versatile. The first appearance of such technique dates back in Gammerman et al. (1998) and it has been later presented in great details in the book of Vovk et al. (2005) and in Balasubramanian et al. (2006). A recent review of the theory of Conformal inference can be found in Zeni et al. (2020). The attractiveness of Conformal Prediction relies on its great generality and versatility, which permits to couple it with potentially any machine learning technique, in order to obtain prediction sets.

It is important to notice that the theory of CP is developed under the only assumption of *exchangeable* data. Such very mild hypothesis, despite being one of the strengthens of CP, is clearly not suitable for the time series context, in which one have to deal with temporal dependence between data. Adapting CP beyond exchangeable data has recently gathered attention in the statistical community. Tibshirani et al. (2020) introduced a weighted version of CP for problems in which the test and training covariate distributions differ, but the likelihood ratio between the two distributions is known. A different approach is carried on by Chernozhukov et al. (2018) who rephrased the CP framework in the context of *randomization inference*, proving approximate validity of the resulting prediction sets under weak assumptions on the conformity scores and on the stationarity of the time series. An interesting recent publication by Xu and Xie (2021) develops a method to build distribution-free prediction intervals for bootstrap ensemble estimators in the context of time series, by combining the work of Chernozhukov with the *jackknife+-after-Bootstrap* (Kim et al. 2020).

Applications of Conformal Prediction to the functional setting is also a novel research field. Whereas Lei et al. (2013) applied CP to compute simultaneous confidence bands as a data exploration routine, Diquigiovanni et al. (2021c) proposed a new family of nonconformity measures which permits finding prediction sets in closed form both for univariate and multivariate functional data (Diquigiovanni et al. 2021b) and later also in the presence of temporal dependence (Diquigiovanni et al. 2021a).

Even though functional data are typically considered to be defined on a univariate domain over the real line and a great research effort has been settled to develop specific analysis techniques, to the best of our knowledge very few publications deal with functional data on bivariate domain. Such type of data are commonly refereed to as *two-dimensional functional data*, in order to distinguish them with *bivariate functional data*, which are rather functions with a bivariate image, instead of a bivariate domain. In this work, we will focus on time series of surfaces, representing them as two-dimensional functional data with temporal dependence. We will build on the already cited research of Diquigiovanni et al. (2021a), extending it to two-dimensional functional data and adapting it to allow for different point predictors.

The remainder of this paper is as follows: we first introduce two-dimensional functional data in Section 2, providing some formal definitions. In Section 3 we illustrate conformal prediction for functional time series. In Section 4 different point-prediction algorithms are presented, adapting them to the Conformal inference setting and consequently comparing them by means of the resulting prediction bands in a simulation study in Section 5. Finally,

in Section 6 we employ the developed techniques to obtain distributional forecasts in a real scenario, predicting the surface level of the Black Sea surface and repeating the procedure day by day.

2 Two-dimensional functional data

Two-dimensional functional data are very common in various scientific fields, since they can arise from records over a raster. Such type of data are frequent in Earth observation missions, as for instance in temperature tracking of specific areas (Zhou and Pan 2014), in NASA's Tropospheric Emission Spectrometer measurements of ozone atmospheric concentrations or also in sea level recordings (Huang et al. 2017). All this type of data share the fact that they are defined over a bivariate domain, and can thus be modeled as surfaces, hence as functional objects. Functional data analysis (FDA) is also accruing interest as a novel option for representing images, since it allows to preserve their continuous nature. P.-Muñoz et al. (2014) give a detailed description of the entire imaging process using the FDA approach, proposing also a representation of iris images through functional data. As shown by Gervini (2010) even mortality rates can be interpreted as two-dimensional functional data, where one dimension is the temporal one ad the other one refers to age.

A common preliminary step for functional data analysis consists in a smoothing procedure, where we aim to create an approximation of the original function, while filtering out noise and measurements error (we refer to Ramsay and Silverman 2005 for an exhaustive discussion on smoothing techniques for one dimensional functional data). A novel regularization technique for Gaussian random fields on a rectangular domain has been proposed by Rakêt (2010), where a roughness measure is introduced as a penalizing term in the likelihood function and a Bayesian model is employed to obtain estimates. Another bivariate smoothing approach in a penalized regression framework is introduced by Ivanescu and Andrada (2013), allowing for the estimation of multiple functional parameters of completely or incompletely sampled two-dimensional functional data. Yan et al. (2018) proposed spatio-temporal smooth sparse decomposition, a novel methodology which serves as a dimensionality reduction and denoising technique in a process monitoring framework for images streams. Functional Principal Component Analysis (FPCA) has also been extended to data on bivariate domains. Zhou and Pan (2014) presented and compared two approaches for performing FPCA on functions on a non-rectangular domain, one based on a singular value decomposition of discretized data and another one which makes use of a mixed effect model.

We proceed now by introducing some formal definitions for two-dimensional functional time series.

2.1 Definitions

A two-dimensional functional time series is an ordered sequence $Y_1, \dots, Y_T$ of random variables with values in a Hilbert space $\mathbb{H}$, equipped with the Borel σ-algebra $\mathcal{B}(\mathbb{H})$. More formally, we consider a probability space $(\Omega, \mathcal{F}, \mathbb{P})$, and define a random function at time t as $Y_t : \Omega \to \mathbb{H}$, measurable with respect to $\mathcal{B}(\mathbb{H})$.

In the rest of the article we will consider functions belonging to $\mathbb{H} = \mathcal{L}^2([c,d] \times [e,f])$, the space of measurable square integrable real-valued functions defined on the rectangle $[c,d] \times [e,f] \subset \mathbb{R}^2$, with $c, d, e, f \in \mathbb{R}$, $c < d$, $e < f$. Such choice is motivated by many reasons, one above all is the fact that, by considering functions in $\mathcal{L}^2([c,d] \times [e,f])$, the usual Frechet mean for functional data coincides with the pointwise mean and the covariance kernel coincides with the point-wise covariance. Moreover, $\mathbb{H}$ is a separable Hilbert space, with the usual inner product:

$$\langle x, y \rangle := \int_c^d \int_e^f x(u,v) y(u,v) du dv \qquad \forall x, y \in \mathbb{H} \tag{1}$$

Hereafter, we will always assume $\{Y_t\}_{t=1}^T \subset \mathcal{L}^4(\Omega, \mathcal{F}, \mathbb{P})$ and consider only stationary time series. We can thus easily define the Frechet mean as the unique element μ of $\mathbb{H}$ that solves $\operatorname{argmin}_{x \in \mathbb{H}} \mathbb{E}[||Y - x||^2]$. As mentioned before, one can easily prove that $\mathbb{E}[Y_t(u,v)] = \mu(u,v)$ $\forall (u,v) \in [c,d] \times [e,f]$. Throughout this work, we will consider only centered random fields, in such a way that μ coincides with the zero function. The covariance operator $\Gamma_0 : \mathbb{H} \to \mathbb{H}$ for a zero-mean stochastic process $\{Y_t\}_{t=1}^T$ can thus be defined as:

$$\Gamma_0 x = \mathbb{E}[\langle Y_t, x \rangle Y_t] \qquad \forall x \in \mathbb{H} \tag{2}$$

Similarly, we introduce the lag-1 autocovariance operator Γ_1:

$$\Gamma_1 x = \mathbb{E}[\langle Y_t, x \rangle Y_{t+1}] \qquad \forall x \in \mathbb{H} \tag{3}$$

Notice that, since functions are elements of $\mathbb{H}$ and are assumed to have finite second moments, the covariance operator can equivalently be defined by introducing the autocovariance function $\gamma_0(u,v;w,z)$:

$$\gamma_0(u,v;w,z) := \operatorname{Cov}[Y_t(u,v), Y_t(w,z)] \tag{4}$$

in such a way that Γ_0 can be seen as a kernel operator:

$$(\Gamma_0 x)(u,v) = \int_c^d \int_e^f \gamma_0(u,v;w,z)x(w,z)dwdz \tag{5}$$

The functional mean μ can be estimated by the sample mean function:

$$\hat{\mu}(u,v) = \frac{1}{T}\sum_{t=1}^{T} Y_t(u,v) \tag{6}$$

and the covariance function γ_0 by its sample counterpart $\hat{\gamma}_0$:

$$\hat{\gamma}_0(u,v;w,z) = \frac{1}{T}\sum_{t=1}^{T} Y_t(u,v)Y_t(w,z) \tag{7}$$

The sample covariance operator $\hat{\Gamma}_0$ can hence be defined as the corresponding kernel operator:

$$(\hat{\Gamma}_0 x)(u,v) = \int_c^d \int_e^f \hat{\gamma}_0(u,v;w,z)x(w,z)dwdz = \frac{1}{T}\sum_{t=1}^{T}\langle Y_t, x\rangle Y_t(u,v) \tag{8}$$

where, as usual, $x \in \mathbb{H}$ and $(u,v) \in [c,d] \times [e,f]$, in such a way that:

$$\hat{\Gamma}_0 x = \frac{1}{T}\sum_{t=1}^{T}\langle Y_t, x\rangle Y_t \tag{9}$$

Similarly, we can estimate the lag-1 autocovariance operator Γ_1:

$$\hat{\Gamma}_1 x = \frac{1}{T-1}\sum_{t=1}^{T-1}\langle Y_t, x\rangle Y_{t+1} \tag{10}$$

Under rather general weak dependence assumptions these estimators are $\sqrt{n}$-consistent. One may, for example, adopt the concept of $\mathcal{L}^p$-m-approximability introduced in Hörmann and Kokoszka (2010) to prove that $\mathbb{E}[||\hat{\mu} - \mu||_{\mathbb{H}}^2] = \mathcal{O}(n^{-1})$ and $\mathbb{E}[||\hat{\Gamma_0} - \Gamma_0||^2] = \mathcal{O}(n^{-1})$, where $||.||$ is the classical operatorial norm: $||F|| := \sup_{||x||_{\mathbb{H}}=1} ||Fx||_{\mathbb{H}}$ for any linear bounded operator $F : \mathbb{H} \to \mathbb{H}$.

3 Conformal inference for functional time series

Conformal Prediction (CP) is a nonparametric approach to the problem of uncertainty quantification in forecasting. While it can be used to obtain prediction sets for a new observation in a very generic setting, we will employ it in a regression framework. In a nutshell, the CP approach is based on the idea of assigning scores to new candidate points in order to assess their non-conformity. Prediction sets are then derived by inverting the hypothesis test obtained using such scores, including only points with a relatively high conformity level.

In this work, we will consider only the Inductive Conformal Prediction or Split Conformal Prediction method (Papadopoulos et al. 2002). Such modification of the original Transductive Conformal method is not only computationally efficient, but is also necessary in the functional framework. Indeed, the main drawback of the Full Conformal approach is that the prediction algorithm needs to be retrained for every possible candidate test point y. In practice, in multivariate problems, where y lies in $\mathbb{R}^p$, one runs the above routine for candidate y over a p-dimensional grid. While such approach is prohibitive for high-dimensional spaces, since computational times grow exponentially with p, it is actually unfeasible in a functional setting, in which y lies in an infinite-dimensional space. On the other hand, employing Split Conformal inference along with a particular nonconformity score, introduced by Diquigiovanni et al. (2021c) and explicitly tailored for functional data, permits deriving prediction sets in closed form. Specifically, we will consider the approach introduced by Chernozhukov et al. (2018) for time-dependent data and later adapted by Diquigiovanni et al. (2021a) to allow for functional time series in a Split Conformal setting. We will extend such method to two-dimensional functional data.

Consider a time series $\boldsymbol{Z}_1, \dots, \boldsymbol{Z}_T$ of regression pairs $\boldsymbol{Z}_t = (\boldsymbol{X}_t, Y_t)$, with $t = 1, \dots, T$. Let Y_t be a random variable with values in $\mathbb{H}$ (namely a *random function*), while $\boldsymbol{X}_t$ is a set of covariates at time t belonging to a measurable space. Notice that $\boldsymbol{X}_t$ is a generic set of regressors, which may contain both exogenous and endogenous variables. In particular, later in the manuscript, we will consider $\boldsymbol{X}_t$ to contain only the lagged version of the function Y_t, namely Y_{t-1}.

We aim to design a procedure that outputs a prediction set $\mathcal{C}_{T,1-\alpha}$ for Y_{T+1} based on $\boldsymbol{Z}_1, \dots, \boldsymbol{Z}_T$ and $\boldsymbol{X}_{T+1}$, with unconditional coverage greater or equal than $1-\alpha$ for any significance level α. More formally, we define $\mathcal{C}_{T,1-\alpha}$ to be a *valid* prediction region if:

$$\mathbb{P}(Y_{T+1} \in \mathcal{C}_{T,1-\alpha}) \geq 1 - \alpha \tag{11}$$

Moreover, we would like to construct a particular type of prediction sets, commonly known as *prediction bands*, formally defined as:

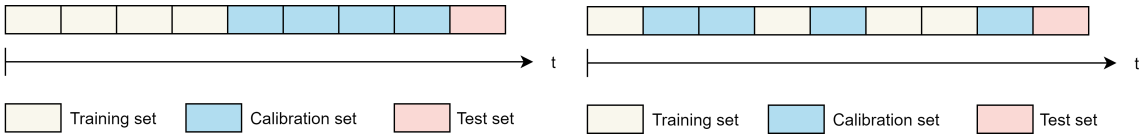
$$\{y \in \mathbb{H} : y(u,v) \in B_n(u,v) \quad \forall (u,v) \in [c,d] \times [e,f]\} \tag{12}$$

with $B_n(u,v) \subseteq \mathbb{R}$ union of finitely many intervals for each $(u,v) \in [c,d] \times [e,f]$. The convenience of such type of prediction sets is extensively motivated in literature (see e.g. Pintado and Romo 2009, Lei et al. 2013 and Diquigiovanni et al. 2021c), since a prediction set of this type can be easily visualized (at least for one-dimensional functional data) in a plot, a property that is instead not guaranteed if the prediction region is a generic subset of $\mathbb{H}$.

Let $\boldsymbol{z}_1, \dots, \boldsymbol{z}_T$ be realizations of $\boldsymbol{Z}_1, \dots, \boldsymbol{Z}_T$. As the name suggests, Split Conformal inference is based on a random split of data into two disjoint sets: let $\mathcal{I}_1$, $\mathcal{I}_2$ be a random partition of $\{1, \dots, T\}$, such that $|\mathcal{I}_1| = m$, $|\mathcal{I}_2| = l$, $m, l \in \mathbb{N}$ $m, l > 0$, $m + l = T$. Historical observations $\boldsymbol{z}_1, \dots, \boldsymbol{z}_T$ are divided into a *training set* $\{\boldsymbol{z}_h,\, h \in \mathcal{I}_1\}$, from which we will estimate the prediction model, and a *calibration set* $\{\boldsymbol{z}_h,\, h \in \mathcal{I}_2\}$, that will be used in an out-of-sample context to measure the nonconformity of a new candidate function. The choice of the split ratio is non-trivial and has motivated discussion in the statistical community. Including more data in the training set improves the estimation of the point predictor $\hat{g}_{\mathcal{I}_1}$. At the same time, having fewer data in the calibration set produces a very rough p-value function, resulting in greater actual coverage with respect to the nominal one. This trade-off problem is enhanced in the time series context, in which one would like to have both training and calibration sets as large as possible, since asymptotic validity is guaranteed when both l and m goes to infinity. Throughout this work, the training-calibration ratio is fixed equal to 50%-50%, as commonly suggested in literature. Moreover, we stress the fact that the split is random. This clearly introduce variability in the procedure since results depend on the particular division of data. We acknowledge a recent advancement in this direction, namely Multi Split Conformal Prediction (Solari and Djordjilović 2021), which aggregates single split CP intervals across multiple splits.

