Prévision non paramétrique de processus à valeurs fonctionnelles : application à la consommation d'électricité

Jairo Cugliari

UNIVERSITÉ PARIS-SUD 11

Faculté des sciences d'Orsay

N° d'ordre: ...

THÈSE

Présentée pour obtenir

LE GRADE DE DOCTEUR EN SCIENCES
DE L'UNIVERSITÉ PARIS-SUD XI

Spécialité: Mathématiques

par

Jairo CUGLIARI

Prévision non paramétrique de processus à valeurs fonctionnelles.

Application à la consommation d'électricité.

Soutenue le 22 Novembre 2011 devant la Commission d'examen:

M.	André MAS	
M.	Guy NASON	
M.	Georges OPPENHEIM	
M.	Pascal MASSART	
M.	Xavier BROSSAT	
M.	Anestis ANTONIADIS	(Directeur de thèse)
M.	Jean-Michel POGGI	(Directeur de thèse)

Rapporteurs:

M.	André MAS
M.	Guy NASON

Thèse préparée au
Département de Mathématiques d'Orsay
Laboratoire de Mathématiques (UMR 8628), Bât. 425
Université Paris-Sud 11
91 405 Orsay CEDEX

Résumé

Nous traitons dans cette thèse le problème de la prédiction d'un processus stochastique à valeurs fonctionnelles. Nous commençons par étudier le modèle proposé par Antoniadis et al. (2006) dans le cadre d'une application pratique -la demande d'énergie électrique en France- où l'hypothèse de stationnarité semble ne pas se vérifier. L'écart du cadre stationnaire est double : d'une part, le niveau moyen de la série semble changer dans le temps, d'autre part il existe groupes dans les données qui peuvent être vus comme des classes de stationnarité.

Nous explorons corrections qui améliorent la performance de prédiction. Les corrections visent à prendre en compte la présence de ces caractéristiques non stationnaires. En particulier, pour traiter l'existence de groupes, nous avons contraint le modèle de prévision à n'utiliser que les données qui appartiennet au même groupe que celui de la dernière observation disponible. Si le regroupement est connu, un simple postraitement suffit pour obtenir des meilleures performances de prédiction.

Si le regroupement en blocs est inconnu, nous proposons de découvrir le regroupement en utilisant des algorithmes d'analyse de classification non supervisée. La dimension infinie des trajectoires, pas nécessairement stationnaires, doit être prise en compte par l'algorithme. Nous proposons deux stratégies pour ce faire, toutes les deux basées sur les transformées en ondelettes. La première se base dans l'extraction d'attributs associés à la transformée en ondelettes discrète. L'extraction est suivie par une sélection des caractéristiques le plus significatives pour l'algorithme de classification. La seconde stratégie classifie directement les trajectoires à l'aide d'une mesure de dissimilarité sur les spectres en ondelettes.

La troisième partie de la thèse est consacrée à explorer un modèle de prédiction alternatif qui intègre de l'information exogène. A cet effet, nous utilisons le cadre des processus Autorégressifs Hilbertiens. Nous proposons une nouvelle classe de processus que nous appelons processus Conditionnels Autorégressifs Hilbertiens (CARH). Nous développons l'équivalent des estimateurs par projection et par résolvant pour prédire de tels processus.

Mots-clefs : Processus autorégressifs hilbertiens, Données fonctionnelles, Ondelettes, Prévision non paramétrique, Consommation d'électricité.

Non parametric forecasting of functional-valued processes. Application to the electricity load.

Abstract

This thesis addresses the problem of predicting a functional valued stochastic process. We first explore the model proposed by Antoniadis et al. (2006) in the context of a practical application -the french electrical power demand- where the hypothesis of stationarity may fail. The departure from stationarity is twofold: an evolving mean level and the existence of groups that may be seen as classes of stationarity.

We explore some corrections that enhance the prediction performance. The corrections aim to take into account the presence of these nonstationary features. In particular, to handle the existence of groups, we constraint the model to use only the data that belongs to the same group of the last available data. If one knows the grouping, a simple post-treatment suffices to obtain better prediction performances.

If the grouping is unknown, we propose it from data using clustering analysis. The infinite dimension of the not necessarily stationary trajectories have to be taken into account by the clustering algorithm. We propose two strategies for this, both based on wavelet transforms. The first one uses a feature extraction approach through the Discrete Wavelet Transform combined with a feature selection algorithm to select the significant features to be used in a classical clustering algorithm. The second approach clusters directly the functions by means of a dissimilarity measure of the Continuous Wavelet spectra.

The third part of thesis is dedicated to explore an alternative prediction model that incorporates exogenous information. For this purpose we use the framework given by the Autoregressive Hilbertian processes. We propose a new class of processes that we call Conditional Autoregressive Hilbertian (carh) and develop the equivalent of projection and resolvent classes of estimators to predict such processes.

Keywords : Autoregressive hilbertian process, Functional data, Wavelets, Nonparametric forecasting, Electricity consumption.

Remerciements

Il y a quatre ans, je débarquais dans un nouveau pays, complètement inconnu. Quelques mois plus tard une nouvelle aventure commençait : faire une thèse. Je n'aurais pas pu tenir la route si ce n'était grâce au support de beaucoup de personnes.

Tout d'abord, je voudrais remercier Badih Ghattas. Ton support a été sans égal pour m'aider à faire mes premier pas en France, m'adapter à une nouvelle langue et me soutenir pour faire une thèse.

Je tiens bien entendu à exprimer toute ma gratitude à Anestis Antoniadis et Jean-Michel Poggi pour m'avoir fait confiance dans cette thèse. Depuis le tout début, vous m'avez guidé et encouragé à trouver mon propre chemns. Vous avez toujours mis en valeur mon travail, surtout dans mes moments de doute.

Je suis très reconnaissant envers André Mas et Guy Nason qui ont accepté de rapporter cette thèse. Je remercie aussi Pascal Massart qui m'a fait l'honneur de participer à mon jury de thèse. Mes remerciements vont également à Georges Oppenheim, pour la gentillesse et la patience qu'il a manifesté à mon égard durant cette thèse, pour tous les conseils, et aussi pour m'avoir fait l'honneur de participer au jury de soutenance.

Je remercie chaleureusement Xavier Brossat pour la qualité de son encadrement au sein d'EDF. Merci pour tout ce moments enrichissants, professionnellement et humainement.

Je voudrais remercier aussi les membres de l'équipe de prévision à OSIRIS, EDF R&D. Merci pour votre patience et votre accueil. J'ai beaucoup apprécié les discussions aux cafés et les pique-niques. Un grand merci à Virginie et Tristan avec qui j'ai successivement partagé le bureau des thésards. Au risque de paraître ditirimbique, j'ai bigrement apprécié votre humour, votre écoute et votre bonne humeur. Un grand merci aussi à Amandine, Aurelie, Nicolas et au jeune docteur Goude pour ces verres après le boulot.

Durant ces années de thèse j'ai pu côtoyer de nombreux doctorants à Orsay et à Grenoble. J'ai voudrais remercier particulièrement Cyprien, Dominique, Hayat, Jean-Patrick, Pierre, Robin et Sebastien pour tout les moments de détente.

Je tiens à remercier ma famille et mes amis. A ma mère en premier lieu, pour sa résilience face à l'adversité et à mon père pour son soutien inconditionnel. J'ai toujours senti que mes amis étaient présents, malgré la distance. Un grand merci pour m'avoir poussé quand je n'avais pas le courage, pour m'avoir freiné quand je croyais être le roi du monde.

Enfin, je ne pourrais pas finir cette page sans avoir une petite pensée pour ma Rose.

Table des matières

Introduction.

The final goal of this thesis is to make predictions about a continuous time process. For this purpose, let us consider a continuous-time univariate stochastic process $X = (X(t), t \in \mathbb{R})$ which is observed over the interval $[0, T]$, $T > 0$ at a relatively high sampling frequency. We are interested on the prediction problem, i.e. we want to say something about the future behaviour of X. It is well known that the best probabilistic predictor (in the least mean square error sense) is the conditional expectation of the future of the process given the past. Still, it is in general unknown so the associated regression function must be estimated. To treat the regression problem one may use parametric and nonparametric models.

Another choice to make is between point and interval prediction. In the first case, one wants to pointwise predict X_{t+s}, the value of X at some time point $t + s$, $s > 0$ (see Bosq (1996, Ch. 5)). In the second case, one is interested in the behaviour of X over a whole interval $[T, T + \delta]$, $\delta > 0$. We will study this second situation. In practice, these processes are usually observed only through a discrete sampling grid and possible with some additional observational error. Still, it may be useful to consider the underlying sample paths as realizations of random functions, i.e. random variables taking values on a functional space. To do this, Bosq (1991) constructs from X another process $Z = (Z_i, i \in \mathbb{N})$ by dividing the interval $[0, T]$ into sub-intervals $[(l-1)\delta, l\delta]$, $l = 1, \ldots, n$ with $\delta = T/n$ (see Figure 1). Then, the functional-valued discrete time stochastic process Z is defined by

$$Z_i(t) = X(t + (i-1)\delta), \qquad i \in \mathbb{N}, \qquad \forall t \in [0, \delta).$$

The random functions Z_i thus obtained, while exhibiting a possibly nonstationary

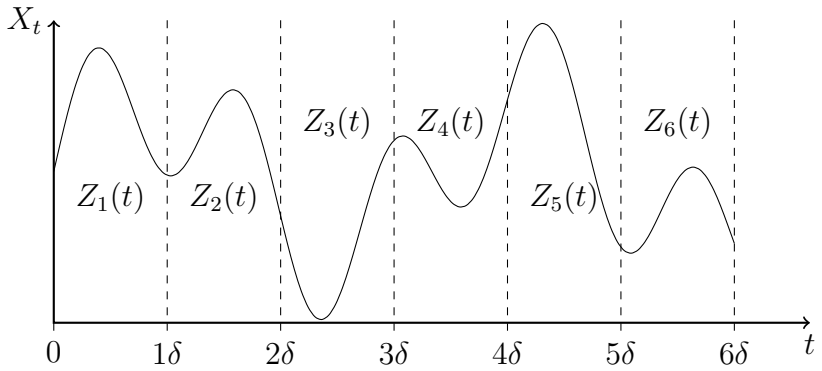


FIGURE 1 – Representation of a continuous time process as a functional time series.

behaviour within each continuous time subinterval, form a functional discrete time series that is usually assumed to be stationary. Such a procedure allows to handle seasonal variation of size δ on X in a natural way. Note that although the mentioned construction is particularly fruitful when X presents a seasonal component, one may obtain a functional time series from more general constructions, including adjacent, disjoint or even overlapping segments.

The original prediction problem is now casted as the prediction of the element Z_{n+1} using a sequence of functions $Z_1, \ldots, Z_n$. The mathematical objects to deal with impose

the adaptation of classical statistical prediction methods to infinite dimensional random variables. The use of function-valued random variables has received increased attention in the past two decades. The branch of statistics dealing with them has been named *Functional Data Analysis* (FDA) in the seminal paper of Ramsay and Dalzell (1991). For the case of independent and identical distributed data, Ramsay and Silverman (1997, 2002) give a detailed introduction on both theoretical and practical aspects. For dependent functional data, Bosq (2000) proposes and studies linear processes.

1 Prediction of functional time series : state of the art.

1.1 Autoregressive Hilbertian processes.

Bosq (1991) introduces the Autoregressive Hilbertian process of order 1 (ARH(1)) to study the prediction problem on functional data. Let H be a separable Hilbert space. A discrete time H valued stochastic process Z is an ARH if it is stationary and for each $n \in \mathbb{Z}$

$$Z_{n+1} - \mu = \rho(Z_{n-1} - \mu) + \epsilon_n, \tag{1.1}$$

with $\mu \in H$ the expectation of the process, ρ a bounded linear operator over H and $\epsilon = (\epsilon_n, n \in \mathbb{Z})$ a strong H-valued white noise (i.e. a sequence of independent and identical distributed random variables on H such that $\mathbb{E}[\epsilon_0] = 0$ and $\mathbb{E}\|\epsilon_n\|_H^2 < \infty$). An extensive study of such processes can be found in Bosq (2000) which can be completed with more recent results from Mas and Pumo (2011).

For such process, the best predictor of Z_{n+1} given the past observations is $\tilde{Z}_{n+1} = \rho Z_n$. Notice that ρ is unknown. Its estimation illustrates some of the challenges FDA can present. Indeed, the estimation is based on the following Yule-Walker like relation,

$$\Delta = \rho\Gamma, \tag{1.2}$$

where $\Gamma = \mathbb{E}[Z_0 \otimes Z_0]$ is the covariance operator of Z and $\Delta = \mathbb{E}[Z_0 \otimes Z_1]$ is the cross-covariance operator of order 1. Note that higher order cross-covariance operators are all null. Mimicking what is done with matrices, one is tempted to use the inverse of Γ to obtain a solution for ρ. However, the inverse of *Gamma* is not defined over the whole space. Moreover, although one can define a dense domain $\mathcal{D}_{\Gamma^{-1}}$ for it, the operator Γ^{-1} is continuous at no point on this domain (Mas (2000)). Note that even if one can do it in practice, because the empirical counterpart of these operators have finite rank, no asymptotic results can be derived. The problem can be circumvented, since the adjoint of a linear operator in H with a dense domain is closed (*closed graph theorem*, see for example Kato (1976, Theorem 5.20)) and since the range of the adjoint of the cross-covariance operator, Δ^*, is in $\mathcal{D}_{\Gamma^{-1}}$. From these fact and from (1.2) one has

$$\rho^* = \Gamma^{-1}\Delta^*.$$

By this way, all theoretical results concerning the estimation of ρ^* are also valid for the estimation of ρ.

Mas (2000) identifies two classes of estimators for ρ^*. The first one, the class of *projection estimators*, projects the data onto an appropriate subspace H_{k_n} of H of finite dimension k_n. Let Π_{k_n} be the projector operator over H_{k_n}, and call Γ_n and Δ_n the empirical counterpart of Γ and Δ respectively. Then one inverts the linear operator defined by the matrix $\Pi_{k_n}\Gamma_n\Pi_{kn}$ and completes with the null operator on the orthogonal subspace. In Bosq (2000), H_{k_n} is set equal to the space generated by the first k_n eigenfunctions, say $e_1, \ldots, e_{k_n}$, of Γ. The subspace H_{k_n} is estimated by $\widehat{H}_{k_n}$, the linear span of the first k_n empirical eigenfunctions. By this way, if P_{k_n} is the projection operator on $\widehat{H}_{k_n}$, the estimator of ρ^* can be written as

$$\rho_n^* = (P_{k_n}\Gamma_n P_{k_n})^{-1}\Delta_n^* P_{k_n}. \tag{1.3}$$

The estimation solution by projection is equivalent to approximate Γ^{-1} by a linear operator with additional regularity $\Gamma^\dagger$ defined as

$$\Gamma^\dagger = \sum_{j=1}^{k_n} b(\lambda_j)(e_j \otimes e_j),$$

where $(k_n)_n$ is an increasing sequence of integers tending to infinity and b is some smooth function converging point-wise to $x \mapsto 1/x$. Indeed, $\Gamma^\dagger \to \Gamma^{-1}$ when $k_n \to \infty$. The choice of taking $b(x) = 1/x$ yields, for a finite k_n, to set $\Gamma^\dagger$ equal to a spectral cut of Γ^{-1}. However, this choice is not unique. Mas (2000) consider a family of functions $b_{n,p} : \mathbb{R}_+ \mapsto \mathbb{R}_+$ with $p \in \mathbb{N}$ such that

$$b_{n,p}(x) = \frac{x^p}{(x+\alpha_n)^{p+1}},$$

with α_n a strictly positive sequence that tends to 0. With this, the second class of estimators for ρ^*, the *resolvent class*, is defined as

$$\rho_{n,p}^* = b_{n,p}(\Gamma_n)\Delta_n^*, \tag{1.4}$$

where we write $b_{n,p}(\Gamma_n) = (\Gamma_n + \alpha_n I)^{-(p+1)}$ with $p \geq 0$, $\alpha_n \geq 0$, $n \geq 0$. As for the projection class, the operators $b_{n,p}(\Gamma_n)$ from the resolvent class can be associated to an regularized approximation of Γ^{-1} to solve the inversion problem (see Antoniadis and Sapatinas (2003) for a discussion on this topic applied to the ARH estimation).

Both classes of estimators allow one to predict the future value Z_{n+1} from a sequence $(Z_1, \ldots, Z_n)$ by first estimating the autocorrelation operator ρ^* using $(Z_1, \ldots, Z_{n-1})$, and then applying it to the last available observation Z_n. Alternatively, one may directly predict Z_{n+1} by estimating the relevant elements of the range of ρ^*. Using a basis of the space, one may decompose ρZ_n and use the adjoint property. This second strategy is proposed in Antoniadis and Sapatinas (2003). Using wavelet basis the authors obtain considerable better prediction errors. Kargin and Onatski (2008) go further in this sense and proposed to use a basis adapted to the prediction task.

Additionally, for prediction purposes, to the good theoretical properties of the estimators, the interest of ARH processes has been also be shown in practice in several situations : traffic prediction (Besse and Cardot (1996)) cash flow and credit transcations (Laukatis (2008)), climatic variation (Besse et al. (2000)) or electricity demand Andersson and Lillestol (2010) for example.

1.2 Nonlinear autoregressive functional process.

As for univariate processes, ARH processes linearly connect future and present observations. In some cases, this kind of relationship may be too restrictive. A more general class of univariate processes $\{Y_i\}_{i\in\mathbb{Z}}$ consists on making each variable Y_n depend on some nonparametric function m of lagged variables $\mathcal{Y}_{n,d} = (Y_{n-1}, \ldots, Y_{n-d+1})$ for some integer $d > 1$. The resulting model is then

$$Y_{n+1} = m(y) + \epsilon_n,$$

where the regression function $m(y) = \mathbb{E}[Y_{n+1}|\mathcal{Y}_{n,d} = y]$ and ϵ_n is some noise process. If one disposes of an estimation of m, then Y_{n+1} is predicted by plugging in of the last observed segment for $\mathcal{Y}_{n,d}$.

The estimation of m can be obtained by means of nonparametric methods which is sometime more desirable than parametric methods. First, parametric models impose a kind of mechanical effect that may be too restrictive. Second, nonparametric methods are robust. Last, sometimes nonstationary patterns like trend and seasonality can be naturally exploited in the prediction. A popular choice between nonparametric estimation is to estimate the regression function m using kernel smoothing. Then, the estimation can be intuitively interpreted by a simple idea : similar futures correspond to similar pasts. In the same sense, the corresponding predictor can be seen as a weighted mean of the futures of past blocs where larger weights correspond to past trajectories that are more similar to the present one (see for instance Poggi (1994)).

In Antoniadis et al. (2006), the authors mimic this reasoning for functional valued processes. They propose to estimate the regression $\mathbb{E}[Z_{n+1}|Z_n]$ by means of a kernel regression estimator adapted to work with functional-valued variables. The prediction methodology, that we will abbreviate KWF, consists of two phases.