Another interesting question regarding the type of split comes up in the time series context. Due to lacking of exchangeability, two different subdivisions are possible in this framework. A first choice could consist in a sequential division of data, where the split point is no longer random, but is a result of the training-calibration proportion (see Figure 1a). Wisniewski et al. (2020) applied this scheme in a rolling window fashion to forecast Market Makers' Net Positions. While this choice may seem more consistent in the presence of

temporal dependence, since it does not split subsequent observations in two different sets, it may lead to very biased results if the training size m is very small or if data present a different trend or seasonal component in the training and calibration sets. The interested reader may refer to Kath and Ziel (2019) for a more comprehensive discussion on this topic. All in all, in order to make the model more robust, we preferred to split data randomly, as reported in Figure 1b.



(a) Consecutive split (b) Random split

Figure 1: Possible types of split with $T = 8$, $m = l = 4$.

We then introduce a *nonconformity measure* $\mathcal{A}$, which is a measurable function with values in $\mathbb{R} \cup \{+\infty\}$. The role of $\mathcal{A}(\{\boldsymbol{z}_h,\, h \in \mathcal{I}_1\}, \boldsymbol{z})$ is to quantify the nonconformity of a new datum $\boldsymbol{z}$ with respect to the training set $\{\boldsymbol{z}_h,\, h \in \mathcal{I}_1\}$. The choice of the nonconformity measure has a fundamental impact on the resulting prediction sets. Indeed, whereas the validity is guaranteed regardless of the specific nonconformity measure employed (Diquigiovanni et al. 2021c), such choice is crucial if we aim to find prediction bands (12) and if we want to find them in closed form. Motivated by such considerations, we will employ the following nonconformity score, introduced by Diquigiovanni et al. (2021c):

$$\mathcal{A}(\{\boldsymbol{z}_h : h \in \mathcal{I}_1\}, \boldsymbol{z}) = \operatorname*{ess\,sup}_{(u,v)\in[c,d]\times[e,f]} \frac{|y(u,v) - g_{\mathcal{I}_1}(u,v;\boldsymbol{x}_{T+1})|}{s_{\mathcal{I}_1}(u,v)} \tag{13}$$

where $\boldsymbol{z} = (\boldsymbol{x}_{T+1}, y)$, $g_{\mathcal{I}_1}$ is a point predictor estimated from the training set $\mathcal{I}_1$, depending also on $\boldsymbol{x}_{T+1}$ and $s_{\mathcal{I}_1}$ is a *modulation function*, which is a positive function depending on the training set itself that allows for prediction bands with non-constant width. Notice that, on a theoretical point of view, the nonconformity measure $\mathcal{A}$ may assume infinite values, since we are embedding functions in $\mathcal{L}^2([c,d] \times [e,f])$ and we have thus no guarantee on their boundness. To overcome this issue, one can instead consider the functional space $\mathcal{L}^\infty([c,d] \times [e,f])$, as done by Diquigiovanni et al. (2021c), however, such space equipped with the usual $\mathcal{L}^2$ scalar product, is not closed, and is therefore not a Hilbert space. For such reason, we decided to settle anyway the analysis in $\mathcal{L}^2([c,d] \times [e,f])$, resorting to the fact that, in practical applications, (13) will only assume finite values, given the finite nature

of observed data.

The estimation of $g_{\mathcal{I}_1}$ is discussed in Section 4, while the functional standard deviation will be employed as modulation function $s_{\mathcal{I}_1}$, allowing for wider bands in the parts of the domain where data show high variability and narrower and more informative prediction bands in those parts characterized by low variability. For an extensive discussion on the optimal choice of modulation function, we refer to Diquigiovanni et al. (2021c).

We now aim to define a family Π of index permutations $\pi_i : \{1, \dots, T+1\} \to \{1, \dots, T+1\}$, that leave unchanged the indices of the training set, and modify only $\{\mathcal{I}_2, T+1\}$, namely the indices of the calibration set and the next time step.

In order to do so, let's first introduce a function $\lambda : \{\mathcal{I}_2, T+1\} \to \{1, \dots, l+1\}$ such that $\lambda(t)$ returns the t-th element of the ordered set $\{\mathcal{I}_2, T+1\}$. Fix now a positive integer $b \in \{1, \dots l+1\}$ such that $\frac{l+1}{b} \in \mathbb{N}$ and define a family $\tilde{\Pi}$ of index permutations that act on the set $\{1, \dots, l+1\}$. Each $\tilde{\pi}_i \in \tilde{\Pi}$ is required to be a bijection $\tilde{\pi}_i : \{1, \dots, l+1\} \to \{1, \dots, l+1\}$, for $i = 1, \dots, \frac{l+1}{b}$. In particular, we will consider non-overlapping blocking permutations, with b representing the size of the blocking scheme:

$$\tilde{\pi}_i(j) = \begin{cases} j + (i-1)b & \text{if } 1 \leq j \leq l - (i-1)b + 1 \\ j + (i-1)b - l - 1 & \text{if } l - (i-1)b + 2 \leq j \leq l + 1 \end{cases} \tag{14}$$

Notice that $|\tilde{\Pi}| = \frac{l+1}{b}$. Moreover, such family of transformations forms an algebraic group, containing among other the identity transformation $\tilde{\pi}_1$.

It is then straightforward to introduce the family Π of index permutations acting on $\{1, \dots, T+1\}$. Each $\pi_i \in \Pi$, with $i = 1, \dots, \frac{l+1}{b}$ is defined as:

$$\pi_i(t) = \begin{cases} t & \text{if } t \in \mathcal{I}_1 \\ \lambda^{-1}(\tilde{\pi}_i(\lambda(t))) & \text{if } t \in \mathcal{I}_2 \cup \{T+1\} \end{cases} \tag{15}$$

In Figure 2 is reported a trivial example of the families of permutation Π and $\tilde{\Pi}$.

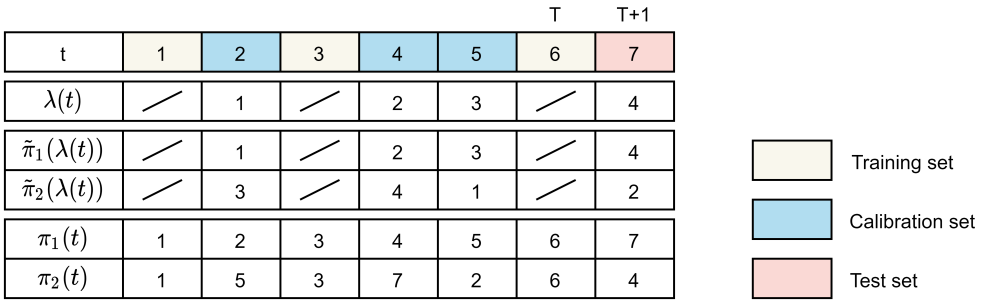


Figure 2: Example of permutation families $\tilde{\Pi}$ and Π, with sample size $T = 6$, training set $\mathcal{I}_1 = \{1, 3, 6\}$, calibration set $\mathcal{I}_2 = \{2, 4, 5\}$, $l = m = 3$, size of blocking scheme $b = 2$. In this case $\lambda : \{\mathcal{I}_2, T + 1\} \equiv \{2, 4, 5, 7\} \to \{1, 2, 3, 4\}$, $\tilde{\Pi} = \{\tilde{\pi}_1, \tilde{\pi}_2\}$ and $\Pi = \{\pi_1, \pi_2\}$.

Consider now a candidate function $y \in \mathbb{H}$ and define the augmented dataset as $\boldsymbol{Z}_{(y)} = \{\boldsymbol{Z}_t\}_{t=1}^{T+1}$, where:

$$\boldsymbol{Z}_t = \begin{cases} (\boldsymbol{X}_t, Y_t), & \text{if } 1 \leq t \leq T \\ (\boldsymbol{X}_{T+1}, y), & \text{if } t = T + 1 \end{cases} \tag{16}$$

We refer to $\boldsymbol{Z}^{\pi}_{(y)} = \{\boldsymbol{Z}_{\pi(t)}\}_{t=1}^{T+1}$ as the randomized version of $\{\boldsymbol{Z}_t\}_{t=1}^{T+1}$. Let's then define the randomization p-value as:

$$p(y) = \frac{1}{|\Pi|} \sum_{\pi \in \Pi} \mathbf{1}(S(\boldsymbol{Z}^{\pi}_{(y)}) \geq S(\boldsymbol{Z}_{(y)})) \tag{17}$$

where the nonconformity scores $S(\boldsymbol{Z}_{(y)})$ and $S(\boldsymbol{Z}^{\pi}_{(y)})$ are defined as:

$$S(\boldsymbol{Z}_{(y)}) = \mathcal{A}(\{\boldsymbol{Z}_h \colon h \in \mathcal{I}_1\}, \boldsymbol{Z}_{T+1}) \tag{18}$$

$$S(\boldsymbol{Z}^{\pi}_{(y)}) = \mathcal{A}(\{\boldsymbol{Z}_h \colon h \in \mathcal{I}_1\}, \boldsymbol{Z}_{\pi(T+1)}) \tag{19}$$

The idea is to apply permutations, modifying the order of observations in the calibration set, while at the same time preserving the dependence between them, thanks to the block structure of Π. For each π, we compute the nonconformity score of $\boldsymbol{Z}^{\pi}_{(y)}$. The p-value of a test candidate value y is then determined as the proportion of randomized versions $\boldsymbol{Z}^{\pi}_{(y)}$ with a higher or equal nonconformity score than the one of the original augmented dataset $\boldsymbol{Z}_{(y)}$. Notice that $p(y)$ is a measure of the *conformity* of the candidate function y with respect to the permutation family Π. It is then natural to include in the prediction set only functions y

with an "high" conformity level. In other words, given a significance level $\alpha \in [b/(l+1), 1]$[1], we define the prediction bands by test inversion:

$$\mathcal{C}_{T,1-\alpha} := \{y \in \mathbb{H}\colon\ p(y) > \alpha\} \tag{20}$$

As mentioned before, the advantage of using the Split Conformal method along with the conformity measure (13) relies on the possibility to find the prediction set in closed form. Define k^s to be the $\lceil(|\Pi|+1)(1-\alpha)\rceil$th smallest value of the set $\{S(\boldsymbol{Z}_{\pi(t)}), \pi \in \Pi\}$.

$$\begin{aligned}
y \in \mathcal{C}_{T,1-\alpha} &\iff p(y) > \alpha \\
&\iff S(\boldsymbol{Z}_{(y)}) \leq k^s \\
&\iff \operatorname*{ess\,sup}_{(u,v)\in[c,d]\times[e,f]} \frac{|y(u,v) - g_{\mathcal{I}_1}(u,v;\boldsymbol{x}_{T+1})|}{s_{\mathcal{I}_1}(u,v)} \leq k^s \\
&\iff |y(u,v) - g_{\mathcal{I}_1}(u,v;\boldsymbol{x}_{T+1})| \leq k^s s_{\mathcal{I}_1}(u,v) \qquad \forall (u,v) \in [c,d]\times[e,f] \\
&\iff y(u,v) \in [g_{\mathcal{I}_1}(u,v;\boldsymbol{x}_{T+1}) \pm k^s s_{\mathcal{I}_1}(u,v)] \qquad \forall (u,v) \in [c,d]\times[e,f]
\end{aligned}$$

Therefore, we have derived the prediction set in closed form:

$$\mathcal{C}_{T,1-\alpha} := \{y \in \mathbb{H} : y(u,v) \in [g_{\mathcal{I}_1}(u,v;\boldsymbol{x}_{T+1}) \pm k^s s_{\mathcal{I}_1}(u,v)] \quad \forall (u,v) \in [c,d]\times[e,f]\} \tag{21}$$

In the case in which regression pairs are exchangeable, the proposed method retains exact, model-free validity (Chernozhukov et al. 2018). However, when such assumption is not met, one can guarantee only approximately validity of the proposed approach under weak assumptions on the conformity score and the stationarity of the time series.
More formally, let $\mathcal{A}^*$ be an oracle nonconformity measure, inducing oracle nonconformity score S^*. Define F to be the cumulative (unconditional) distribution function of the oracle nonconformity scores, namely $F(x) = \mathbb{P}(S^*(\boldsymbol{Z}^{\pi}_{(y)}) < x)$ and $\hat{F}$ the empirical counterpart, obtained by applying permutations $\pi \in \Pi$: $\hat{F}(x) := \frac{1}{|\Pi|}\sum_{\pi\in\Pi} \mathbb{1}\{S^*(\boldsymbol{Z}^{\pi}_{(y)}) < x\}$. Let $\{\delta_{1\bar{l}}, \delta_{2\bar{m}}, \gamma_{1\bar{l}}, \gamma_{2\bar{m}}\}$ be sequences of numbers converging to zero. Theorem 1 of Diquigiovanni et al. (2021a) prescribes sufficient conditions in order to guarantee asymptotic exactness of the prediction set. We report such result with a slightly modified notation. Let here $\boldsymbol{Z} = \boldsymbol{Z}_{(Y_{T+1})}$, where the candidate function y is now substituted by the random function Y_{T+1}.

[1] if $\alpha \in (0, b/(l+1))$ the resulting prediction set will coincide with the entire space $\mathbb{H}$

Theorem 1. *If the following conditions hold:*

- $\sup_{a\in\mathbb{R}} |\hat{F}(a) - F(a)| \leq \delta_{1\bar{l}}$ *with probability* $1 - \gamma_{1\bar{l}}$
- $\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\left[S(\boldsymbol{Z}^{\pi}) - S^{*}(\boldsymbol{Z}^{\pi}_{(y)})\right]^2 \leq \delta^2_{2\bar{m}}$ *with probability* $1 - \gamma_{2\bar{m}}$
- $|S(\boldsymbol{Z}^{\pi}) - S^{*}(\boldsymbol{Z}^{\pi})| \leq \delta_{2\bar{m}}$ *with probability* $1 - \gamma_{2\bar{m}}$
- *With probability* $1 - \gamma_{2\bar{m}}$ *the pdf of* $S^{*}(\boldsymbol{Z}^{\pi})$ *is bounded above by a constant* D

then the Conformal confidence set has approximate coverage α*:*

$$|\mathbb{P}(Y_{T+1} \in \mathcal{C}_{T,1-\alpha}(\boldsymbol{X}_{T+1}) - (1-\alpha))| \leq 6\delta_{1\bar{l}} + 2\delta_{2\bar{m}} + 2D\left(\delta_{2\bar{m}} + 2\sqrt{\delta_{2\bar{m}}}\right) + \gamma_{1\bar{l}} + \gamma_{2\bar{m}} \quad (22)$$

The first condition concerns the approximate ergodicity of $\hat{F}$ for F, a condition which holds for strongly mixing time series using blocking permutation Π defined in (15) (Chernozhukov et al. 2018). The second condition is a requirement for the quality of approximating the oracle $S^{*}(Z^{\pi})$ with $S(Z^{\pi})$, intuitively, $\delta^2_{2\bar{m}}$ bounds the discrepancy between the nonconformity scores and their oracle counterparts.