First, one searches for curves similar to the present one Z_n and constructs a vector of weights $w_{n,i}, i = 1, \ldots, n-1$. Then, the future values of these curves are used to obtain the prediction

$$\widehat{Z}_{n+1} = \sum_{i=1}^{n-1} w_{n,i} Z_{i+1}.$$

As before, the resulting predictor can be seen as a weighted mean of futures of past curves. The use of an appropriate similarity measure is the key element of the methodology. It should be capable to deal with the functional nature of the curves. Moreover, as mentioned before, the introductory construction of a functional time series Z, one only need to assume the stationarity of Z. The behaviour within each curve $Z_i(t), t \in [0, 1]$ may be nonstationary and the similarity measure should take this into account.

Both issues are treated using the wavelet transform in order to approximate the underlying sample paths. Wavelets are known to have very good approximation properties even for quite rough trajectories (see for example Vidakovic (1999)). Additionally, wavelets appear as a successful tool to capture local features from the observed values of the sampling of curves. This is exploited in the construction an appropriate dissimilarity D between two curves. Indeed, one first writes the wavelet approximation truncated at

some level J of the observation $Z_i(t), t \in [0,1]$,

$$Z_{i,J}(t) = \sum_{k=0}^{2^{j_0}-1} c_{j_0,k}^{(i)} \phi_{j_0,k}(t) + \sum_{j=j_0+1}^{J} \sum_{k=0}^{2^j-1} d_{j,k}^{(i)} \psi_{j,k}(t),$$

where $c_{j_0,k}^{(i)}$ and $d_{j,k}^{(i)}$ are respectively the scale and wavelet coefficients for the scale j and the position k. Then, the Euclidean distance is used between the detail coefficients of the observations Z_i and $Z_{i'}$ at each scale j,

$$\mathrm{dist}_j(Z_i, Z_{i'}) = \left(\sum_{k=0}^{2^j-1} (d_{j,k}^{(i)} - d_{j,k}^{(i')})^2 \right)^{1/2}.$$

Finally, all the scale dependent distances are aggregated in one measure D of the dissimilarity between the curves Z_i and $Z_{i'}$, taking into account the different number of coefficients by scale

$$D(Z_i, Z_{i'}) = \sum_{j \geq j_0+1} 2^{-j/2} \mathrm{dist}_j(Z_i, Z_{i'}).$$

As for the scalar case, a vector of weights is derived via a kernel probability function that uses this dissimilarity measure.

We have discussed how wavelets are used on the first step of the prediction methodology, that is the construction of the weight vector. Furthermore, they also play an important role on the second step. Indeed, the resulting kernel regression predictor is used on the scaling coefficients of the wavelet transform to obtain the scaling coefficients of the prediction of Z_{n+1}. This allows one to make assumptions on the distribution of the scaling coefficients instead of making them directly on the observed discrete samplings. To end, one uses the inverse of the transform to obtain the predicted trajectory of Z_{n+1} approximated at some resolution level.

Finally, let us mention that the good theoretical behaviour of the predictor has been shown, as well as its nice performance on prediction over a simulated and some real-life data examples. Bootstrap based confidence interval for each time point of the predictor are also available.

2 Prediction of nonstationary nonlinear autoregressive functional processes.

The starting point of this thesis is the practical study of the implementation of KWF as a prediction model for the French national electricity demand. We concentrate ourselves on daily forecasts. In this context, each daily load curve is represented by a sampling regular grid of 48 time points.

Some stylized fact of this data should be mentioned. First, the electricity demand is highly dependent on meteorological conditions, which is translated by a seasonal cycle. Second, social and economic phenomena (like holidays or the dichotomy between working-days and weekends) are present in the electricity demand. Last, the structure of the weeks forms a weekly cycle where the profiles vary not only in their mean level but

also in their shape. It is also usual to found intraweekly differences, usually on the days next to weekends.

The analysis of the prediction error distribution along the year reveals that the departures from stationarity of the data seriously hampers the performance of KWF prediction model. Two facts affecting the prediction performance are well identified. The first is associated to the evolution of the mean level driven by a long term trend, the annual cycle and short term fluctuations produced by meteorological variations. The second fact is the existence of groups of daily curves with very different shapes like, for instance, bank or public holiday. We explore and propose solutions for both of these problems.

2.1 Correction of an evolving mean level.

The two phases of the prediction model deal differently the mean level of the curves. Since we use the wavelet approximation, the mean level is associated to the approximation part represented by the scaling coefficients. We write the approximation and detail part by $\mathcal{S}_i(t)$ and $\mathcal{D}_i(t)$ respectively, then we have

$$Z_i(t) = \underbrace{\sum_k c_{j_0,k}^{(i)} \phi_{j_0,k}(t)}_{\mathcal{S}_i(t)} + \underbrace{\sum_{j \geq j_0} \sum_k d_{j,k}^{(i)} \psi_{j,k}(t)}_{\mathcal{D}_i(t)} .$$

The original methodology implicitly assumes that the sequence of approximation parts does not present significant variations. We work at the coarser resolution, that is $j_0 = 0$. In this case, the approximations are constant and proportional to the mean level of each curve. Moreover, we can treat the sequence $\{c_{j_0,k}^{(i)}\} = \{c_{j_0}^{(i)}\}_{i=1,\ldots,n}$ as an univariate time series. In order to obtain a prediction of the future curve $n+1$, we write

$$\widehat{Z_{n+1}(t)} = \widehat{\mathcal{S}_{n+1}(t)} + \widehat{\mathcal{D}_{n+1}(t)}.$$

The detail part is predicted used the KWF model. For the approximation part, we explore the following variants :

BASE Use the same weight vector to predict

$$\widehat{\mathcal{S}_{n+1}(t)} = \sum_{m=1}^{n-1} w_{m,n} \mathcal{S}_{m+1}(t)$$

.

PRST A simple persistence model

$$\widehat{\mathcal{S}_{n+1}(t)} = \mathcal{S}_n(t)$$

.

DIFF Predict the increment of the mean level using the weight vector

$$\widehat{\mathcal{S}_{n+1}(t)} = \mathcal{S}_n(t) + \sum_{m=1}^{n-1} w_{m,n} \Delta(\mathcal{S}_n)(t)$$

SAR Exploiting the fact that we dispose with an univariate time series, classical Box-Jenkins modelling is used to predict next value of $c_{j_0}^{(n+1)}$.

The use of any of these variants yields on a net improvement of the prediction error. However, it is the DIFF variant the one that presents the better behaviour. Note that this variant explicitly incorporates the transition between two curves.

2.2 Stationarity within a class.

As already noticed above, for many functional data the segmentation into subintervals of length δ may not suffice to make reasonable the stationary hypothesis of the resulting segments. For instance, in modeling the electrical power demand process the seasonal effect of temperature and the calendar configuration strongly affects the shape of the daily load demand profile. *Recognizing this, our aim is therefore to propose a clustering technique that clusters the functional valued (discrete) times series segments into groups that may be considered as stationary so that in each group more or less standard functional prediction procedures such as the ones cited above can be applied.*

Load profiling and forecasting are associated in many references. Let us comment two references connecting electricity load consumption, clustering and forecasting. In Piao et al. (2008), individual load curve encoding is performed using features of the daily load curve shape. The representative load profiles are then obtained by clustering the consumer's load patterns using k-means. Finally, classification methods are used to predict the customer load pattern. In Goia et al. (2010), the problem of short-term peak load forecasting for each customer is addressed. It combines the functional clustering procedure of Abraham et al. (2003) to classify the daily load curves and then, according to the obtained groups, a family of functional linear regression models are defined for the forecasting step.

We propose two strategies to cluster functional data taking into account the fact that the curves may present nonstationary patterns. The first strategy is based on the *feature extraction* of a number of handy features from the infinite dimensional sample paths. Wavelets are used to both approximate the sample paths, and to compute what we call the *absolute energy contributions* of the scales of the transform. Although these quantities can be interpreted in terms of the associate cycle of the time series for each scale, nothing warranties that they are useful to detect the cluster structure. As for regression analysis, one can use feature selection algorithms to find the subset of feature that significantly explains the cluster structure. Finally, the selected features and an appropriate estimation of the number of clusters are used as inputs on unsupervised learning algorithms to estimate the clusters.

The second strategy directly uses a *dissimilarity between curves*. The success of any clustering algorithm depends on the adopted dissimilarity measure. Direct similarity measures such as L_p norms match two functional objects in their original representations without explicit feature extraction. When $p = 2$, this reduces to commonly used Euclidean distance. L_p norms are straightforward and easy to compute. However, in many cases such as in shifting and scaling, the distance of two sequences cannot reflect the desired (dis)similarity between them. We propose a dissimilarity based on the *wavelet coherence* to direct asses the relationships between time series at specific scales.

The *coherence* is used to determine the association between two square-integrable signals z and x. In Fourier analysis, the coherence function is a direct measure of the correlation between the spectra of two time-series (Chatfield (1989)). To quantify the relationships between two non-stationary signals, the *wavelet coherence* is a generalization to signal whose frequency content changes with time. We adapt the extended multiple coefficient of determination to address scale-specific questions in non stationary time series data through the wavelet coherency. In the same way that the coefficient of

determination is associated to the Euclidean distance of the standardize version of two vectors, the wavelet extended coefficient of determination is associated to our proposed dissimilarity. Again, more or less classical clustering algorithms are used to estimate the clusters.

The results of using these two strategies are different. While the first one has the advantage of being very fast and interpretable (thanks to the feature selection), the second seems to produce classes formed by more homogenous shapes.

In our context, the clustering is closely related to forecasting objectives in order to render more reasonable the stationarity assumption of the underlying functional process. The clustering output should be incorporated in the prediction model. As mentioned above, the transitions between two consecutive observations play a crucial role in the prediction performance. In order to incorporate them in the model, we may range chronologically the estimated cluster memberships. Then, the transition probabilities of the process associated to the membership trajectory can be estimated. Then, predictions of the future function produced with the KWF model for each of the future classes would be weighted by the associated probability.