4 Point Prediction

In order to guarantee validity of Conformal Prediction bands, the choice of an accurate point predictor is crucial. As mentioned before, whereas in the typical i.i.d. case finite-sample unconditional coverage still holds when the model is heavily misspecified, in the time series context a strong model misspecification compromises the coverage guarantees and not only the efficiency of the resulting prediction bands. The interested reader may further investigate this issue in the work of Chernozhukov et al. (2018) and Diquigiovanni et al. (2021a).

4.1 FAR(1)

One of the most popular statistical models used to capture temporal dependence between functional observations is the functional autoregressive process (FAR). The theory of functional autoregressive processes in Hilbert spaces is developed in the monograph of Bosq (2000) and a comprehensive collection of statistical advancements for the FAR model can be found in the book by Horváth and Kokoszka (2012).

A sequence of mean zero random functions $\{Y_t\}_{t=1}^T \subset \mathbb{H}$ follows a non-concurrent functional autoregressive process of order 1 if:

$$Y_t = \Psi Y_{t-1} + \varepsilon_t \qquad t = 2, \dots, T \tag{23}$$

where $\{\varepsilon_t\}_{t\in\mathbb{N}}$ is a sequence of iid mean-zero innovation errors with values in $\mathbb{H}$ satisfying $\mathbb{E}[||\varepsilon_t||^2] < +\infty$ and Ψ is a linear bounded operator from $\mathbb{H}$ to itself. In particular, we will consider Ψ to be a Hilbert-Schmidt operator with kernel ψ, in such a way that:

$$(\Psi x)(u,v) = \int_c^d \int_e^f \psi(u,v;w,z)x(w,z)dwdz \qquad \forall x \in \mathbb{H},\ \forall (u,v) \in [c,d]\times[e,f] \tag{24}$$

In order to ensure existence of a stationary solution to the functional AR(1) equation (23), one has to require the existence of an integer $j_0 \in \mathbb{N}$ such that $||\Psi||^{j_0} < 1$ (Bosq 2000, Lemma 3.1).

A very popular estimator of Ψ can be derived following a procedure similar to the Yule-Walker estimation in the scalar setting. Calling Γ_0 and Γ_1 the autocovariance and the lag-1 autocovariance operators respectively and proceeding similarly to Horváth and Kokoszka (2012), one can derive the operatorial equation:

$$\Gamma_1 = \Psi\Gamma_0 \tag{25}$$

A natural idea may consist in computing estimators $\hat{\Gamma}_0$, $\hat{\Gamma}_1$ from historical data and defining then $\hat{\Psi} = \hat{\Gamma}_0\hat{\Gamma}_0^{-1}$. Unfortunately, the inverse operator Γ_0^{-1} is unbounded on $\mathbb{H}$ (Horváth and Kokoszka 2012), however, thanks to Γ_0 being a symmetric, compact, positive-definite operator, one can exploit its spectral decomposition to introduce a pseudo-inverse operator $\Gamma_{0,M}^{-1}$, defined as:

$$\Gamma_{0,M}^{-1}x = \sum_{j=1}^{M} \lambda_j^{-1}\langle x, \xi_j\rangle \xi_j \qquad \forall x \in \mathbb{H} \tag{26}$$

where $\xi_1, \dots, \xi_M$ are the first M normalized functional principal components (FPC's), $\lambda_1, \dots, \lambda_M$ are the corresponding eigenvalues and $\langle x, \xi_1\rangle, \dots, \langle x, \xi_M\rangle$ are the scores of x along the FPC's. We formally define ξ_i and λ_i as eigenfunctions and eigenvalues that solve the functional equation:

$$\Gamma_0\xi_i = \lambda_i\xi \qquad i = 1, \dots, M \tag{27}$$

Appendix B is dedicated to the illustration of two different estimation techniques for ξ_i and

λ_i, one based on a discretization of functions on a fine grid and the other designed starting from an expansion of data on a finite basis system,

We can now combine (25) and (26), plugging in estimated eigenfunctions and eigenvalues and calling $\hat{\Gamma}_{0,M}^{-1}$ the resulting estimator of $\Gamma_{0,M}^{-1}$, to finally derive:

$$\hat{\Psi}_M = \hat{\Gamma}_1 \hat{\Gamma}_{0,M}^{-1} x \tag{28}$$

$$\hat{\Psi}_M x = \frac{1}{T-1} \sum_{i,j=1}^{M} \sum_{t=1}^{T} \hat{\lambda}_j \langle x, \hat{\xi}_j \rangle \langle Y_{t-1}, \hat{\xi}_j \rangle \langle Y_t, \hat{\xi}_i \rangle \hat{\xi}_i \qquad \forall x \in \mathbb{H} \tag{29}$$

Notice that the operator $\hat{\Gamma}_{0,M}^{-1}$ is bounded on $\mathbb{H}$ if $\hat{\lambda}_j$ are strictly greater than zero for $j = 1, \ldots, M$. Nevertheless, even if such condition is met, in practice one should cautiously select the number of principal components M, because very small eigenvalues will result in very high reciprocals $\hat{\lambda}_j^{-1}$, providing in practice unbounded estimates of $\Gamma_{0,M}^{-1}$. Such observation motivated Didericksen et al. (2010) to add a positive baseline to the estimated eigenvalues $\hat{\lambda}_j$. This small modification improves the estimation of the operator Ψ, and most importantly, contributes to weaken the dependency of $\hat{\Psi}_M$ on M.

A different yet simpler forecasting procedure has been proposed by Aue et al. (2012) for one dimensional functional data and will be here extended to the two-dimensional setting. Calling once again $\xi_1, \ldots, \xi_M$ the first M principal components, we can decompose the functional time series as follows:

$$Y_t(u,v) = \sum_{j=1}^{M} \langle Y_t, \xi_j \rangle \xi_j(u,v) + e_t(u,v) = \tag{30}$$

$$= \boldsymbol{Y}_t^T \boldsymbol{\xi}(u,v) + e_t(u,v) \tag{31}$$

where $\boldsymbol{Y}_t = [\langle Y_t, \xi_1 \rangle, \ldots, \langle Y_t, \xi_M \rangle]^T$ contains the scores of the projection, $\boldsymbol{\xi}(u,v) = [\xi_1(u,v), \ldots, \xi_M(u,v)]^T$ and $e_t(u,v)$ is the approximation error due to the truncation of the expansion on the first M principal components. Neglecting the approximation error e_t, one can prove that the vector $\boldsymbol{Y}_t$ follows a multivariate autoregressive process of order 1. Plugging in the estimated FPC's $\hat{\xi}_1, \ldots, \hat{\xi}_M$, the parameters of the resulting model can be easily estimated using classical multivariate statistical techniques and $\boldsymbol{Y}_{T+1}$ is then forecasted based on historical data $Y_1, \ldots, Y_T$. The predicted function $\hat{Y}_{T+1}$ can be simply reconstructed as:

$$\hat{Y}_{T+1}(u,v) = \hat{\boldsymbol{Y}}_{T+1}^T \boldsymbol{\xi}(u,v) \tag{32}$$

Throughout the rest of the work, we will employ both the estimator $\hat{\Psi}_M$ (29) of the Hilbert-Schmidt operator Ψ and the forecasting routine (32) presented just above, eventually comparing them in Section 5 in terms of prediction performances.

4.2 Conformal Inference for a FAR(1)

In this section, we aim to adapt the previous estimator to the conformal inference setting. The goal is to accommodate the regression algorithm in order to estimate the point predictor $g_{\mathcal{I}_1}$ from the training set only.

As a preliminary step, given that the FAR(1) model has been presented for mean-centered data, one has to estimate the mean function $\hat{\mu}_{\mathcal{I}_1}$ *from the training set only* and consequently center all the observations in the training and calibration set around $\hat{\mu}_{\mathcal{I}_1}$. If the sample size at disposal is sufficiently large and if the stationarity assumption is fulfilled, it should make no great difference to estimate the population function μ with the sample mean $\hat{\mu} = \frac{1}{T}\sum_{t=1}^{T} Y_t$ or with its restriction on the training set $\hat{\mu}_{\mathcal{I}_1} = \frac{1}{m}\sum_{t\in\mathcal{I}_1} Y_t$. Another fundamental step is the estimation of functional principal components. Since in the CP framework we are allowed to use only the information from the training set in order to compute $\hat{\xi}_1, \dots, \hat{\xi}_M$, it is then natural to employ only training data $\{\boldsymbol{z}_h : h \in \mathcal{I}_1\}$ in such estimation routine.

In order to obtain the "Yule-Walker" estimator $\hat{\Psi}_M$ (29), one has to compute $\hat{\Gamma}_1$ and $\hat{\Gamma}_{0,M}^{-1}$ from the training set. While the computation of the sample pseudo-inverse of the autocovariance estimator is straightforward:

$$\hat{\Gamma}_{0,M}^{-1} x = \frac{1}{m}\sum_{j=1}^{M} \hat{\lambda}_j \langle x, \hat{\xi}_j\rangle \hat{\xi}_j \qquad \forall x \in \mathbb{H} \tag{33}$$

the CP counterpart of $\hat{\Gamma}_1$ is more delicate and requires further discussion. Recall indeed that the classical estimator for the lag-1 autocovariance operator from $Y_1, \dots, Y_T$ is:

$$\hat{\Gamma}_1 x = \frac{1}{T-1}\sum_{t=1}^{T-1} \langle Y_t, x\rangle y_{t+1} = \tag{34}$$

$$= \frac{1}{T-1}\sum_{t=2}^{T} \langle Y_{t-1}, x\rangle Y_t \tag{35}$$

In the CP setting, however, we could define three different estimators for Γ_1:

$$\hat{\Gamma}_1 x = \frac{1}{m-1} \sum_{t \in \mathcal{I}_1[1:m-1]} \langle Y_t, x \rangle Y_{t+1} \tag{36}$$

$$\hat{\Gamma}_1 x = \frac{1}{m-1} \sum_{t \in \mathcal{I}_1[2:m]} \langle Y_{t-1}, x \rangle Y_t \tag{37}$$

$$\hat{\Gamma}_1 x = \frac{1}{m} \sum_{t \in \mathcal{I}_1} \langle Y_{t-1}, x \rangle Y_t \tag{38}$$

where $x \in \mathbb{H}$ and $\mathcal{I}_1[i:j]$ contains the indices from the i-th to the j-th element of $\mathcal{I}_1$. Notice that the three estimators differ because if $t \in \mathcal{I}_1$, we have no assurance that $\{t-1\} \in \mathcal{I}_1$ or even $\{t+1\} \in \mathcal{I}_1$. We stress also the fact that the third operator is well-defined only if we reserve a burn-in set of length 1 at the front of the time series, in such a way that, if $\{2\} \in \mathcal{I}_1$, we can still compute the estimator. Among the three options, we will prefer the third one (38), since it averages over a larger set. One may also argue that such estimators are not coherent with the CP setting, since they are inevitably based on data from the calibration set. However, as mentioned before, we are considering the time series of *regression pairs* $\boldsymbol{Z}_t = (\boldsymbol{X}_t, Y_t)$, $t = 1, \ldots, T$. The key consideration is that, according to the FAR(1) model, we use as regressors the lagged version of the time series, namely $\boldsymbol{X}_t = Y_{t-1}$ and the regression couples becomes $\boldsymbol{Z}_t = (Y_{t-1}, Y_t)$, for each $t = 1, \ldots, T$. From this perspective, one could rephrase the definition of the sample covariance operator by making explicit its dependence from the regressor X_t instead of Y_{t-1}:

$$\hat{\Gamma}_1 x = \frac{1}{m-1} \sum_{t \in \mathcal{I}_1} \langle Y_{t-1}, x \rangle Y_t = \tag{39}$$

$$= \frac{1}{m} \sum_{t \in \mathcal{I}_1} \langle X_t, x \rangle Y_t \tag{40}$$

It is then straightforward to derive the estimator $\hat{\Psi}_{M,\mathcal{I}_1}$:

$$\hat{\Psi}_{M,\mathcal{I}_1} x = \frac{1}{m} \sum_{i,j=1}^{M} \sum_{t \in \mathcal{I}_1} \hat{\lambda}_j \langle x, \hat{\xi}_j \rangle \langle Y_{t-1}, \hat{\xi}_j \langle Y_t, \hat{\xi}_i \rangle \hat{\xi}_i = \tag{41}$$

$$= \frac{1}{m} \sum_{i,j=1}^{M} \sum_{t \in \mathcal{I}_1} \hat{\lambda}_j \langle x, \hat{\xi}_j \rangle \langle X_t, \hat{\xi}_j \langle Y_t, \hat{\xi}_i \rangle \hat{\xi}_i \qquad \forall x \in \mathbb{H} \tag{42}$$

The point predictor finally becomes $\hat{Y}_{T+1} = g_{\mathcal{I}_1}(u, v; \boldsymbol{X}_{T+1}) = (\hat{\Psi}_{M,\mathcal{I}_1} \boldsymbol{X}_{T+1})(u, v) = (\hat{\Psi}_{M,\mathcal{I}_1} Y_T)(u, v)$.

In order to compute (42), it is first mandatory to project the calibration set onto the EPFC's in order to compute the scores $\langle X_h, \hat{\xi}_j \rangle$ for $h \in \mathcal{I}_2$. Notice that, in the non-Conformal setting, such step comes for free when performing FPCA. In the CP context, however, EPFC's are computed from the training set only, therefore, in order to access the scores of the calibration set, one needs to explicitly add this projection step.

4.3 Other forecasting algorithms

We finally introduce a model that may appear simplistic, since it neglects time dependence between different points of the functional domain, but that in practice provides quite satisfying results. The prediction method assumes an autoregressive structure in each location (u, v) of the domain, ignoring the dependencies between different points. More formally, we define the concurrent FAR(1) as:

$$X_t(u, v) = \psi_{u,v} X_{t-1}(u, v) + \varepsilon_t(u, v) \qquad \forall (u, v) \in [c, d] \times [e, f],\ t = 2, \dots, T \tag{43}$$

where $\psi_{u,v} \in \mathbb{R}$. Supposing to have observed all functional data $y_1, \dots, y_T$ on the same two-dimensional grid $\{(u_i, v_j)\}$ with $i = 1, \dots, N_1$ and $j = 1, \dots, N_2$. The goal becomes to estimate ψ_{u_i,v_j} for each location (u_i, v_j) in the grid.

In order to fix a benchmark on the forecasting performances, we will employ as reference regression algorithm the naive predictor $\hat{Y}_{T+1} = \boldsymbol{X}_{T+1} = Y_T$, which coincides with the function at the previous time, thus ignoring the autoregressive structure at all.