While the idea of cluster functions was guided by the need of render more reasonable the hypothesis of stationary, the clustering may also be used to incorporate exogenous information. For instance, the temperature daily curves may be used as the input of the clustering in order to obtain temperature classes. In the next section we study another way of incorporating exogenous information in a different context of functional prediction.

3 Introducing exogenous variables into the functional predictor.

When observing functional process one may also count with additional information that may want to include in the predictor. For example, Damon and Guillas (2002) introduce functional-valued regressors on ARH processes to obtain the ARHX model. The proposed model is used to forecast the air quality using meteorological variables as functional regressors. Both the variable and the covariate are assumed to follow ARH processes. The same authors propose a `R` package to simulate, estimate and predict ARH and ARHX processes (see Damon and Guillas (2005)). In Mas and Pumo (2007), the derivative of the actual function is included as covariate is the ARH framework. These variants include additive factors to the autoregression equation. A different angle is adopted by Guillas (2002) who proposes to model an ARH process that randomly chooses an autocorrelation operator from two possible values. The state of this operator is taken into account by introducing a Bernoulli variable at each drawn. By this way, the inherent linearity of the ARH process is avoided. However, the dichotomy of the covariate (which may be extended to multiple classes) seems to be insufficient to predict efficiently some real life problems.

For the electricity demand we will use the classes issued from the functional clustering to create transition classes between days. By this way, the use of an autocorrelation operator by class rends the prediction model more flexible. Moreover, classes can be used on the nonlinear autoregressive model mentioned before : one first chooses a class of

day, then within this class the prediction is performed following the chosen functional prediction model. This prediction strategy can be seen as if one conditioned the functional valued process Z by a sequence of covariates $V = (V_n, n \in \mathbb{Z})$. The covariate V may be discrete, as for the groups of stationarity, or more general. So, after conditioning by it we will get back to original stationary framework. In the ARH model, conditioning by V would give a model such that the predictor of Z_{n+1} would be

$$\widetilde{Z_{n+1}} = \mathbb{E}(Z_{n+1}|Z_n, \dots, Z_1, V = v)$$

if we have observed the $n-$th segment together with the value v for V. We propose to make the autocorrelation operator depend on the sequence of covariates V that we will assume to have a continuous distribution on $\mathbb{R}^d$.

The conditional expectation is characterized by the conditional distribution of Z given V, i.e. by the conditional probability $\mathbb{P}_{Z|V}$ on $\mathcal{B}_H$. In order this conditional probability be properly defined as a measure (in the sense that it represents a regular version of the conditional probability), it is assumed that a transition probability exists that associates to each v a probability measure $\mathbb{P}^v$ on $(H, \mathcal{B}_H)$ such that

$$\mathbb{P}^v_{Z|V}(A) = \mathbb{P}^v(A), \qquad \text{for every } A \in \mathcal{B}_H.$$

Then, we can express the resulting process, that we call *Conditional Autoregressive Hilbertian* (CARH), as

$$Z_k - a = \rho_{V_k}(Z_{k-1} - a) + \epsilon_k, \qquad k \in \mathbb{Z}, \tag{3.1}$$

where a is the conditional (on V) expectation of the process Z, $a(v) = \mathbb{E}[a|V]$. Note that conditionally on V, Z is an ARH process. Under some mild conditions, Equation (3.1) has one unique stationary solution. By this way, the model approaches the time-varying regression framework that can be find in literature (see Hastie and Tibshirani (1993) and Wu et al. (2010) for the scalar and functional cases respectively).

We mimic the estimation and prediction strategies developed on the ARH framework. First we obtain the conditional versions of the covariance and cross-covariance operators, namely

$$\begin{aligned} z \in H \mapsto \Gamma_v z &= \mathbb{E}^v[(Z_0 - a) \otimes (Z_0 - a)(z)|V] \qquad \text{and} \\ z \in H \mapsto \Delta_v z &= \mathbb{E}^v[(Z_0 - a) \otimes (Z_1 - a)(z)|V]. \end{aligned}$$

Second, estimators of a, Γ_v and Δ_v are proposed using Nadaraya-Watson like kernel estimators in the case of $H = L_2([0,1])$. Here, the covariance operators are written as integral operators with their respective kernels $\gamma(v,s,t)$ and $\delta(v,s,t)$ with $(s,t) \in L_2([0,1]^2)$ and $v \in \mathbb{R}^d$. Therefore, we estimate the kernels to obtain the estimators $\widehat{\Gamma}_{v,n}$ and $\widehat{\Delta}_{v,n}$.

Last, we obtain the equivalent of the projection and resolvent predictor classes for CARH processes, given respectively by

$$\begin{aligned} \widehat{\rho}^*_{v,n} &= (P_v^{k_n} \widehat{\Gamma}_{v,n} P_v^{k_n})^{-1} \widehat{\Delta}^*_{v,n} P_v^{k_n} \\ \widetilde{\rho}^*_{v,n} &= b_{n,p}(\widehat{\Gamma}_{v,n}) \widehat{\Delta}^*_{v,n}, \end{aligned}$$

where $P_v^{k_n}$ is the projector operator over a subspace of dimension k_n spanned, for example the one spanned by the first k_n eigen directions of the conditional covariance operator Γ_v. For this, we follow Cardot (2007) to obtain the direction of principal variance of Z, taking into account both the functional nature of the curves and the fact that they are generated conditionally to some covariate. Our procedure is an extension to dependent data of his proposal.

4 Practical application : French national electricity demand

The practical motivation of the present thesis is the problem that faces a producer of electricity when it has to predict the demand of his clients. Electricity is a specific commodity, namely because it can not be stored, so the production must equals the demand at every moment. Otherwise, the electrical grid used in the distribution of the production may be damaged and/or blackouts may occur.

Prediction of electricity consumption is needed at several horizons. Larger ones helps to anticipate the needs of the production and distribution means. The shorter ones are used to decide the production and distribution plans (production mix, location, etc.). With reliable predictions, the production is obtained at lesser cost.

Whilst the liberalization of the electricity markets may conduce to a more favourable situation for customers, it clearly increases the complexity of the prediction problem (Weron (2006)). More suppliers in the market, and for each one the clients perimeter may vary at any time. The market is the more reactive to changes possibly producing strategies that change dynamically with faster or slower reactions to changes on the market price level.

The recent development of electrical smart grids communicating with (smart) meters adds an extra layer of complexity to the market. Now customers have more favourable conditions to dynamically adjusts its consumption strategies with respect to market signals. Last but no least, the sources of energy used in the electricity generation -and the dependence of some of this sources to meteorological conditions (e.g. solar or eolian)- throws an even more messy landscape.

In view of these facts, how does existent forecasting methods perform in the new competitive universe? We will use data provided by EDF (*Électricité de France*) focusing ourselves on the short term prediction, where here short means one day. The methods that were historically used to predict the daily load demand rely on the analysis of time series sampled at discrete time. For instance, the sampling rate of the electrical power demand used on the prediction is 30 minutes. This may lead to a poor representation of some features of the daily load curve, *e.g.* the daily load peak may not be located at a 30 minute point. Moreover, the discrete time sampling is not able to cope with the underlying temporal continuity of the electricity demand signal.

4.1 Brief literature review of load curve forecasting.

We give here some references that help to situate our work in the available literature. More references with details can be found in Weron (2006). Two articles in press give numerous references : Cancelo et al. (2008); Taylor (2010).

The different models treating the short-term forecast of load curves can be classed in four groups, namely :

1. time series analysis,
2. machine learning (mainly Artificial Neural Networks but also Support Vector Machines),
3. similarity search models,
4. regression analysis.

Each of these classes of models may be considered for univariate (half-hourly our hourly sampled) or multivariate (on 24 or 48 dimensions) processes.

Time series analysis methods like SARIMA models combine past values of the series in a parametric form to obtain a prediction. When predicting electricity demand, the parametric form induces a mechanical effect on these models producing too large prediction errors specially on periods of high volatility such as winter or special days (see for example Nowicka-Zagrajek and Weron (2001) which implements an ARMA variant using hyperbolic noise for the error structure). In Dordonnat (2009) a state-space model is proposed, allowing the parameters to vary on time. To overcome with the linearity of some parametric forms, one may use *similarity search* models usually associated with kernel regression methods Poggi (1994); Antoniadis et al. (2006).

In order to incorporate exogenous information, one can use *regression models* to describe the electricity demand in terms of the felt temperature or other climatological variables (Bruhns et al. (2005); Cancelo et al. (2008); Soares and Medeiros (2008)). For instance, the operational models at EDF use this strategy to take into account the felt temperature inside buildings through a nonlinear model.

The *machine learning* approaches differ considerably from the precedent ones. The prediction performance of these methods is not clear for the electricity demand. For instance, different studies using artificial neural networks throw conclusions that are sometime contradictory (Weron (2006)). However, the use of online mixing algorithms for bunch of predictors have been shown to be useful when the context of the prediction changes (Goude (2008)).

Let us mention some references dealing with prediction of curves instead of time points. Antoch et al. (2008) and Andersson and Lillestol (2010) test an autoregressive model over daily load curves. They conclude that this modelisation strategy seems to be competitive to other more classical ones. Both agree to say that a more thoughtful study should be performed, in particular to deal with special days events. In Aneiros et al. (2011) a semi-functional partial linear model is developed, incorporating additive exogenous variables on the functional autoregression of the daily electricity demand process.

A general remark on all the mentioned references is the need of a different treatment for special days. Usually this is an *adhoc* treatment performed directly by the operator

of the model and based on his experience.

Plan of the thesis.

This document is divided in three parts, chronologically ranged. Fist, we study in Part I the prediction model proposed by Antoniadis et al. (2006) for strictly stationary functional valued processes applied to the french electrical power demand. Here, the french national load curve record is divided on blocs representing the daily demand. Prediction for next day bloc is obtained using the original prediction model. The data presents departures from stationary that the model is not able to cope with, namely the evolution in time of the mean level and the existence of groups of blocs. We explore some corrections that enhance the prediction performance. The corrections aim to take into account the presence of these nonstationary features. We adapt some of the ideas proposed in Poggi (1994) in order to predict the mean level evolution of the process. To handle the existence of groups, we constraint the predictor to use only the blocs of the past that belongs to the same group of the present bloc. If one knows the blocs grouping, a simple post-treatment suffices to obtain better prediction performances. The results of this part were presented in WIPFOR 2010 (Workshop Industry & Price Forecasting, June 2010, Paris).

If the block grouping is unknown, we propose to discover the grouping from data using clustering analysis. The infinite dimension of the not necessarily stationary trajectories may be taken into account by the clustering algorithm. In Part II we propose two strategies to do it, both based on wavelet transforms. The first one uses a feature extraction approach through the DWT. Extracted features quantify the cycle they represent. A feature selection algorithm is then used to select the significant features to be used on a classical clustering algorithm. The second approach clusters directly the functions by means of a dissimilarity measure over the CWT spectra. After clustering the functional process, one may use the resulting cluster memberships as groups of stationarity and use the mentioned post-treatment on the prediction. This second part of the thesis is submitted to *Advances on Data Classification and Clustering* on 2010 and was also presented in COMPSTAT 2010.

While the resulting model of prediction produces nice results sometimes comparable to operational models (namely on special days like public holidays days), we should remark that no exogenous information has been introduced yet. On the particular case of the electricity demand, one may want to use the available information of predicted weather conditions to enhance the predictions. The third part of thesis is dedicated to explore one way of incorporating an exogenous variable when using the ARH framework for prediction. We propose a new class of processes that we call *Conditional Autoregressive Hilbertian* (CARH) and develop the equivalent of projection and resolvent classes of estimators to predict such processes.

Première partie

Prédiction par des méthodes à noyau pour des variables fonctionnelles.

Sommaire

Dans ce chapitre nous présentons la méthode de prévision appliquée à une suite de fonctions proposée dans Antoniadis et al. (2006). La méthode approche les fonctions à l'aide de la transformée en ondelettes discrètes (DWT) par un vecteur de dimension finie. Les bonnes propriétés d'approximation de la DWT permettent de n'utiliser que la suite des vecteurs ainsi obtenue et d'accomplir la tâche de prévision comme dans le cas d'un processus multivarié.

Tout d'abord, nous commençons par faire un rappel de la prévision par méthode à noyau pour dans le cas d'un processus vectoriel (Section 5) et de la transformée en ondelettes discrète (Section 6). Ensuite, nous traitons dans la Section 7 les cas d'autorégression d'un processus fonctionnel. Les fondements théoriques de la méthode reposent sur l'hypothèse de stationnarité du processus fonctionnel. Or, la suite des consommations d'électricité journalières ne peut pas être considérée comme stationnaire. D'une part le niveau moyen de la consommation évolue avec notamment deux composantes non stationnaires : une tendance lente et croissante et un cycle annuel très marqué. D'autre part, il existe une claire différence dans l'allure des profils journaliers entre les jours de la semaine et les jours du weekend ou les jours fériés. Nous utilisons et étendons quelques idées proposées dans Poggi (1994) pour gérer les non-stationnarités (Section 9). Quelques expériences numériques additionnelles sont fournies dans la Section 10. Elles ont pour but de mieux comprendre les éléments constitutifs du prédicteur.

5 Le cas des processus multivariés.

Considérons un processus stochastique stationnaire à valeurs réelles à temps discret $Y = (Y_i, i \in \mathbb{N})$. Si l'on suppose que le processus vérifie la propriété de Markov, on peut écrire pour un entier d

$$\mathbb{E}[Y_n|Y_{n-1}, \ldots, Y_0] = \mathbb{E}[Y_n|Y_{n-1}, \ldots, Y_{n-d}].$$

Nous allons noter $\mathcal{Y}_{n,d} = (Y_n, Y_{n-1}, \ldots, Y_{n-d+1})$. Nous supposons le modèle suivant pour le processus

$$Y_{n+1} = m(\mathcal{Y}_{n,d}) + \epsilon_n,$$

où $m(y) = \mathbb{E}(Y_{n+1}|\mathcal{Y}_{n,d} = y)$ avec $y \in \mathbb{R}^d$ et $(\epsilon_n, n \in \mathbb{N})$ est un bruit blanc (i.e. une suite de variables aléatoires réelles avec $\mathbb{E}\epsilon_n = 0$ pour tout $n \in \mathbb{N}$ et $\mathbb{E}(\epsilon_i\epsilon_j) = \sigma^2$ si $i = j$ et $\mathbb{E}(\epsilon_i\epsilon_j) = 0$ sinon) indépendant de $\mathcal{Y}_{n,d}$.

Nous sommes intéressés par le comportement futur de Y à horizon s, $Y_{n+s}, s > 0$, étant donné les observations $Y_1, \ldots, Y_n$, pour $n > d$. Nous appelons $\widetilde{Y}_{n+s}$ le prédicteur de Y_{n+s} construit à partir de ces observations. Un critère pour l'obtenir est de minimiser l'erreur quadratique moyenne, c'est-à-dire prendre $\widetilde{Y}_{n+s} = y_0$, où y_0 est l'argument qui minimise $w \mapsto \mathbb{E}[(w - Y_{n+s})^2|\mathcal{Y}_{n,d}]$. La solution s'avère être $y_0 = m(\mathcal{Y}_{n,d})$. Bien sûr ceci n'est pas un prédicteur statistique car il dépend de la loi inconnue du processus sous-jacent.

Toutefois, un prédicteur dans le sens statistique peut être obtenu si l'on remplace la loi du processus par son analogue empirique. Pour ce faire, remarquons que m est une espérance conditionnelle. Elle correspond à la fonction de régression de l'instant $n + s$ du processus Y, i.e. Y_{n+s}, contre les d dernières observations retardées du même processus.

On parle donc d'une autorégression. Nous supposons que le processus Y est stationnaire et qu'il existe une densité $f_{Y,\mathcal{Y}}$ pour le couple $(Y_{n+s}, \mathcal{Y}_{n,d})$. Il est utile d'introduire les quantités suivantes

$$g(y) = \int_{\mathbb{R}} w f_{Y_s,\mathcal{Y}_{0,d}}(w,y)dw, \qquad y \in \mathbb{R}^d$$
$$f(y) = \int_{\mathbb{R}} f_{Y_s,\mathcal{Y}_{0,d}}(w,y)dw, \qquad y \in \mathbb{R}^d.$$

Puis, on écrit $m(y) = g(y)/f(y)$ si la densité f est non nulle en y, sinon nous définissons $m(y) = E(Y_0)$.

Du fait que m soit une espérance conditionnelle (au moins pour tout y dans le support de f), il est habituel de l'estimer par une technique non paramétrique car la régression entre Y_s et $\mathcal{Y}_{0,d}$ peut être non linéaire et assez complexe. Nous allons utiliser un estimateur introduit par Nadaraya (1964) et Watson (1964) appelé estimateur à noyau.

Pour simplifier, nous allons utiliser comme noyau $K : \mathbb{R}^d \mapsto \mathbb{R}$ une fonction de densité de probabilité multivariée symétrique autour de l'origine. L'estimateur à noyau $\hat{m}_n$ de m s'écrit de la façon suivante

$$\hat{m}_n(y) = \begin{cases} \hat{g}_n(y)/\hat{f}_n(y) & \text{si } \hat{f}_n(y) > 0 \\ (1/n)\sum_{i=1}^n Y_i & \text{si } \hat{f}_n(y) = 0 \end{cases},$$

où

$$\hat{g}_n(y) = \frac{1}{n_0 h_n^d} \sum_{i=d}^{n-s} K\left(\frac{\mathcal{Y}_{i,d} - y}{h_n}\right) Y_{i+s},$$
$$\hat{f}_n(y) = \frac{1}{n_0 h_n^d} \sum_{i=d}^{n-s} K\left(\frac{\mathcal{Y}_{i,d} - y}{h_n}\right),$$

pour $y \in \mathbb{R}^d$, $n_0 = n - d - s$ et $(h_n)_n$ une suite réelle positive décroissante. Le prédicteur à noyau s'en déduit

$$\widehat{Y_{n+s}} = \hat{m}_n(\mathcal{Y}_{n,d}).$$

Pour mieux comprendre le rôle du noyau, il est utile de réécrire l'estimateur $\hat{m}_n$. Nous présentons dans le cas $\hat{f}_n(y) > 0$ et posons $w_{n,i}(y) = \frac{K_{h_n}(\mathcal{Y}_{i,d}-y)}{\sum_{t=d}^{n-s} K_{h_n}(\mathcal{Y}_{i,d}-y)}$ avec $K_{h_n}(.) = K(./h_n)$ pour obtenir

$$\hat{m}_n(y) = \sum_{i=d}^{n-s} w_{n,i}(y) Y_{t+s}.$$

Les $\{w_{n,i}, i = 1, \ldots, n-s\}$ sont des quantités positives dont la somme fait 1. Ainsi, la prévision à horizon s du processus Y étant donné l'historique

$$\widehat{Y_{n+s}} = \sum_{t=d}^{n-s} w_{n,i}(\mathcal{Y}_{n,d}) Y_{i+s},$$

peut être vue comme un barycentre des futurs d'horizon s des blocs du passé pondérés par la similarité entre le bloc présent et ses homologues dans le passé. Ainsi, le poids de chaque bloc $\mathcal{Y}_{i,d}$ est croissant avec sa similarité par rapport au bloc actuel $\mathcal{Y}_{n,d}$. Ainsi, l'idée est que des passés semblants ont des futurs similaires.