5 Simulation study

In this section, we will evaluate the previously presented procedure through a simulation study. The goal is twofold: we aim to assess the quality of the proposed Conformal Prediction bands and, at the same time, evaluate different point predictors in terms of the resulting prediction regions. We will employ as a data generating process a FAR(1) model in order to compare the different estimation routines presented in Section 4.1. Alongside, we will juxtapose the performances of such estimation procedures with the simpler algorithms presented in Section 4.3, with the intention of fixing a reference target on the prediction task, in order to understand how much can be gained by utilizing the autoregressive struc-

ture of data. Moreover, by including forecasting algorithms that are not coherent with the data generating process, we can illustrate how the presented CP procedure performs when a good point predictor $g_{\mathcal{I}_1}$ is not available. Although as reported in Section 3 an accurate forecasting algorithm is sufficient to guarantee asymptotic validity, we will see that in the performed simulations CP bands will be valid even when such assumption will not hold.

It is important to clarify how we will evaluate the various regression algorithms in the different scenarios. Since this work is focused on uncertainty quantification in the context of two-dimensional functional data, we will compare forecasting performances by means of the resulting Conformal Prediction bands. Firstly and foremost, we will estimate the unconditional coverage by computing the *empirical unconditional coverage* in order to compare it with the nominal confidence level $1-\alpha$. In the second place, we will consider the size of the prediction bands obtained since, intuitively, a small prediction band is preferable because it includes subregions of the sample space where the probability mass is highly concentrated (Lei et al. 2013) and it is typically more informative in practical applications.

Without loss of generality, throughout this section we will consider functions in $\mathbb{H} = \mathcal{L}^2([0,1]\times[0,1])$. In each scenario, we will compare the performances of five selected prediction algorithms, three of which do not exploit the autoregressive structure. To obtain further insights, we also include the errors obtained by assuming perfect knowledge of the operator Ψ. For ease of reference, we briefly describe these methods, and introduce some convenient notation:

- **EK** (Estimated Kernel) denotes the first estimation procedure presented in Section 4.1, where we explicitly compute $\hat{\Psi}_M$ as prescribed by (29) and then set $\hat{Y}_{T+1} = \hat{\Psi}_M Y_T$.

- **EK+** (Estimated Kernel improved) is a modification of the above method, where eigenvalues $\hat{\lambda}_i$ are replaced by $\hat{\lambda}_i + 1.5(\hat{\lambda}_1 + \hat{\lambda}_2)$, as recommended by Didericksen et al. (2010).

- **VAR-efpc** denotes the forecasting procedure (32) presented in Section 4.1, where we exploit the expansion on the estimated functional principal components and forecast Y_{T+1} thanks to the underlying VAR(1) model.

- **Concurrent** refers to the forecasting algorithm based on the estimation of the concurrent FAR(1) model (43).

- **Naive**: we just set $\hat{Y}_{T+1} = Y_T$. This method does not attempt to model temporal dependence, it is included to see how much can be gained by exploiting the autoregressive

structure of data.

- **Oracle**: we set $Y_{T+1} = \Psi Y_T$, using the actual Ψ from which data are simulated. This point predictor is clearly not available in practical application, but it is interesting to include it in order to see if poor predictions might be due to poor estimation of Ψ.

When it is required (namely in EK, EK+, VAR-efpc), FPCA is performed using the discretization approach, as motivated in Appendix B. The number of principal components is selected by the cumulative proportion of variance criterion. Calling $\hat{\lambda}_1, \dots, \hat{\lambda}_M$ the M largest estimated eigenvalues, we choose $M \in \mathbb{N}$ such that $\sum_{j=1}^{M} \hat{\lambda}_j / \sum_{j=1}^{\infty} \hat{\lambda}_j$ exceeds a predetermined percentage value, which is in this case fixed equal to 0.8. We noticed that, on average, this entails to select a number of harmonics between 4 and 6.

Throughout the whole simulation study, we set the significance level $\alpha = 0.1$. In the first simulation, in Section 5.1, we will fix the size b of the blocking scheme (15) equal to 1 and the sample size T will take values $19, 49, 99, 499$. Secondly, in Section 5.2, we will instead keep the sample size fixed equal to 119 and repeat the simulations with $b = 1, 3, 6$. As usually done in the time series setting, the first observation is taken into account as a covariate only and will neither take part of the training set, nor of the calibration set. The proportion of data in the training and in the calibration set are hence equal to one half of the remaining observations, *id est* $m = l = (T-1)/2$. Thanks to the chosen values of T, l and α, we can guarantee an actual coverage of $1 - \frac{\lfloor (l+1)\alpha \rfloor}{(l+1)} = 1 - \alpha$. For each value of T, we repeat the procedure by considering $N = 1000$ simulations. The simulations are performed using the R Programming Language (R Core Team 2020).

Following the implementation of Hörmann and Łukasz (2017), in order to simulate a sequence of functions $\{Y_t\}_{t=1,\dots,T}$ from a functional autoregressive process of order one:

$$Y_t(u,v) = \Psi Y_{t-1}(u,v) + \varepsilon_t(u,v) = \tag{44}$$

$$= \int_0^1 \int_0^1 \psi(u,v;w,z) Y_{t-1}(w,z) dw dz + \varepsilon_t(u,v), \qquad t = 1, \dots, M \tag{45}$$

we assume that observations lie in a finite dimensional subspace of the function space $\mathbb{H}$, spanned by orthonormal basis functions $\phi_1, \dots, \phi_M$, with $M \in \mathbb{N}$ representing the dimension

of such subspace. Therefore, we have:

$$Y_t(u,v) = \boldsymbol{\phi}(u,v)^T \mathbf{Y}_t \tag{46}$$

$$\varepsilon(u,v) = \boldsymbol{\phi}(u,v)^T \boldsymbol{\varepsilon}_t \tag{47}$$

$$\psi(u,v;w,z) = \boldsymbol{\phi}(u,v)^T \boldsymbol{\Psi} \boldsymbol{\phi}(w,z) \tag{48}$$

where $\boldsymbol{\phi}(u,v) = [\phi_1(u,v), \dots, \phi_M(u,v)]^T \in \mathbb{R}^M$, $\forall (u,v) \in [0,1] \times [0,1]$, $\mathbf{Y}_t$, $\boldsymbol{\varepsilon}_t \subset \mathbb{R}^M$, $\forall t = 1, \dots, M$ and $\boldsymbol{\Psi} \in \mathbb{R}^{M \times M}$. It follows that:

$$\mathbf{Y}_t = \boldsymbol{\Psi} \mathbf{Y}_{t-1} + \boldsymbol{\varepsilon}_t \qquad t = 1, \dots, T \tag{49}$$

We set $\mathbf{Y}_0 = \boldsymbol{\varepsilon}_0$ to initialize the procedure and then perform 50 burn-in iterations, in order to achieve stationarity.

The basis system $\phi_1, \dots, \phi_M$ is constructed as the tensor product basis of two cubic B-spline systems $\{g_i\}_{i=1,\dots,M_1}$, $\{h_j\}_{j=1,\dots,M_2}$, defined respectively on $[0,1]$ and $[0,1]$. We set $M_1 = M_2 = 5$, in such a way that $M = 25$. Notice that, by including more functions, we will better approximate the space $\mathbb{H}$, though inevitably producing rougher curves. On the other hand, by reducing the size of the basis system, one renounce to have a good representation of $\mathbb{H}$, but this permits to obtain smoother functions. The choice proposed for M is arbitrary, but provides a good compromise between the two presented extremes. For an exhaustive discussion on the tensor product basis system, we refer to Appendix B.0.2.

The matrix $\boldsymbol{\Psi}$ is defined as $\boldsymbol{\Psi} := 0.9 \frac{\tilde{\boldsymbol{\Psi}}}{||\tilde{\boldsymbol{\Psi}}||_F}$, with $\tilde{\boldsymbol{\Psi}}$ having diagonal values equal to 0.8 and out-diagonal elements equal to 0.3. One can easily prove that, if relation (48) holds, then $||\Psi|| = ||\boldsymbol{\Psi}||_F$, where $||.||$ is the usual operatorial norm and $||.||_F$ denotes the Frobenius norm. Innovation errors $\boldsymbol{\varepsilon}_t$ are independently sampled from a multivariate Student's t-distribution, with 4 degrees of freedom and scale matrix $\boldsymbol{\Sigma}$ having diagonal elements equal to 0.5 and out-diagonal entries equal to 0.3. We report in Figure 3 an example of the first three realizations of a simulated Functional Autoregressive Process of order one, represented on a grid of 10^4 points[2].

[2] A GIF of the FAR(1) process evolution can be found in this GitHub repository.

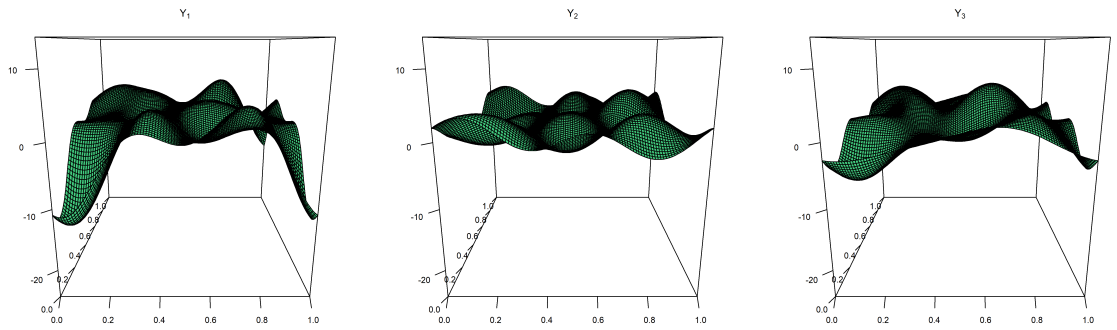


Figure 3: Example of the first three realizations (Y_1, Y_2, Y_3) of a simulated Functional Autoregressive Process of order one.

5.1 Increasing the sample size

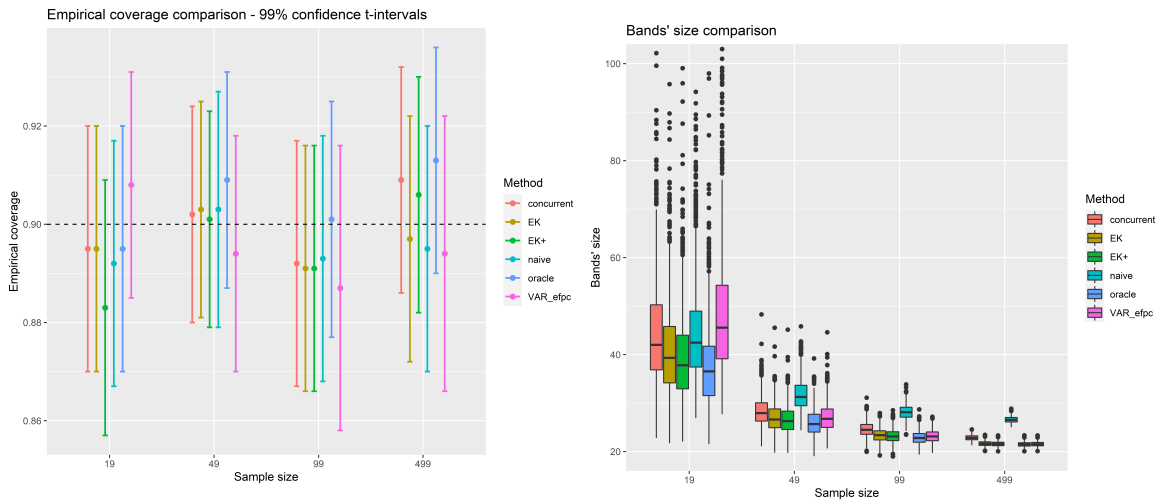
As mentioned before, we first fix the size b of the blocking scheme equal to 1 and let the sample size T take values $19, 49, 99, 499$.

Figure 4a shows the empirical coverage, together with the related 99% confidence interval. Specifically, the empirical coverage is computed as the fraction of the $N = 1000$ replications in which y_{T+1} belongs to $\mathcal{C}_{T,1-\alpha}$, and the confidence interval is reported in order to provide an idea of the variability of the phenomenon, rather than to make inferential conclusion on the unconditional coverage in the various settings. We stress the fact that different point predictors will intrinsically have dissimilar coverages, consequently this analysis aims to compare forecasting algorithm in terms of their predictive performances. We can appreciate that, in all the cases, the 99% confidence interval for the empirical coverage includes the nominal confidence level, regardless of the sample size at disposal. Moreover, it's interesting to notice that, even when an accurate forecasting algorithm $g_{\mathcal{I}_1}$ is not available (namely with Concurrent and Naive), the proposed CP procedure still outputs valid prediction regions.

Following once again the work of Diquigiovanni et al. (2021c), we define the size of a two-dimensional prediction band as the *volume* between the upper and the lower surfaces that define the prediction band:

$$\mathcal{Q}(s_{\mathcal{I}_1}) := \int_0^1 \int_0^1 2k s_{\mathcal{I}_1}(u, v) du dv = 2k \tag{50}$$

Figure 4b reports the boxplots concerning the size of the $N = 1000$ prediction bands, while in Table 1b we collected mean sizes to allow for easier comparison.



(a) Empirical coverage of CP bands. The dashed line represents nominal coverage $1 - \alpha$.

(b) Size of CP bands.

Figure 4: Results of first simulation study.

Method	T			
	19	49	99	499
concurrent	0.895	0.902	0.892	0.909
EK	0.895	0.903	0.891	0.897
EK+	0.883	0.901	0.891	0.906
Naive	0.892	0.903	0.893	0.895
Oracle	0.895	0.909	0.901	0.913
VAR-efpc	0.908	0.894	0.887	0.894

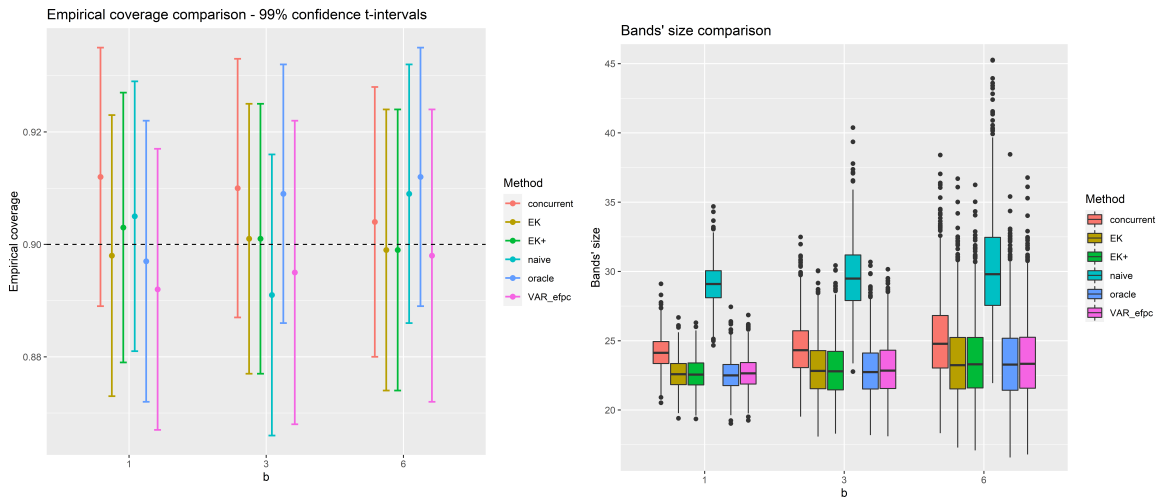
(a) Empirical coverage of CP bands

Method	T			
	19	49	99	499
concurrent	44.66	28.33	24.59	22.84
EK	41.08	26.98	23.42	21.64
EK+	39.65	26.63	23.23	21.58
Naive	44.98	31.75	28.14	26.58
Oracle	37.75	25.99	22.87	21.50
VAR-efpc	49.30	27.05	23.17	21.54

(b) Average size of CP bands

Table 1: Results of first simulation study.