Pour un échantillon de taille n donnée, la quantité h_n appelée *largeur de fenêtre* est cruciale pour la performance de la méthode (Härdle (1990)). Elle est un compromis entre le biais et la variance de l'estimateur. Une valeur trop faible conduira à un estimateur très local et en conséquence avec une variabilité importante. Si l'on considère des valeurs de plus en plus grandes (on dit alors qu'on élargit la fenêtre), on permet à de plus en plus d'observations de contribuer à l'estimation ce qui diminue la variance mais peut conduire à un estimateur plus biaisé. Le réglage de ce paramètre se fait habituellement par validation croisée.

6 Transformée en Ondelettes.

La *Transformée en Ondelettes* est une technique de décomposition hiérarchique des signaux d'énergie finie qui permet de représenter un signal dans le domaine temps-échelle, où l'échelle joue un rôle analogue à celui de la fréquence dans l'analyse de Fourier (Daubechies (1992)). Elle permet de décrire une fonction à valeurs réelles au travers de deux objets : une approximation de cette fonction et un ensemble de détails. La partie approximation résume la tendance globale de la fonction, alors que les changements localisés (en temps et fréquence) sont capturés dans les composantes de détails à différentes résolutions.

L'analyse des signaux est réalisée par des fonctions analysantes appelées *ondelettes* obtenues à partir de transformations simples d'une *ondelette mère*. Une ondelette est une fonction oscillatoire assez régulière avec une rapide décroissance vers zéro. En définitive, une ondelette est une fonction oscillante du temps et elle est localisée dans le domaine fréquentiel et temporel. La Figure 2 montre une ondelette orthogonale à support compact. De telles ondelettes sont souvent très asymétriques. Toutefois, Daubechies a construit une famille d'ondelettes orthogonales où les fonctions analysantes sont, pour une longueur de support donnée, les moins asymétriques possibles (Daubechies (1992)).

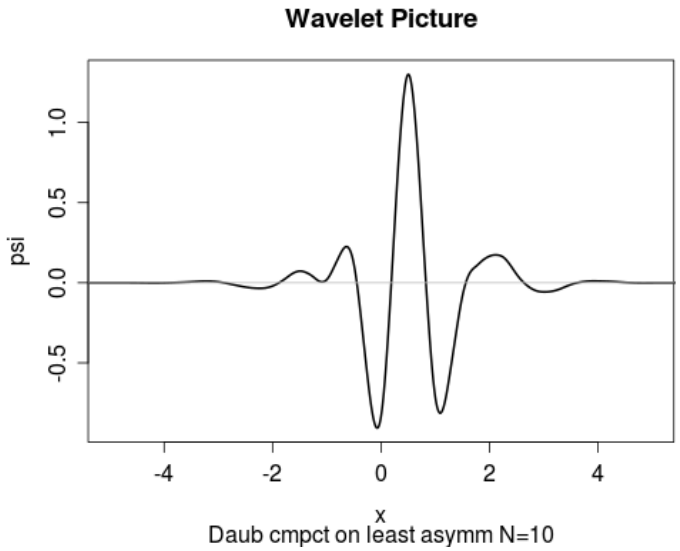


FIGURE 2 – Une des ondelettes à support compact proposées par Daubechies.

6.1 Transformée en ondelettes discrète.

Dans cette première partie de la thèse nous allons utiliser la transformée en ondelettes discrète (DWT par son nom en anglais *Discrete Wavelet Transform*), introduite dans la suite, pour représenter des trajectoires d'un processus temporel. Dans un premier temps nous allons considérer que ces trajectoires sont des objets dans l'espace de fonctions d'énergie finie $L_2(\mathbb{R})$. Après un choix approprié de l'ondelette mère, la transformée en ondelettes discrète fournira une base orthonormale de l'espace, obtenue par des dilatations dyadiques et des translations entières de cette unique fonction mère, i.e. par une famille du type

$$\left\{\psi_{j,k}(t) = \frac{1}{\sqrt{2^j}}\psi\left(\frac{t-2^j k}{2^j}\right)\right\}_{(j,k)\in\mathbb{Z}^2}.$$

En conséquence, si $z(t) \in L_2(\mathbb{R})$, elle admettra une décomposition dans cette base

$$z(t) = \sum_{(j,k)\in\mathbb{Z}^2} d_{j,k}\psi_{j,k}(t), \tag{6.1}$$

où les *coefficients d'ondelettes* $d_{j,k}$ de z à l'échelle j et la position k s'obtiennent par la projection de z sur $\psi_{j,k}$, i.e., $d_{j,k} =< z, \psi_{j,k} >$. Mallat (1989a,b) propose d'utiliser une suite de sous-espaces d'approximation appelée analyse multirésolution (AMR). Si nous choisissons les éléments de la base à support compact (Daubechies (1988)), nous aurons des algorithmes encore plus efficaces.

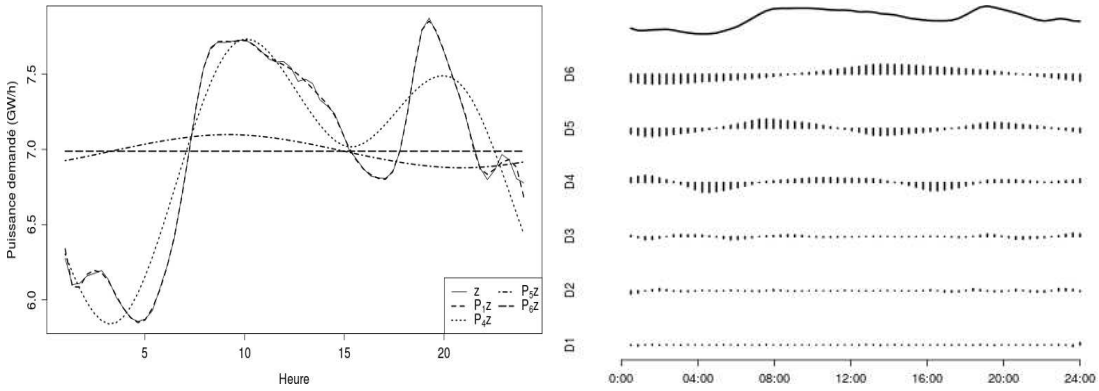
6.2 Analyse multirésolution.

Une *analyse multirésolution* de $L_2(\mathbb{R})$ est une suite de sous-espaces fermés $\{V_j\}_{j\in\mathbb{Z}}$ qui vérifie toutes les propriétés suivantes :

1. $V_j \subset V_{j+1}, \forall j \in \mathbb{Z}$
2. $\bigcap_{j\in\mathbb{Z}} V_j = \{0\}$,
3. $\overline{\bigcup_{j\in\mathbb{Z}} V_j} = L_2(\mathbb{R})$,
4. $z(\bullet) \in V_j \Leftrightarrow z(2\bullet) \in V_{j-1}$,
5. il existe une fonction $\phi \in V_0$, nommée *fonction d'échelle*, telle que

$$V_0 = \{f \in L_2(\mathbb{R}) : f(\bullet) = \sum_{k\in\mathbb{Z}} \alpha_k \phi(\bullet - k)\}.$$

L'AMR de $L_2(\mathbb{R})$ permet de décrire les éléments de l'espace $L_2(\mathbb{R})$ comme une suite d'approximations sur les sous-espaces V_j liés à des résolutions 2^j, $j \in \mathbb{Z}$. En passant de la résolution 2^j au niveau 2^{j+1} on obtient une version plus raffinée de l'approximation. Ainsi, si l'on appelle $A_j z$ une version lissée de $z \in L_2(\mathbb{R})$ liée à la résolution 2^j nous obtenons par passage à la limite $\lim_{j\to+\infty} A_j z = z$. À titre d'exemple, nous représentons dans la Figure 3 une fonction de charge journalière par ses approximations à des niveaux de résolution différents. L'approximation la plus grossière ne donne qu'une idée du niveau moyen de la courbe. Au fur et à mesure que l'on augmente la résolution, l'approximation s'affine de plus en plus laissant apparaître des phénomènes locaux comme la montée du matin et le pic du soir.



(a) Approximations (b) Détails

FIGURE 3 – À droite, une courbe de charge journalière et ses approximations à différentes résolutions. À gauche, la courbe de charge, son approximation à la plus basse résolution et touts les détails de résolutions supérieures.

La fonction d'échelle ϕ joue un rôle central. Une fois spécifiée, nous construisons l'AMR en posant V_j égal à la fermeture du sous-espace vectoriel engendré par des transformations d'échelle dyadique et translations entières $\{\phi_{j,k} = 2^{-j/2}\phi(2^j t - k)\}_{(j,k)\in\mathbb{Z}^2}$. La fonction d'échelle est caractérisée par un filtre discret nommé filtre miroir conjugué (Mallat (1999)). Il est habituel de choisir un filtre au lieu de la fonction elle-même car des conditions nécessaires et suffisantes existent pour qu'un tel filtre définisse une AMR. Sans perte de généralité on peut supposer que $\{\phi(t-k)\}_{k\in\mathbb{Z}}$ est une base orthonormale de V_0 (dans le cas contraire, on peut la rendre orthonormale sans altérer ses propriétés d'analyse, Daubechies (1988)). Alors, on en déduit que pour chaque j, la famille $\{\phi_{j,k}\}_{k\in\mathbb{Z}}$ est une base orthonormée de V_j.

De manière intuitive, pour chaque niveau de résolution j nous fixons $A_j = P_j$ avec P_j l'opérateur de projection orthogonale sur V_j. Ainsi, $P_j z$ est alors la projection orthogonale de z. La définition donne aussi une interprétation des cas extrêmes : au fur et à mesure que le niveau d'échelle j tend vers l'infini, l'erreur de l'approximation diminue donnant $\|P_j z - z\|_{L_2}$ égal à zéro à la limite, et donc l'approximation converge vers le vrai signal.

Par la suite, on pose W_j le sous-espace de V_{j+1} orthogonal a V_j. Ainsi, on obtient une autre suite de sous-espaces $\{W_j\}_{j\in\mathbb{Z}}$ fermés orthogonaux de $L_2(\mathbb{R})$. L'information nécessaire pour passer de l'approximation au niveau d'échelle j à celle du niveau plus détaillé $j+1$ se trouvera dans les sous-espaces W_j. A partir de la fonction d'échelle ϕ on peut construire une fonction ondelette mère ψ (Daubechies (1992)). La construction conduit à ce que la famille

$$\{\psi_{j,k} = 2^{-j/2}\psi(2^j t - k)\}_{(j,k)\in\mathbb{Z}^2}$$

soit une base orthonormale de W_j pour chaque entier j.

Avec les deux suites de sous-espaces considérés, l'espace de départ s'écrit sous la forme d'une somme directe :

$$L_2(\mathbb{R}) = V_{j_0} \oplus \left(\bigoplus_{j\geq j_0} W_j \right)$$

pour tout entier j_0. En conséquence, si $z \in L_2$ le développement en ondelettes permet d'avoir l'écriture suivante

$$z = \sum_{k\in\mathbb{Z}} c_{j_0,k}\phi_{j_0,k} + \sum_{j\geq j_0}\sum_{k\in\mathbb{Z}} d_{j,k}\psi_{j,k}. \tag{6.2}$$

Les coefficients sont définis par

$$c_{j,k} =< z, \phi_{j,k} > \text{ et } d_{j,k} =< z, \varphi_{j,k} >,$$

et on les appelle respectivement *coefficients d'approximation (ou d'échelle)* et *coefficients d'ondelettes (ou de détails).*

Dans l'expression (6.2) le premier et le second termes à droite de l'égalité sont, respectivement, l'approximation au niveau de résolution j_0 donnée par la projection orthogonale de z dans V_{j_0} et l'erreur d'approximation (projection sur le complément orthogonal de V_{j_0}) composée de l'agrégation des détails des niveaux d'échelles $j \geq j_0$. Ces deux composantes, l'approximation et les détails, peuvent être vues comme une partie d'approximation lisse non stationnaire qui contient les bases fréquences, et une composante qui garde l'information de détails localisés dans le temps pour les petites échelles. Enfin, le paramètre j_0 détermine la séparation entre les composantes.

Du fait que la DWT utilise une base orthonormale de L_2, on a la conservation de l'énergie des signaux de carré intégrable pour la transformée en ondelettes orthogonale.

En plus d'un développement d'une fonction, l'équation (6.2) permet d'obtenir la fonction originale à l'aide des coefficients d'ondelettes et d'échelle. Le passage à la transformée inverse s'effectue sans perte d'information.

Lorsque l'on combine la puissance de la DWT et l'AMR, on obtient une interprétation intuitive de la transformée. Les bonnes propriétés de localisation dans le domaine temps-fréquence et la capacité d'adapter la résolution de l'analyse à l'échelle, font que la transformée est appropriée pour approcher des courbes qui contiennent des structures très localisées.

6.3 Aspects pratiques.

Jusqu'à présent on a traité le cas de la transformée d'une fonction observée sur la droite réelle. D'un point de vue pratique, une fonction n'est observée que sur une grille fine de N points, et souvent les mesures sont faites avec du bruit. Encore plus important, les fonctions à traiter sont définies sur des intervalles. Nous allons aborder maintenant ces questions.

Nous sommes intéressés par des fonctions échantillonnées sur une grille équidistante de taille $N = 2^J$ pour un entier J. Si N n'est pas une puissance de 2, on peut utiliser une interpolation jusqu'aux 2^J points pour un entier J vérifiant $2^{J-1} < N < 2^J$. La structure hiérarchique de l'AMR permet d'obtenir un algorithme, en forme d'arbre ou pyramide, très performant pour le calcul de la DWT. Ainsi, tous les coefficients de la DWT d'une fonction z sont obtenus à partir des coefficients d'échelle à l'échelle J la plus fine $c_{J,k} =< z, \phi_{J,k} >_{L_2(\mathbb{R})}$ (Mallat (1989)). Cependant, la projection $P_J z$ doit être approchée par un opérateur Π_J à partir des valeurs échantillonnées $\mathbf{z} = \{z(t_l) : l = 0, \ldots, N-1\}$ de

z car aucune méthode générale existe pour calculer les $c_{J,k}$. Si l'ondelette est suffisamment régulière, l'approximation $\widetilde{z}_J$ de z est de très bonne qualité (Antoniadis, 1994). Des variantes existent pour des grilles plus générales, pas nécessairement équidistantes (voir par exemple Amato et al. (2006)). Alors, l'approximation Π_J de la projection P_j de z s'écrit à partir de (6.2) en imposant une troncature due aux 2^J points de l'échantillon $\mathbf{z}$ et le niveau de résolution j_0

$$\widetilde{z}_J(t) = \sum_{k=0}^{2^{j_0}-1} c_{j_0,k}\phi_{j_0,k}(t) + \sum_{j=j_0}^{J-1}\sum_{k=0}^{2^j-1} d_{j,k}\psi_{j,k}(t), \tag{6.3}$$

où maintenant $c_{j_0,k}$ et $d_{j,k}$ sont les coefficients empiriques de la DWT des valeurs de $\mathbf{z}$.

Quand l'on utilise le vecteur $\mathbf{z}$ comme donnée d'entrée dans l'algorithme de la DWT, des problèmes de bord se posent. Pour les traiter, plusieurs variantes existent. On peut par exemple supposer que les fonctions à analyser sont périodiques ou symétriques pour les prolonger en dehors de intervalle de définition..

Dans la suite, notre attention sera portée par des fonctions définies sur un intervalle compact. Cohen et al. (1993) proposent une version de la transformée en ondelettes spécialement développée pour traiter des fonctions définies sur un intervalle compact appelée "ondelettes sur l'intervalle". Nous allons utiliser ici un chemin plus simple. Sans perte de généralité, nous allons considérer que les fonctions en question seront définies sur l'intervalle $[0,1]$. Une AMR de $L_2([0,1])$ peut être obtenue si nous définissons des versions périodiques de la fonction d'échelle et de l'ondelette mère, à savoir

$$\phi^P(t) = \sum_{l\in\mathbb{Z}} \phi(t-l) \qquad \text{et} \qquad \psi^P(t) = \sum_{l\in\mathbb{Z}} \psi(t-l), \qquad t \in [0,1].$$

Ainsi, des versions dilatées et translatés de ϕ^P et ψ^P sont définies en posant

$$\phi^P_{j,k}(t) = 2^{j/2}\phi^P(2^j t - k), \qquad \psi^P_{j,k}(t) = 2^{j/2}\psi^P(2^j t - k).$$

Enfin, pour tout $j_0 \in \mathbb{Z}$, la collection

$$\{\phi^P_{j_0,k}, k = 0,1,\ldots,2^{j_0}-1; \psi^P_{j,k}, j \geq j_0, k = 0,1,\ldots,2^j-1\},$$

est une base orthonormale de $L_2[0,1]$.

7 Le cas des processus fonctionnels.

Nous allons procéder par analogie au cas multivarié que nous venons d'étudier pour décrire la méthode de prévision proposé par Antoniadis et al. (2006) que nous appellerons KWF. Maintenant nous considérons une processus stochastique stationnaire $Z = (Z_i)_{i\in\mathbb{Z}}$ à valeurs dans un espace fonctionnel H (par exemple $H = L_2([0,1])$). Nous disposons d'un échantillon de n courbes $Z_1,\ldots,Z_n$ et l'objectif est de prévoir Z_{n+1}. Rappelons que la méthode de prévision étudiée en Section 5 se décompose en deux phases :

- D'abord, trouver parmi les blocs du passé ceux qui sont le plus semblables au dernier bloc observé.
- Ensuite construire un vecteur de poids $w_{n,i}, i = 1,\ldots,n-1$ pour obtenir la prévision souhaitée en moyennant les futurs des blocs correspondant aux indices $2,\ldots,n$ respectivement.

7.1 Présentation du prédicteur.

La méthode de prévision nécessite une distance appropriée entre les objets observés. Dans notre cas, les objets observés sont des courbes plus ou moins régulières. Nous devrons utiliser une distance qui prenne en compte la nature infini-dimensionnelle de ces objets. Dans Antoniadis et al. (2006) les auteurs proposent de représenter chaque segment $Z_i, i = 1, \ldots, n$ par son développement sur une base d'ondelettes tronqué à une échelle $J > j_0$. Ainsi, chaque observation Z_i est décrite par sa version tronquée

$$Z_{i,J}(t) = \sum_{k=0}^{2^{j_0}-1} c_{j_0,k}^{(i)} \phi_{j_0,k}(t) + \sum_{j=j_0+1}^{J} \sum_{k=0}^{2^j-1} d_{j,k}^{(i)} \psi_{j,k}(t), \qquad t \in [0,1].$$

Le premier terme de l'équation est une approximation lisse à la résolution j_0 du comportement global de la trajectoire. Elle contient les composantes non stationnaires associées à des basses fréquences et à une tendance. Le deuxième terme conserve l'information de la structure locale de la fonction. Cette information, décrite en fonction de l'échelle j et la position temporelle k dans l'échelle, sera exploitée pour construire une distance.