One can notice that the size tends to decrease as long as the number of observations T increases, hence improving the efficiency of the prediction sets. As expected, Naive predictor provides larger prediction bands and while the difference is less emphasized with small sample sizes, when T grows the size of the prediction regions of other methods systematically dominates the Naive's one. On the other hand, EK and EK+, that are both based on the estimation of autoregressive operator Ψ, provide the tightest prediction bands, not only when numerous observations is available, but also in small sample sizes scenario. Moreover, one can notice that EK+ do not significantly improves EK neither in terms of coverage, nor



(a) Empirical coverage of CP bands. The dashed line represents nominal coverage $1 - \alpha$.

(b) Size of CP bands.

Figure 5: Results of second simulation study.

in terms of band size. We acknowledge that, when $T = 19$, VAR-efpc performs remarkably worse than the other methods. Indeed, the aforementioned method produces wider prediction bands, which are further source of the higher empirical coverage in Figure 4a. However, when the sample size increases, such forecasting algorithm performs comparably with the already mentioned EK and EK+. Finally, although the Conformal Prediction bands produced by the oracle predictor are obviously the most performing one, we can appreciate that both EK and EK+ provide CP bands with coverage and size comparable to the theoretically perfect oracle forecasting method.

5.2 Increasing the blocking scheme size

We repeat here the previous simulation with a blocking scheme of increasing size b and sample size T fixed equal to 119. Results are reported in Figure 5 and Table 2

Once again, in all the scenarios the 99% confident interval for the empirical coverage includes the target level of $1 - \alpha$, hence confirming the validity of the Conformal Prediction bands even for higher values of b. Moreover, one can notice that, as already pointed out by Diquigiovanni et al. (2021a) in the one-dimensional functional setting, the band size tends to decrease when b decreases, thus providing more efficient prediction regions. A comparison of the different forecasting algorithms performances validates the consideration in Section 5.1.

Method	b		
	1	3	6
concurrent	0.912	0.910	0.904
EK	0.898	0.901	0.899
EK+	0.903	0.901	0.899
Naive	0.905	0.891	0.909
Oracle	0.897	0.909	0.912
VAR-efpc	0.892	0.895	0.898

(a) Empirical coverage of CP bands

Method	b		
	1	3	6
concurrent	24.15	24.51	25.12
EK	22.63	23.00	23.58
EK+	22.62	22.99	23.59
Naive	29.13	29.65	30.29
Oracle	22.56	22.95	23.55
VAR-efpc	22.69	23.04	23.65

(b) Average size of CP bands

Table 2: Results of second simulation study.

6 Case study: Black Sea level anomaly forecasting

6.1 Dataset

Having proved the benefits of the proposed method on simulated data, we now aim to illustrate its application potential on a proper case study. We will consider data from Copernicus (add reference), the European Union's Earth observation program, which collects vast amounts of global data from satellites and ground-based, airborne, and seaborne measurement systems, in order to provide information to help service providers, public authorities, and other international organizations.

More specifically, we will analyze a data set from Copernicus Climate Change Service (C3S), a project operated by the European Center for Medium-Range Weather Forecasts (ECMWF), collecting daily sea level anomalies of the Black Sea in the last twenty years (Mertz and Legeais 2018). Sea level anomalies are measured as the height of water over the mean sea surface in a given time and region. Anomalies are computed with respect to a twenty-year mean reference period (1993-2012). Up-to-date altimeter standards are used to estimate the sea level anomalies with a mapping algorithm dedicated to the Black Sea region. Observations are collected on a spatial raster, with a 0.125° resolution both on the longitude and on the latitude axis. Since observations are collected on a geoid, the domain actually lies on a manifold, however, because both longitude and latitude ranges are very small (14° and 7° respectively), we will ignore this detail and assume data to be observed on a rectangular grid. The resulting lattice can hence be considered as the Cartesian product of a grid on the longitude axis made by $N_1 = 120$ points and a latitude grid of $N_2 = 56$ points. We will refer to (u_i, v_j), with $i = 1, \dots, N_1$ and $j = 1, \dots, N_2$ as the point (i, j)-th of such

two-dimensional mesh. Since the Black Sea hasn't a rectangular shape, we will consider each surface to be identically equal to zero outside the perimeter of the sea.

Altimetry instruments give access to daily observations of the Sea Surface Height (SSH) above the reference ellipsoid (see Figure 6), which is calculated as the difference between the orbital altitude of the satellite and the measured altimetric distance of the satellite from the sea at time t:

$$SSH_t(u_i, v_j) = \text{orbital altitude} - \text{altimetric range} \tag{51}$$

Staring from this information, one can compute the Mean Sea Surface (MSS) as the temporal mean of SSH over a reference period with $\tilde{N}$ observations. The mean surface level above the reference ellipsoid is computed from a twenty-year reference period (1993-2012).

$$MSS_N(u_i, v_j) = \frac{1}{\tilde{N}} \sum_{t=1}^{\tilde{N}} SSH_t(u_i, v_j) \tag{52}$$

The Sea Level Anomaly at time t in (u_i, v_j), $SLA_{t,\tilde{N}}(u_i, v_j)$, is finally computed as the anomaly of the signal SSH_t around the mean component $MSS_{\tilde{N}}(u_i, v_j)$:

$$SLA_{t,\tilde{N}}(u_i, v_j) = SSH_t(u_i, v_j) - MSS_{\tilde{N}}(u_i, v_j) \tag{53}$$

We will eventually drop the subscript $\tilde{N}$ for ease of notation and just refer to $SLA_t(u_i, v_j)$. Since we are settling the study in a functional data analysis framework, we will consider the time series $\{SLA_t\}$, without making explicit the dependence on the bivariate domain.

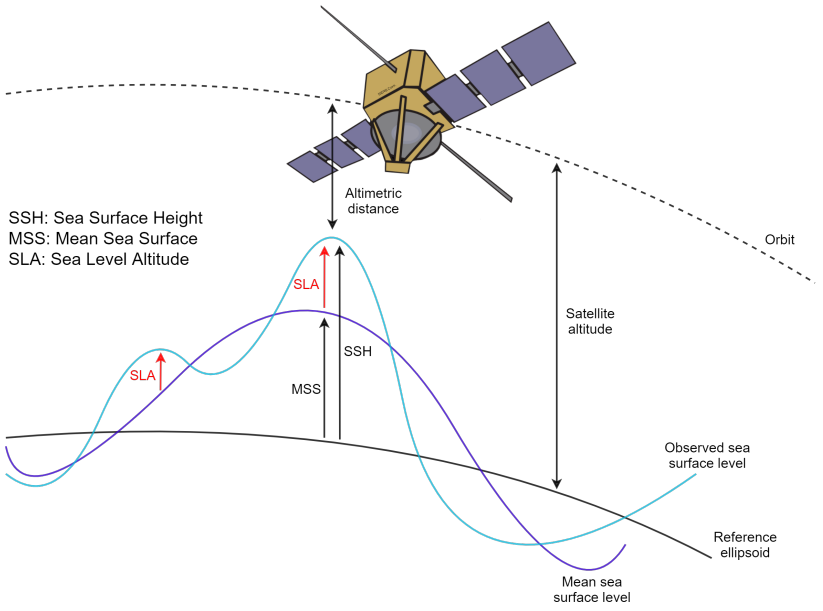
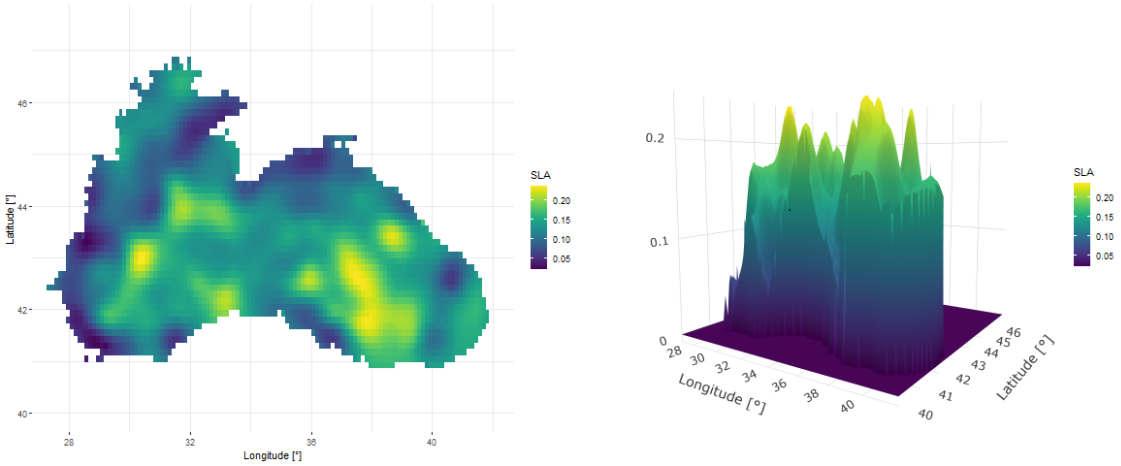


Figure 6: Data measurement process. The image is a replica of Figure 1 in Copernicus' Product User Guide and Specification v2.4.



(a) Raster representation

(b) Surface representation

Figure 7: Sea Level Anomaly [m] on 01/01/2018.

6.2 Preliminary analysis

If possible, one would preferably forecast directly the time series of Sea Level Anomalies (SLA_t). However, given the nature of the dataset, we expect anomalies to exhibit a periodical behaviour and perhaps also a trend component due to recent climate changes. In order to investigate this assumption, we should proceed by testing the functional time series $\{SLA_t\}_t$ for stationarity. However, despite for one dimensional functional time series one could resort to the test proposed by Horváth et al. (2014), to the best of our knowledge ad-hoc stationarity test for two-dimensional functional time series hasn't still be implemented. For such reason, and aware of the limits of this approach, we will resort here to the analysis of univariate time series $SLA_t(u_i, v_j)$, fixing some random locations (u_i, v_j). We stress the fact that stationarity is indeed not necessary to obtain valid CP bands, but as proved by Chernozhukov et al. (2018), it is a sufficient condition to guarantee the first assumption of Theorem 1, that we would hence like to be satisfied.

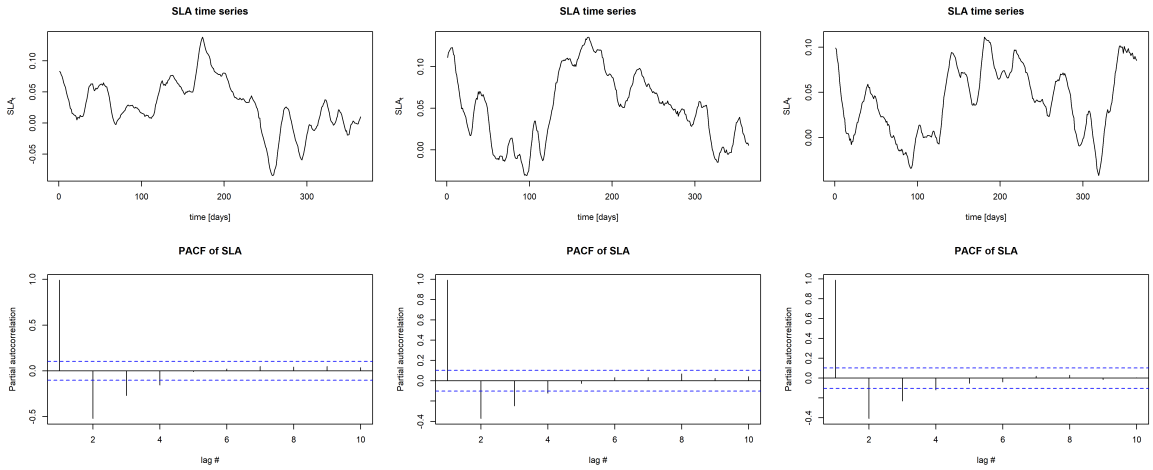


Figure 8: Univariate time series of Sea Level Anomaly (SLA_t), with correspondent ACF plots. Each column represents a different location. The time windows refer to data from 01/01/2014 to 01/01/2015.

In Figure 8 we report univariate time series $SLA_t(u_i, v_j)$ in 3 fixed locations (u_i, v_j), along with the correspondent partial autocorrelation function (PACF) plots. We acknowledge not negligible partial autocorrelation up to lag 2 or 3 depending on circumstances and the evident presence of a cyclical behaviour. Moreover, Augmented Dickey Fuller (ADF) stationarity test fails to reject the null hypothesis of unit root against the alternative one of a stationary

process. As usually done in time series analysis, we proceed by differentiating $\{SLA_t\}_t$, hence considering the time series of first differences with lag 1 $\{\Delta SLA_t\}_t$ defined as $\Delta SLA_t := SLA_t - SLA_{t-1}$. As reported in Figure 9, differentiated data still exhibit high partial autocorrelation for lags greater than one. A similar behaviour has also been found after a seasonal differentiation, where we employed as differentiation lag both a delay of 29 days, namely the moon phase cycle, and a lag of 365 days, coinciding with the Earth revolution time. Since we aim to eventually fit a Functional Autoregressive Process of order one, and also because in many locations (u_i, v_j) the related univariate time series still exhibit a non-stationary behaviour, we proceed with a second differentiation, hoping to obtain stationary time series with negligible partial autocorrelation for lags greater than one.

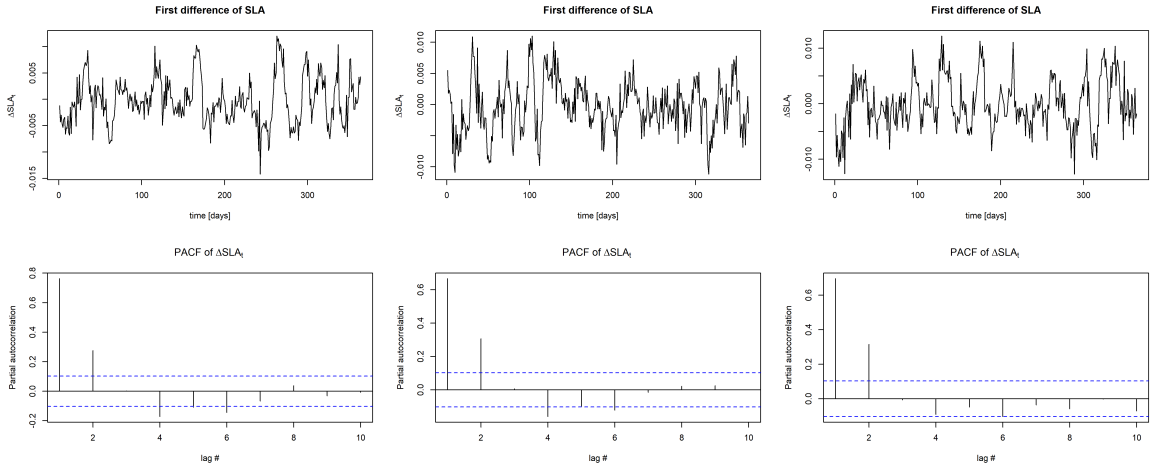


Figure 9: Univariate time series of ΔSLA_t, with correspondent ACF plots. Each column represents a different location. The time windows refer to data from 01/01/2014 to 01/01/2015.

We can finally appreciate in Figure 10 what appear to be strongly mixing time series, as also confirmed by ADF test. Moreover, the PACF plots exhibit almost null partial autocorrelation for lags greater than one. This last consideration provides a solid motivation to proceed with modelling with a FAR(1) the time series of second differences $\{Y_t\}_t$, formally defined as:

$$Y_t := \Delta^2 SLA_t = (SLA_t - SLA_{t-1}) - (SLA_{t-1} - SLA_{t-2}) \tag{54}$$

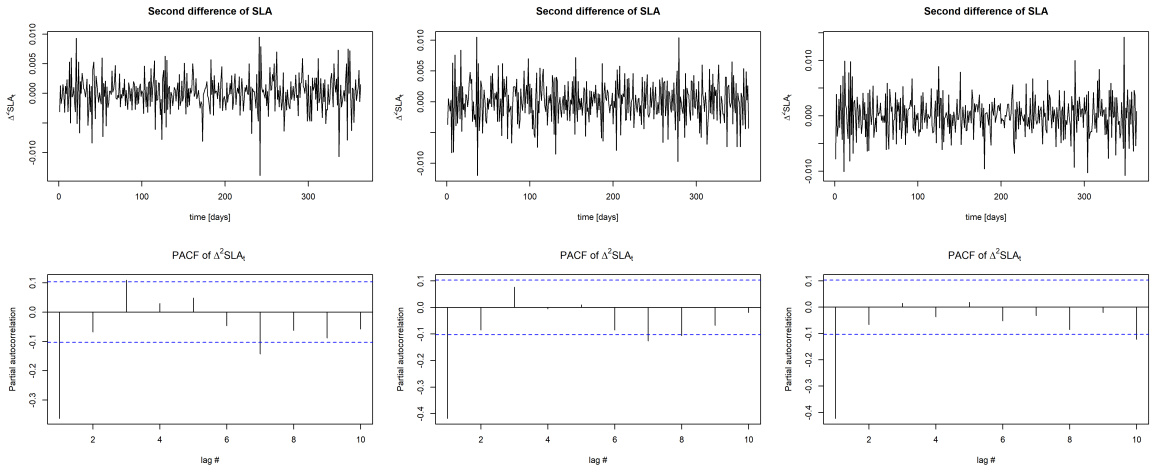


Figure 10: Univariate time series of $\Delta^2 SLA_t$, with correspondent ACF plots. Each column represents a different location. The time windows refer to data from 01/01/2014 to 01/01/2015.

6.3 Results

The case study will employ a common rolling estimation framework which recalculates the model parameters on a daily basis and consequently shifts the entire training, calibration and test windows by 24 hours, as shown in Figure 11. As before, we will use a random split of data in the training and calibration sets, with split proportion equal to 50%. The significance level α is once again fixed equal to 0.1 and the sample size T is chosen equal to 99 in order to guarantee an actual coverage of $1 - \frac{\lfloor (l+1)\alpha \rfloor}{(l+1)} = 1 - \alpha$. The size of the blocking scheme will instead be fixed equal to 1, since, as motivated in Section 5.2, this choice produces the narrowest prediction bands. The rolling window will be shifted 1000 times, thus iterating for almost three years the forecasting of the next day based on the last 99 observations. More specifically, we will consider a rolling window ranging from 01/01/2017 to 04/01/2020.

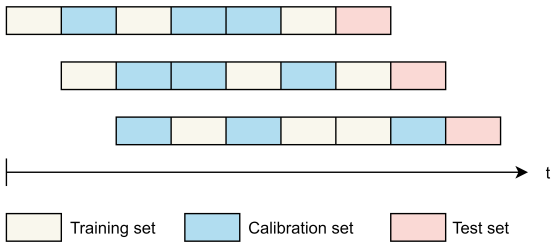


Figure 11: Training-calibration-test split in a rolling window scenario.

The point predictors used throughout this application will be the same described in Section 5. The number of Functional Principal Components is this time selected equal to 8 through a dedicated validation procedure. Given the vast number of observations at our disposal, we can indeed reserve a fraction of them as a validation set, and use it to estimate the optimal number of principal components by means of the Mean Squared Error (72) between the predicted curve and the actual forthcoming one. By choosing beforehand the number of harmonics rather than selecting them each time by means of the cumulative proportion of variance (as instead done in Section 5), we expect to obtain better results in terms of predictive performances. Specifically, historical data from 01/01/2014 to 04/01/2017 are used in order to determine the optimal number of harmonics.

For each shift of the rolling window and for each forecasting algorithm, we will check if y_{T+1} belongs to the $\mathcal{C}_{T,1-\alpha}$, saving also the size of the corresponding prediction band. After having collected such results, we can calculate the average coverage. We stress the fact that such quantity does not provide a good estimate of the empirical coverage of employed methods, since it is computed from correlated data. Indeed, by shifting the rolling window by one day at the time, we are inevitably including in the new window all the previous data but once. Nevertheless, a similar setting is often used in practical application, and it is still interesting to compare performances of the different point predictors in this scenario.

We report in Figure 12 the average coverage along with a 99% confidence interval. Notice that in this case the confidence interval may be biased, due to correlation between data used to construct it, however, we still decided to include it in order to assess the dispersion of the average coverage around the mean. We can notice that, in this more complex scenario, the Naive predictor struggles to output valid prediction regions, since average coverage is quite far from the nominal one. This is coherent with the theory presented in Section 3, because validity of CP intervals is guaranteed only when a good point predictor is available. On the other hand, all the other methods provide prediction bands with average coverage very close to $1 - \alpha$. For what concerns the size of the prediction bands, the Naive ones are by far the widest ones (see Figure 13), and, as pointed out before, this fact does not reflect in a greater coverage compared to the other methods. On the other hand, prediction bands obtained with forecasting algorithms that model the autoregressive structure provide narrower prediction regions. Among these, we can see that the non-concurrent FAR(1) is the most performing one, despite the way in which it is estimated (namely with EK, EK+ or VAR-efpc). Nevertheless, also the concurrent FAR(1) model provides very tight prediction bands, almost comparable with the ones produced by the non-concurrent prediction algorithm.

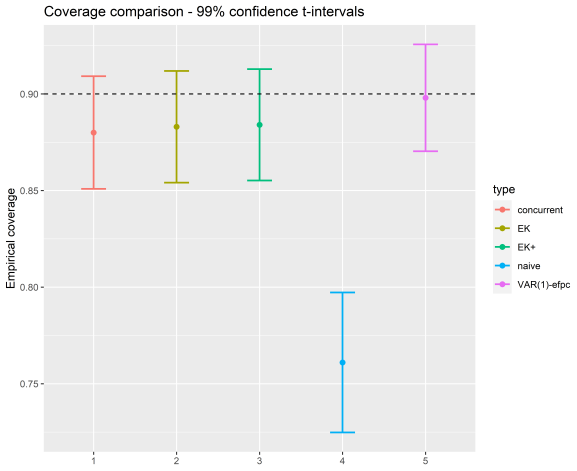
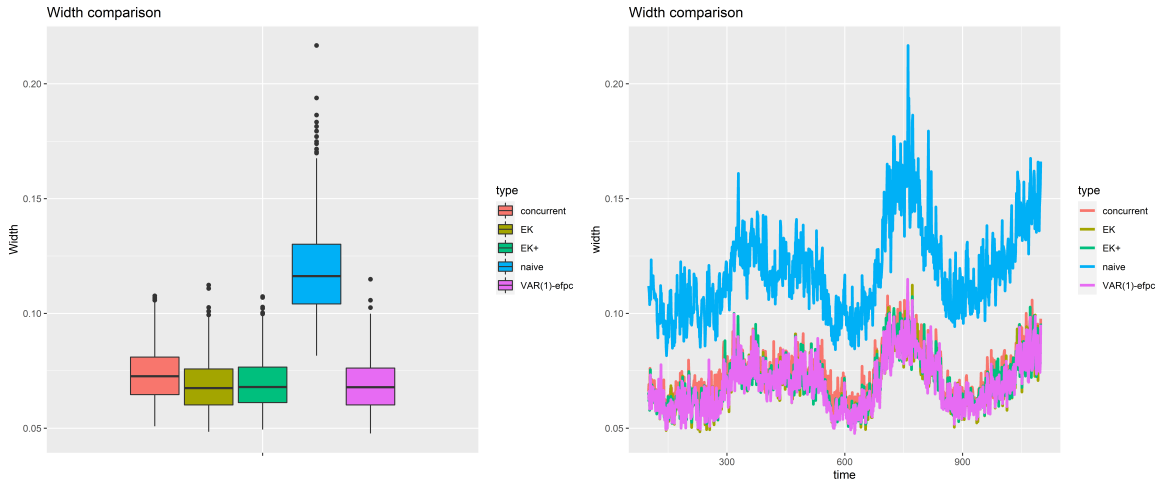


Figure 12: Coverage of CP bands. The dashed line represents nominal coverage $1 - \alpha$.



(a) Boxplot of CP bands' size.

(b) Evolution of CP bands' size during time.

Figure 13: Size of CP bands.

7 Conclusions and Further Developments

In this paper, we applied Conformal Prediction in order to quantify uncertainty in forecasting two-dimensional functional time series. Given the novelty of the subject, we have extended classical procedures from the one-dimensional functional framework, such as functional principal component analysis (FPCA). In particular, we have focused on extending the non-concurrent Functional Autoregressive process of order one, which represents the state of the art of functional time series modelling, proposing different estimation techniques. Moving from the work of Diquigiovanni et al. (2021a), we employed Split Conformal prediction, describing in Section 3 the randomization inference procedure already proposed by Chernozhukov et al. (2018) and adapting it to our specific setting. Despite Theorem 1 provides theoretical performance guarantees, we were interested in verifying empirical properties of CP bands, and, at the same time, testing and comparing different forecasting algorithms in terms of the resulting prediction regions. We proved the robustness of the proposed method, emphasizing the advantages of using a correctly specified point predictor in the procedure. We have finally applied the proposed technique to a real case study, employing a novel time series dataset (Mertz and Legeais 2018), which consists in daily observations of Sea Anomaly Level over the Black Sea during the last 20 years. In modelling such complex data, we had to introduce some major simplifications. Notice that this circumstance does not invalidate the method at all, but certainly leaves room for improvement. In particular, one could extend the modelling procedure by taking in to consideration the challenging nature of the domain, thus adapting the techniques to bivariate functional data on a manifold. On a more simple but useful level, we shall also extend the proposed scheme to more complex domains, thus going beyond the rectangular case exploiting numerical integration techniques for functions defined on a generic subset of $\mathbb{R}^2$. Another interesting development could regard the implementation of a stationarity test for two-dimensional functional data, perhaps extending the already mentioned work of Horváth et al. (2014). For what concerns FPCA though, we would like to acknowledge some recent improvements in the estimation of functional principal components (FPC's) for one-dimensional functional data, which may be extended to the two-dimensional framework of this paper. One may indeed argue the time-dependency is not considered in the proposed algorithms for FPC's estimation. functional principal component analysis is in fact a static procedure which ignores the information provided by the serial dependence structure of the functional data under study. Motivated by such considerations, Hörmann et al. (2015) proposed a dynamic version of FPCA which is based on a frequency domain approach and Trinka et al. (2021) developed two forecasting algorithms based on

functional singular spectrum analysis that incorporates the time-dependency into the decomposition of a functional time series. We conjecture that exploiting such techniques in the FPC's estimation could improve the forecasting performances of the proposed methods.

Acknowledgments

This work is partially supported by ACCORDO Attuativo ASI-POLIMI "Attività di Ricerca e Innovazione" n. 2018-5-HH.0, collaboration agreement between the Italian Space Agency and Politecnico di Milano.

Code

All the analysis are implemented using the R Programming Language (R Core Team 2020). Codes from Diquigiovanni et al. (2021c), implementing Conformal Inference in the functional setting, have been adapted to allow for functions defined on a bivariate domain. Functional Principal Component Analysis in the two-dimensional functional setting, estimation methods for the Functional Autoregressive Process of order one and the other forecasting algorithms are all implemented from scratch, as long as simulations and tests.
Codes are so far not publicly available, but the authors are at disposal for any clarification on the implementation details.

References

Alexander Aue, Diogo Norinho, and Siegfried Hörmann. On the prediction of stationary functional time series. *Journal of the American Statistical Association*, 110, 08 2012. doi: 10.1080/01621459.2014.909317.

Siva Balasubramanian, James Karrh, and Hemant Patwardhan. Audience response to product placements: An integrative framework and future research agenda. *Journal of Advertising*, 35:115–141, 10 2006. doi: 10.2753/JOA0091-3367350308.

D. Bosq. *Linear Processes in Function Spaces*. Springer New York, 2000.

Victor Chernozhukov, Kaspar Wuthrich, and Yinchu Zhu. Exact and robust conformal inference methods for predictive machine learning with dependent data, 2018.

Devin Didericksen, Piotr Kokoszka, and Xi Zhang. Empirical properties of forecasts with the functional autoregressive model. *Computnl Statist.*, 27:285–298, 01 2010. doi: 10.1007/s00180-011-0256-2.

Jacopo Diquigiovanni, Matteo Fontana, and Simone Vantini. Distribution-free prediction bands for multivariate functional time series: an application to the italian gas market, 2021a.

Jacopo Diquigiovanni, Matteo Fontana, and Simone Vantini. Conformal prediction bands for multivariate functional data, 2021b.

Jacopo Diquigiovanni, Matteo Fontana, and Simone Vantini. The importance of being a band: Finite-sample exact distribution-free prediction sets for functional data, 2021c.

A. Gammerman, V. Vovk, and V. Vapnik. Learning by transduction. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, UAI'98, page 148–155, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc. ISBN 155860555X.

D. Gervini. The functional singular value decomposition for bivariate stochastic processes. *Comput. Stat. Data Anal.*, 54:163–172, 2010.

Nicolás Hernández, Jairo Cugliari, and Julien Jacques. Simultaneous predictive bands for functional time series using minimum entropy sets, 2021.

L. Horváth and P. Kokoszka. *Inference for Functional Data with Applications*. Springer Series in Statistics. Springer New York, 2012. ISBN 9781461436553. URL https://books.google.it/books?id=OVezLB__ZpYC.

Lajos Horváth, Piotr Kokoszka, and Gregory Rice. Testing stationarity of functional time series. *Journal of Econometrics*, 179(1):66–82, 2014. ISSN 0304-4076. doi: https://doi.org/10.1016/j.jeconom.2013.11.002. URL https://www.sciencedirect.com/science/article/pii/S0304407613002327.

Boyin Huang, Peter W. Thorne, Viva F. Banzon, Tim Boyer, Gennady Chepurin, Jay H. Lawrimore, Matthew J. Menne, Thomas M. Smith, Russell S. Vose, and Huai-Min Zhang. Noaa extended reconstructed sea surface temperature (ersst), version 5, 2017.

Rob Hyndman and Han Lin Shang. Functional time series forecasting. *Journal of the Korean Statistical Society*, 38, 07 2009. doi: 10.1016/j.jkss.2009.06.002.

Rob Hyndman and Shahid Ullah. Robust forecasting of mortality and fertility rates: A functional data approach. *Computational Statistics & Data Analysis*, 51:4942–4956, 02 2007. doi: 10.1016/j.csda.2006.07.028.

Siegfried Hörmann and Piotr Kokoszka. Weakly dependent functional data. *The Annals of Statistics*, 38(3):1845 – 1884, 2010. doi: 10.1214/09-AOS768. URL https://doi.org/10.1214/09-AOS768.

Siegfried Hörmann, Łukasz Kidziński, and Marc Hallin. Dynamic functional principal components. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 77(2):319–348, 2015. doi: https://doi.org/10.1111/rssb.12076. URL https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/rssb.12076.

Siegfried Hörmann and Kidzinski Łukasz. freqdom.fda: Functional time series: Dynamic functional principal components, 09 2017. URL https://cran.r-project.org/web/packages/freqdom.fda/index.html.

Ivanescu and Andrada. A note on bivariate smoothing for two-dimensional functional data. *International Journal of Statistics and Probability*, 2, 04 2013. doi: 10.5539/ijsp.v2n2p102.

Vladislav Kargin and A. Onatski. Curve forecasting by functional autoregression. *Journal of Multivariate Analysis*, 99:2508–2526, 12 2005. doi: 10.1016/j.jmva.2008.03.001.

Christopher Kath and Florian Ziel. Conformal prediction interval estimations with an application to day-ahead and intraday power markets, 02 2019.

Byol Kim, Chen Xu, and Rina Foygel Barber. Predictive inference is free with the jackknife+-after-bootstrap, 2020.

Jochen Knaus. snowfall: Easier cluster computing (based on snow), 10 2015. URL https://CRAN.R-project.org/package=snowfall.

Jing Lei, Alessandro Rinaldo, and Larry Wasserman. A conformal prediction approach to explore functional data. *Annals of Mathematics and Artificial Intelligence*, 74, 02 2013. doi: 10.1007/s10472-013-9366-6.

Françoise Mertz and Jean-François Legeais, 06 2018. URL https://cds.climate.copernicus.eu/cdsapp#!/dataset/satellite-sea-level-black-sea?tab=overview.

NASA. Tes/aura l3 ozone monthly gridded v006. URL https://doi.org/10.5067/AURA/TES/TL3O3M.006.

P.-Muñoz, Dania, Francisco Mata, Noslen Hernández, and Isneri Talavera. Functional data analysis as an alternative for the automatic biometric image recognition: Iris application. *Computación y Sistemas*, 18:111–121, 03 2014. doi: 10.13053/CyS-18-1-2014-022.

H. Papadopoulos, Kostas Proedrou, V. Vovk, and A. Gammerman. Inductive confidence machines for regression. In *ECML*, 2002.

Sara Pintado and Juan Romo. On the concept of depth for functional data. *Journal of the American Statistical Association*, 104, 06 2009. doi: 10.1198/jasa.2009.0108.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2020. URL https://www.R-project.org/.

Lars Lau Rakêt. 2d functional data analysis, with applications to image analysis. Master's thesis, Statistics Department of Mathematical Sciences, Faculty of Science, University of Copenhagen, 2010.

J. Ramsay and B.W. Silverman. *Functional Data Analysis*. Springer Series in Statistics. Springer, 2005. ISBN 9780387400808. URL https://books.google.it/books?id=mU3dop5wY_4C.

Jacopo Rossini and Antonio Canale. Quantifying prediction uncertainty for functional-and-scalar to functional autoregressive models under shape constraints. *Journal of Multivariate Analysis*, 170, 10 2018. doi: 10.1016/j.jmva.2018.10.007.

Peter Rousseeuw and Sabine Verboven. Robust estimation in very small samples. *Computational Statistics and Data Analysis*, 40:741–758, 02 2002. doi: 10.1016/S0167-9473(02)00078-6.

Han Lin Shang. ftsa: An R package for analyzing functional time series. *The R Journal*, 5 (1):64–72, 2013. URL https://journal.r-project.org/archive/2013-1/shang.pdf.

Aldo Solari and Vera Djordjilović. Multi split conformal prediction, 2021.

Ryan J. Tibshirani, Rina Foygel Barber, Emmanuel J. Candes, and Aaditya Ramdas. Conformal prediction under covariate shift, 2020.

Jordan Trinka, Hossein Haghbin, and Mehdi Maadooliat. Functional time series forecasting: Functional singular spectrum analysis approaches, 2021.

V. Vovk, A. Gammerman, and G. Shafer. Algorithmic learning in a random world, 2005.

W. Wisniewski, David Lindsay, and Siân Lindsay. Application of conformal prediction interval estimations to market makers' net positions. In *COPA*, 2020.

Chen Xu and Yao Xie. Conformal prediction interval for dynamic time-series, 2021.

Hao Yan, Kamran Paynabar, and Jianjun Shi. Real-time monitoring of high-dimensional functional data streams via spatio-temporal smooth sparse decomposition. *Technometrics*, 60:181–197, 04 2018. doi: 10.1080/00401706.2017.1346522.

Gianluca Zeni, Matteo Fontana, and S. Vantini. Conformal prediction: a unified review of theory and new challenges. *ArXiv*, abs/2005.07972, 2020.

Lan Zhou and Huijun Pan. Principal component analysis of two-dimensional functional data. *Journal of Computational and Graphical Statistics*, 23(3):779–801, 2014. doi: 10.1080/10618600.2013.827986. URL https://doi.org/10.1080/10618600.2013.827986.

Tingyi Zhu and Dimitris N. Politis. Kernel estimates of nonparametric functional autoregression models and their bootstrap approximation. *Electronic Journal of Statistics*, 11(2):2876 – 2906, 2017. doi: 10.1214/17-EJS1303. URL https://doi.org/10.1214/17-EJS1303.

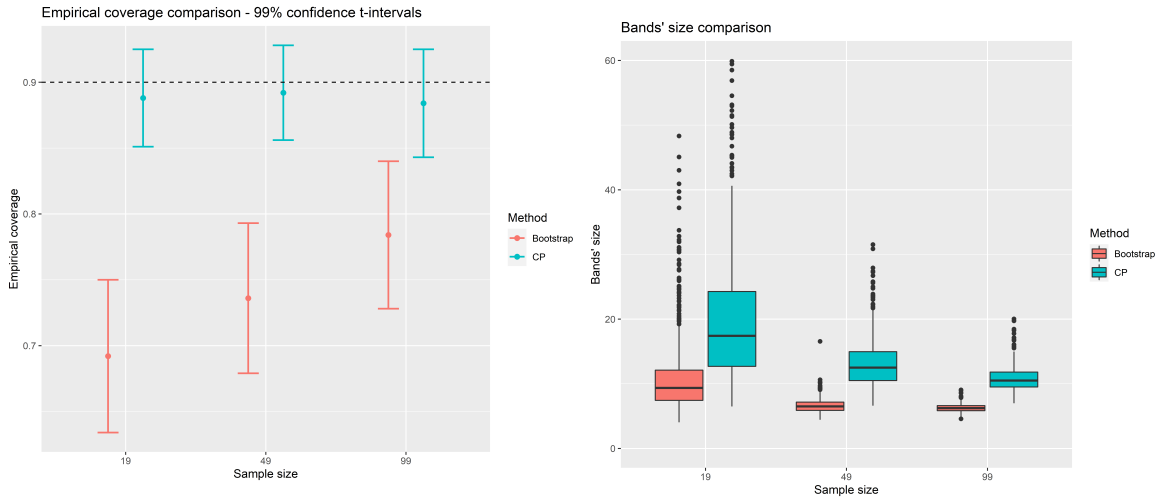
A CP vs Bootstrap

The objective of this section is to prove how Conformal Prediction can overcome some limitations of the Bootstrap approach. The main advantage of CP is the guarantee of asymptotic validity and the result in Theorem 1, that provides a theoretical bound on the miscoverage level of the resulting prediction bands. Nevertheless, we now aim to provide empirical justifications of the advantage of using the proposed approach over the Bootstrap one. In order to do so, we will first generate one-dimensional functional data from a functional autoregressive process of order one and compare the bands provided by the two methods. The forecasting algorithm here employed is the one in (32), that exploits the expansion on principal components and makes use of the underlying VAR(1) to obtain predictions for the next time. Both the choice of the point predictor and the preference of settling the simulation study in a one-dimensional setting are motivated by the possibility to exploit already implemented routines for obtaining Bootstrap prediction sets. We will in fact make use of the `R` package `ftsa` from Shang (2013), which implements in the `farforecast` function a multivariate time-series forecasting method coherent with (32). We slightly modified such routine in order to use the classical FPCA method and not the default robust principal component algorithm of Rousseeuw and Verboven (2002). Along with this adjustment, we also overrode the automatic order selection procedure of the VAR and manually set the order equal to 1 in accordance with our model. Notice that both these changes are necessary in order to employ the same point predictor and properly compare the performances of the two methods.

As usual, we simulate data exploiting a basis expansion on the tensor product basis of two Fourier basis systems with 5 basis each (see Section 5.2 for further details). The parameters of the data generating process are selected as follows: the matrix $\mathbf{\Psi}$ is defined as $\mathbf{\Psi} := 0.9\frac{\tilde{\mathbf{\Psi}}}{||\tilde{\mathbf{\Psi}}||_F}$, with $\tilde{\mathbf{\Psi}}$ having diagonal values equal to 0.8 and out-diagonal elements equal to 0.3. Innovation errors $\boldsymbol{\varepsilon}_t$ are independently sampled from a multivariate Student's t-distribution, with 4 degrees of freedom and scale matrix $\mathbf{\Sigma}$ having diagonal elements equal to 0.5 and out-diagonal entries equal to 0.3.

Whereas the significance level α will be fixed equal to 0.1, we will apply the evaluation procedure with increasing sample sizes: $T = 19, 49, 499$, in order to compare the convergence of the two methods. For each value of T, we will repeat the procedure by performing $N = 1000$ parallel simulations, implemented with the R package `snowfall` (Knaus (2015)). The size of the blocking scheme of the CP procedure is fixed equal to 1, because such choice provides narrower bands, as explained in Section 5.2.

Prediction bands will be compared by means of the empirical coverage first and then by considering their size (as defined by Diquigiovanni et al. 2021c in the case of prediction bands for one-dimensional functional data).



(a) Empirical coverage, the dashed line represents nominal coverage $1 - \alpha$.

(b) Band size.

Figure 14: Results of the simulation study.

We can notice in Figure 14a that the Bootstrap method always outputs regions with much lower coverage than the nominal one, despite fortunately generating better results when the sample size grows. On the other hand, CP bands are always valid, even when few number of observations are available. Such results confirm *global* validity of Conformal Prediction bands, in contrast to the pointwise coverage of bootstrap ones. For what concerns the efficiency of prediction regions, bootstrap ones are systematically narrower than the ones produced by CP, regardless of the number of observations (see Figure 14b). Notice however that such smaller sizes are related to different coverages and are thus not easily comparable. In conclusion, the simulation study showed how the bootstrap procedure struggles to produce valid predictions bands, displaying slow convergence to the nominal coverage. Moreover, as it is often the case for bootstrap techniques, computational times becomes quickly prohibitive, thus proving once again the advantage of employing the Conformal approach.

B Comparison of FPC's estimators for two-dimensional functional data

A fundamental aspect in the design of many regression algorithms is the estimation of functional principal components (FPC's) $\{\xi_i\}i \in \mathbb{N}$. We define FPC's as functions $\xi_i \in \mathcal{L}^2([c,d] \times [e,f])$ solving the functional equation:

$$\Gamma_0 \xi = \lambda \xi \tag{55}$$

In practice, we can only estimate the first $M \in \mathbb{N}$ eigenfunctions, implicitly performing dimensionality reduction. The choice of M is non-trivial and depends on the application framework. Whereas Kargin and Onatski (2005) suggested selecting it in a cross-validation setting, Aue et al. (2012) proposed a fully automatic criterion for choosing the number of principal components in terms of predictive performances. Plugging in the estimator of Γ_0, we define estimated eigenfunctions and eigenvalues as solutions of:

$$\hat{\Gamma}_0 \hat{\xi} = \hat{\lambda} \hat{\xi} \tag{56}$$

On a theoretical point of view, we would like to guarantee that population eigenfunctions can be consistently estimated by empirical eigenfunctions even in the non-iid framework of Functional Time Series. We refer to Theorem 16.2 in Horváth and Kokoszka (2012), which provides asymptotic arguments for such question.

The following subsections are dedicated to the estimation of eigenfunctions and eigenvalue in the two-dimensional functional case. Extending the work of Ramsay and Silverman (2005), we present two different estimation procedures, based respectively on a discretization of the functions to a fine grid and on a linear expansion of data on a finite set of basis functions. Among the two alternatives, we would resort to the function discretization. Indeed, such choice does not require the selection of a specific type of basis and not even the number of basis to employ, which are not trivial problem-dependent questions. Moreover, notice that also the discretization procedure can be seen as a particular case of the basis expansion, using as basis system indicator functions on the grid points. Furthermore, in the subsequent, Appendix B.1 we will demonstrate with a simulation study that there is no significant evidence to prefer one method against the other in terms of estimation quality.

We want to stress the fact that our methodology for FPCA is general, it works for two-dimensional functional data regardless of the presence of temporary dependence between

observations. Not modeling the serial dependence structure will not invalidate the PCA procedure, but we still have to require that the dynamic is stationary in order for the covariance estimation to make sense and thus to provide meaningful estimates.

B.0.1 FPCA by grid discretization

Consider a grid discretization $\{u_i\}_{i=1,\dots,N_1}$ of $[c,d]$ and $\{v_j\}_{j=1,\dots,N_2}$ of $[e,f]$, let $\omega_1 = \frac{1}{N_1}$, $\omega_2 = \frac{1}{N_2}$. For any point (u_i, v_j) of the discretized grid, the lhs of the functional eigenequation (56) can be rewritten as:

$$\hat{\Gamma}_0\hat{\xi}(u_i,v_j) = \int_c^d \int_e^f \hat{\gamma}_0(u_i,v_j;w,z)\hat{\xi}(w,z)dwdz \approx \tag{57}$$

$$\approx \omega_1 \sum_{l=1}^{N_1} \int_e^f \hat{\gamma}_0(u_i,v_j;u_l,z)\hat{\xi}(u_l,z)dz \approx \tag{58}$$

$$\approx \omega_1\omega_2 \sum_{l=1}^{N_1}\sum_{m=1}^{N_2} \hat{\gamma}_0(u_i,v_j;u_l,v_m)\hat{\xi}(u_l,v_m) \tag{59}$$

By defining $N := N_1N_2$ and introducing a bijection $\zeta : \{1,\dots,N_1\} \times \{1,\dots,N_2\} \to \{1,\dots,N\}$, we can vectorize the two-dimensional grid. Therefore, we can group observed data into a bidimensional matrix, and proceed with a usual multivariate analysis. Let $\mathbb{Y} \in \mathbb{R}^{T\times N}$ be defined as $\mathbb{Y}[t,\zeta(i,j)] = y_t(u_i,v_j)$. We hence introduce the estimated variance-covariance matrix of the just defined multivariate dataset: $\hat{\mathbf{\Gamma}}_0 \in \mathbb{R}^{N\times N}$, $\hat{\mathbf{\Gamma}}_0 = \frac{1}{T}\mathbb{Y}^T\mathbb{Y}$. Notice that $\hat{\mathbf{\Gamma}}_0[\zeta(i,j),\zeta(l,m)] = \hat{\boldsymbol{\gamma}}_0(u_i,v_j;u_l;v_m)$. Let also $\hat{\boldsymbol{\xi}} \in \mathbb{R}^N$, $\boldsymbol{\xi}[\zeta(i,j)] = \xi(u_i,v_j)$. The eigenequation can thus be rewritten in the following matricial form:

$$\omega_1\omega_2\hat{\mathbf{\Gamma}}_0\hat{\boldsymbol{\xi}} = \hat{\lambda}\hat{\boldsymbol{\xi}} \tag{60}$$

It is then straightforward to find the eigenvalues ρ and eigenvectors $\boldsymbol{\theta}$ of the matrix $\mathbf{\Gamma}_0$ and to derive $\hat{\lambda} = \omega_1\omega_2\rho$ and $\hat{\boldsymbol{\xi}} = \omega_1^{-1/2}\omega_2^{-1/2}\boldsymbol{\theta}$. Finally, to obtain an approximate eigenfunction $\hat{\xi}$ from discrete values $\hat{\boldsymbol{\xi}}$, we can use any convenient interpolation method.

B.0.2 FPCA by basis expansion

Let $\{g_i\}_{i\in\mathbb{N}}$ be a basis system for $\mathcal{L}^2([c,d])$ and $\{h_j\}_{j\in\mathbb{N}}$ a basis system for $\mathcal{L}^2([e,f])$. Consider now the tensor product basis $\{g_i \otimes h_j\}_{i,j}$, where $g_i \otimes h_j = g_ih_j$. Unfortunately, the space spanned by the tensor product basis is a proper subset of $\mathbb{H} = \mathcal{L}^2([c,d]\times[e,f])$, however, one

could also prove that such subspace is *dense* in $\mathbb{H}$, thus arguing that the tensor product basis system is sufficient to model functions of $\mathbb{H}$. Therefore, we will assume that each $x \in \mathbb{H}$, admits the decomposition:

$$x(u,v) = \sum_{i\in\mathbb{N}} \sum_{j\in\mathbb{N}} c_{i,j} g_i(u) h_j(v), \qquad c_{ij} = \langle x, g_i \otimes h_j \rangle \tag{61}$$

Thanks to the existence of a bijection[3] between $\mathbb{N}$ and $\mathbb{N}^2$ we can rearrange the terms of the basis system in order to obtain one depending on a single index instead of two, namely $\{\phi_k(u,v)\}_k$ instead of $\{g_i(u) \otimes h_j(v)\}_{i,j}$. We can thus rewrite (61) as:

$$x(u,v) = \sum_{k\in\mathbb{N}} c_k \phi_k(u,v) \tag{62}$$

In practical applications, one typically truncates the number of basis functions on the two univariate domains to K_1 and K_2 respectively, obtaining a total number of basis equal to $K := K_1 + K_2$. Let us now introduce the vector $\boldsymbol{\phi}(u,v) \in \mathbb{R}^K$, $\boldsymbol{\phi}(u,v) = [\phi_1(u,v), \ldots, \phi_K(u,v)]^T$ and the matrix $\boldsymbol{C} \in \mathbb{R}^{T\times K}$ with elements $C[t,k] = c_{tk}$ containing the coefficients of basis projection of the random functions $Y_1, \ldots, Y_T$, in such a way that:

$$Y_t(u,v) = \sum_{k=1}^{K} c_{tk} \phi_k(u,v) + \delta_t(u,v) \qquad t = \ldots, T \tag{63}$$

where $\delta_t(u,v)$ is a projection error, which is present due to the truncation of the basis system to the first K terms. In the remainder of this section, we will neglect the projection error and identify the observed functions with the ones reconstructed from the first M basis.

Following the work of Ramsay and Silverman (2005) and exploiting representation (62), we aim to rephrase the eigenproblem (56) in a matricial form. The estimated covariance

[3]The authors want to thank Edoardo Marchionni for a fruitful discussion on this topic.

function can be expressed in matrix terms:

$$\begin{aligned}\hat{\gamma}_0(u,v;w,z) &= \frac{1}{T}\sum_{t=1}^{T} Y_t(u,v)Y_t(w,z) = \\ &= \frac{1}{T}\sum_{t=1}^{T}\sum_{l,m=1}^{K} c_{il}\phi_l(u,v)c_{im}\phi_m(w,z) = \\ &= \frac{1}{T}\boldsymbol{\phi}(u,v)^T\boldsymbol{C}^T\boldsymbol{C}\boldsymbol{\phi}(w,z)\end{aligned}$$

Suppose now that an eigenfunction ξ admits the decomposition:

$$\xi(u,v) = \sum_{l=1}^{K} b_l\phi_l(u,v) + \kappa(u,v) = \tag{64}$$

$$= \boldsymbol{\phi}(u,v)^T\boldsymbol{b} + \kappa(u,v) \tag{65}$$

where $\boldsymbol{b} = [b_1,\dots,b_K]^T = [\langle\xi,\phi_1\rangle,\dots,\langle\xi,\phi_K\rangle]^T$ contains coefficients of basis projection of ξ. Neglecting once again the projection error κ, the goal becomes now to estimate the coefficients $\boldsymbol{b}$ and the corresponding eigenvalue λ for each eigenfunction ξ_j, for $j = \dots, M$. Let's introduce finally $\boldsymbol{W} \in \mathbb{R}^{K\times K}$, defined as $\boldsymbol{W} := \int_c^d\int_e^f \boldsymbol{\phi}(u,v)\boldsymbol{\phi}(u,v)^T dudv$, notice that the tensor product basis is composed by two orthonormal basis systems, the resulting tensor product basis system is itself orthonormal and thus $\boldsymbol{W} = \boldsymbol{I}$, where $\boldsymbol{I}$ denotes the diagonal matrix. The lhs of (56) can be rewritten as:

$$\hat{\Gamma}_0\hat{\xi}(u,v) = \int_c^d\int_e^f \frac{1}{T}\boldsymbol{\phi}(u,v)^T\boldsymbol{C}^T\boldsymbol{C}\boldsymbol{\phi}(w,z)\boldsymbol{\phi}(w,z)^T\hat{\boldsymbol{b}}dwdz = \tag{66}$$

$$= \frac{1}{T}\boldsymbol{\phi}(u,v)^T\boldsymbol{C}^T\boldsymbol{C}\left(\int_c^d\int_e^f \boldsymbol{\phi}(w,z)\boldsymbol{\phi}(w,z)^T dwdz\right)\hat{\boldsymbol{b}} = \tag{67}$$

$$= \frac{1}{T}\boldsymbol{\phi}(u,v)^T\boldsymbol{C}^T\boldsymbol{C}\boldsymbol{W}\hat{\boldsymbol{b}} \tag{68}$$

The eigenequation thus becomes:

$$\frac{1}{T}\boldsymbol{\phi}(u,v)^T\boldsymbol{C}^T\boldsymbol{C}\boldsymbol{W}\hat{\boldsymbol{b}} = \lambda\boldsymbol{\phi}(u,v)^T\hat{\boldsymbol{b}} \qquad \forall u,v \tag{69}$$

$$\frac{1}{T}\boldsymbol{C}^T\boldsymbol{C}\boldsymbol{W}\hat{\boldsymbol{b}} = \lambda\hat{\boldsymbol{b}} \tag{70}$$

We can hence derive the eigenvectors $\hat{\boldsymbol{b}}_j$ of $\frac{1}{T}\boldsymbol{C}^T\boldsymbol{C}\boldsymbol{W}$ and the corresponding eigenvalues and

finally reconstruct the eigenfunctions $\hat{\xi}_j$ thanks to (65).

B.1 Comparison of estimation methods

In this section, we aim to compare the two proposed approach for performing functional principal component analysis, namely the basis expansion and the discretization approach. Without loss of generality, we will settle the study in $\mathcal{L}^2([0,1]\times[0,1])$.

In general, given a finite basis system $\{\phi_k\}_{k=1,\dots,K}$, one can represent a functional time series $\{Y_t\}_{t=1}^T$ by means of representation (63), that we report here for ease of reference:

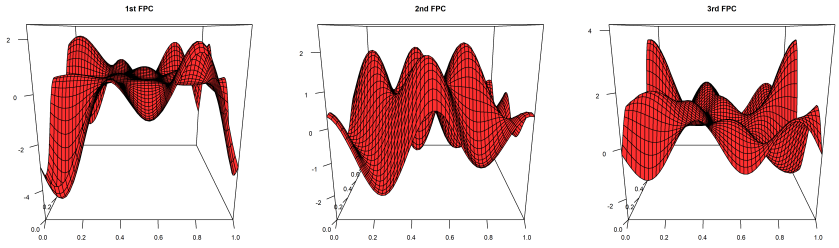
$$Y_t(u,v) = \sum_{k=1}^{K} c_{tk}\phi_k(u,v) + \delta_t(u,v) \qquad t = 1,\dots,T \tag{71}$$

Starting from this decomposition, we have derived in Appendix B.0.2 an estimator of the functional principal components ξ_j, which, however, neglects the contribution of the error δ_t. For such reason, the choice of the basis system $\{\phi_k\}_{k=1}^K$ is crucial, and when a meaningful option is not available, the approximation residual δ_t will be large and this procedure will inevitably provide biased estimates of ξ_j. On the other hand, the FPCA approach based on data discretization provides good result as long as the sampling grid is sufficiently dense.

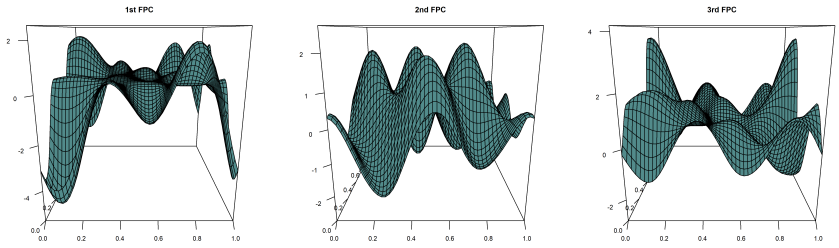
In order to prove such thesis, we simulate a time series of functions $\{Y_t\}_{t=1}^T \subset \text{span}\{\phi_1,\dots,\phi_K\}$, from a non-concurrent FAR(1) process with Gaussian errors. Data are simulated based on a basis expansion on $\{\phi_1,\dots,\phi_K\}$, which is constructed as the tensor product basis of two Fourier basis systems $\{g_i\}_{i=1,\dots,K_1}$, $\{h_i\}_{i=1,\dots,K_2}$ both defined on $[0,1]$. The sample size is chosen equal to $T = 50$, the number of basis in each of the one-dimensional systems is selected equal to 5, in order to have a total number of basis $K = 25$.

Since we know the space where the functions are embedded, we can apply the estimation procedure in Appendix B.0.2 using as basis system the same one used in the simulation. Notice that in this case the approximation error δ_t in (71) will be exactly zero and one can derive optimal estimates of the functional principal components. Estimators of the first three functional principal components are represented in Figure 15a. We repeat the same estimation procedure, this time modelling functions with a basis built from the tensor product of two one-dimensional cubic B-Spline basis systems with 5 basis each. In this case, the basis system do not coincide with the one from which functions are simulated. Nevertheless, as reported in Figure 15b all the scaled eigenfunctions are very close to the optimal ones estimated before. Finally, we compare the aforementioned estimators with

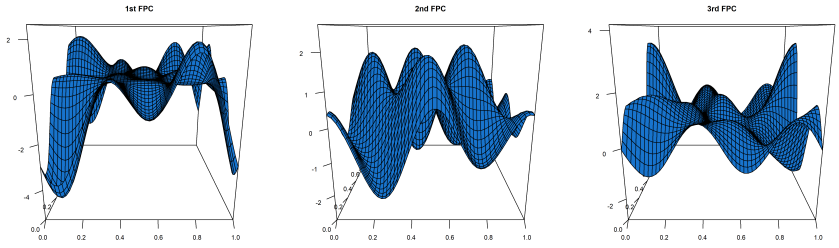
the ones coming from the discretization approach. Each of the one-dimensional grids is discretized using a step size equal to 0.02, thus resulting in a total of 2500 points. Also in this case (see Figure 15c), estimated harmonics are very close to the optimal ones in Figure 15a.



(a) FPCA by basis expansion on the same basis system used for the simulation.



(b) FPCA by basis expansion on a different basis system.



(c) FPCA by grid discretization.

Figure 15: First three functional principal components, estimated with three different methods. As usual in PCA, EFPC's are unique up to a constant. For such reason, in order to compare the different approaches, estimated harmonics in the second and third row are each rescaled by means of the mean difference ratio with the ones in the first row.

To enable for better comparison, we report in Table 3 the Mean Squared Error (MSE) (72)

between the FPC's y estimated using full knowledge of the basis system from which functions are simulated and the ones estimated using other techniques ($\hat{y}$). Such quantity is computed starting from values of y and $\hat{y}$ on a two-dimensional grid $\{(u_i, v_j)\}_{i=1,\dots,N_1; j=1,\dots,N_2}$.

$$MSE(y, \hat{y}) = \frac{1}{N_1 N_2} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} (y(u_i, v_j) - \hat{y}(u_i, v_j))^2 \tag{72}$$

We can appreciate very low values of MSE, regardless of the technique used for FPCA, thus suggesting that both the basis expansion and the discretization approach are valid options on a practical point of view.

FPCA method	EFPC's		
	1st FPC	2nd FPC	3rd FPC
Basis on B-Spline	$2.62 \cdot 10^{-3}$	$4.45 \cdot 10^{-3}$	$2.28 \cdot 10^{-3}$
Discretization	$7.69 \cdot 10^{-4}$	$2.45 \cdot 10^{-3}$	$1.38 \cdot 10^{-3}$

Table 3: MSE between estimated FPC's on the basis system used for the simulation and FPC's estimated using another basis system or the discretization approach.