Pour deux segments observés $Z_i(t)$ et $Z_{i'}(t)$, nous utilisons la distance euclidienne entre les vecteurs des coefficients de détail d'ondelettes à chaque échelle $j > j_0$

$$\mathrm{dist}_j(Z_i, Z_{i'}) = \left(\sum_{k=0}^{2^j-1} (d_{j,k}^{(i)} - d_{j,k}^{(i')})^2 \right)^{1/2},$$

puis on agrège les distance des échelles en prenant en compte le nombre de coefficients par échelle

$$D(Z_i, Z_{i'}) = \sum_{j \geq j_0+1} 2^{-j/2} \mathrm{dist}_j(Z_i, Z_{i'}).$$

En effet D définit une distance dans le sous-espace orthogonal à l'espace d'approximation V_{j_0}, pour tout $Z_m, Z_l, Z_k \in L_2$ nous avons les propriétés suivantes :

$$\begin{aligned}
&D(Z_m, Z_l) \geq 0 && \text{(positive)}\\
&D(Z_m, Z_l) = 0 \Leftrightarrow (I - P_{j_0})(Z_m - Z_l) = 0 && \text{(séparation)}\\
&D(Z_m, Z_l) = D(Z_l, Z_m) && \text{(symétrie)}\\
&D(Z_m, Z_l) \leq D(Z_m, Z_k) + D(Z_l, Z_k) && \text{(inégalité triangulaire)}
\end{aligned}$$

Les trois premières propriétés sont évidentes. Pour la dernière il suffit de noter que D est une combinaison linéaire de distances euclidiennes avec des coefficients positifs.

Les coefficients d'échelle ne contient pas d'information utile pour la prévision car le processus Z est stationnaire. De ce fait, ils ne sont pas pris en compte dans la distance proposée. En d'autres mots, la "distance" D permet de trouver de motifs similaires entre courbes même si elles ont des approximations très différentes. Encore, la capacité de la DWT pour détecter des caractéristiques locales d'une fonction motive l'utilisation de cette distance, car structures locales ne s'expriment qu'à travers les détails.

Nous voudrions calculer les distances de chaque segment $Z_1, \ldots, Z_{n-1}$ au segment présent Z_n. En pratique, nous ne pouvons pas utiliser D mais une version $\widetilde{D}$ issue de l'approximation des observations Z_i par $Z_{i,J}$

$$\widetilde{D}(Z_i, Z_j) = \sum_{j=j_0+1}^{J} 2^{-j/2} \mathrm{dist}_j(Z_i, Z_{i'}).$$

Pour la deuxième phase de la méthode, nous allons noter $\Xi_i = \{c_{J,k}^{(i)} : k = 0, 1, \ldots, 2^J - 1\}$ l'ensemble des coefficients d'échelle du i-ème segment Z_i à la résolution J, la plus fine. La prévision par noyau des coefficients d'échelle $\widehat{\Xi_{n+1}}$ de Ξ_{n+1} est donnée par

$$\widehat{\Xi_{n+1}} = \frac{\sum_{m=1}^{n-1} K_{h_n}(\widetilde{D}(Z_n, Z_m))\Xi_{m+1}}{1/n + \sum_{m=1}^{n-1} K_{h_n}(\widetilde{D}(Z_n, Z_m))}.$$

Notons que la distance $\widetilde{D}$ utilise les coefficients d'ondelettes et non les trajectoires entières. De ce fait, nous pouvons coder dans un premier temps les courbes Z_i par Ξ à l'aide de la DWT. Puis, en utilisant l'algorithme pyramidal sur Ξ, nous obtenons l'ensemble des coefficients d'ondelettes à utiliser dans la distance.

Finalement, nous appliquons la transformée inverse de la DWT sur $\widehat{\Xi_{n+1}}$ pour obtenir la prévision de la courbe Z_{n+1} dans le domaine temporel

$$\widehat{Z_{n+1}(t)} = \sum_{k=0}^{2^J-1} \widehat{c_{J,k}^{(n+1)}} \phi_{J,k}(t).$$

La prévision ponctuelle de la courbe Z_{n+1} peut être accompagnée par une bande de confiance. Les auteurs proposent de le faire par la technique de bootstrap sur les courbes en utilisant les poids $w_{n,i}$ issus du prédicteur. La bande de confiance est considérée point par point pour les instants $t_l, l = 1, \ldots, P$. La validité de la méthode de prévision, ainsi que la couverture de la bande de confiance sont démontrés théoriquement ainsi que par de exemples que nous allons reprendre dans les sections suivantes.

Désormais nous allons nous centrer dans un cadre plus appliqué. La mise-en-œuvre de la méthode de prévision pour fournir une prévision de la consommation d'électricité nous a amené d'abord à une réflexion sur le rôle de chacun des paramètres de réglage du prédicteur. Puis, l'utilisation de la méthode sur de vrais historiques de consommation a mis en évidence sa faiblesse vis-à-vis d'écarts à l'hypothèse de stationnarité du processus. Ces deux sujets sont abordés par la suite.

7.2 Les paramètres de réglage du prédicteur.

Nous allons traiter dans cette section de problèmes plutôt pratiques liés au choix des éléments utiles pour construire le prédicteur de la méthode KWF. Nous les regroupons en trois, à savoir :

- ceux concernant le découpage en blocs pour obtenir le processus Z,
- ceux concernant la transformation dans le domaine d'ondelettes,
- ceux concernant l'estimateur à noyau.

Mais avant de les examiner, revenons sur le paramètre δ définissant la taille des blocs.

7.2.1 Paramètres liés au découpage en blocs.

Taille des blocs : paramètre δ. Pour obtenir le processus à temps discret $Z = (Z_i(t), t \in [0, \delta])_{i\in\mathbb{N}}$, nous avons procédé au découpage d'une trajectoire d'un processus stochastique continu à valeurs réelles X en blocs de taille δ. Il n'y a pas d'éléments théoriques qui puissent guider le choix de δ. Néanmoins, nous pouvons le choisir de manière à modéliser une saisonnalité. Ainsi, par exemple pour les données de la consommation d'électricité qui présentent une périodicité journalière, le fait de découper en courbes journalières rend plus raisonnable l'hypothèse de stationnarité du processus fonctionnel Z. Dans le même direction, nous pourrions penser à des saisonnalités hebdomadaires et ainsi de suite. Cependant, un choix de δ trop large peut conduire à de segments d'allure très particulière. Ceci peut devenir un problème pour des méthodes à base de similarités, car il y aura peu ou pas de segments qui ressemblent dans le passé à un segment donné. Nous serons face à ce problème dès que nous traiterons les jours fériés ou les jours à tarification spéciale dans la consommation d'électricité.

7.2.2 Paramètres liés à la DWT.

L'ondelette. Afin d'utiliser la DWT, une ondelette mère admissible doit être choisie. Bien qu'en théorie différentes ondelettes peuvent produire de coefficients différents, il a été montré que la technique est assez robuste vis-à-vis l'ondelette choisie. Alors le choix est guidé par des aspects pratiques.

Dans Vidakovic (1999) nous pouvons trouver un catalogue de différentes ondelettes. Notre intérêt est porté par des ondelettes orthonormales à support compact proposées par Daubechies (Daubechies, 1992) qui conduisent à un algorithme pour la transformée en ondelette très performant. Les ondelettes orthonormales à support compact ne peuvent pas être de fonctions symétriques si elles sont à valeurs réelles. Or, parmi les familles proposés il y en a une qui présente des ondelettes avec une allure moins asymétrique. Elle est connue comme la famille d'ondelettes "moins asymétriques" ou "symmlets". Ses éléments peuvent être paramétrisés par la taille du filtre associé. Un filtre de petite taille peut introduire des effets non souhaités dans les coefficients alors que un taille trop grande peuvent provoquer des problèmes de bords et générer trop de coefficients non nuls (Percival and Walden (2006)).

Nous utilisons l'ondelette de la famille d'ondelettes "moins asymétriques" appelée *Symmlet 6*, avec un filtre de taille 6 ce qui parait un bon compromis compte tenu de la taille de nos segments.

Interpolation à $N = 2^J$ Afin d'utiliser l'algorithme pyramidal de Mallat pour le calcul de la transformée en ondelettes, nous devons compter avec des vecteurs d'entrée de $N = 2^J$ points avec J un entier non nul. Si ce n'est pas le cas, nous utilisons une interpolation par splines naturelles pour le plus proche entier J qui vérifie $2^{J-1} < N < 2^J$. Des tests préliminaires nous ont permis de vérifier de manière empirique ce choix en regardant la répartition des énergies de chaque échelle de la DWT. Nous avons remarqué que quand une interpolation avec $2^{J'}$ points, $J' > J$, est faite, des échelles supplémentaires sont introduites avec peu ou aucune information significative (essentiellement il n'y a que du bruit dans les échelles additionnelles).

Résolution de l'approximation. Pour une ondelette et une taille de filtre donnés, nous approchons les courbes à la résolution j_0 qui doit être choisie. Toutes les résolutions $j < j_0$ ne feront pas partie de la prévision. C'est justement la capacité des ondelettes de faire une analyse à plusieurs résolutions qui nous intéresse. L'échelle j_0 fait la séparation entre les échelles de l'analyse multirésolution qui sont liées aux basses fréquences (et ainsi associées à des composantes non stationnaires comme la tendance) et celles liées aux hautes fréquences (associées à la partie stationnaire du processus). Il y a un compromis à faire : d'un côté une fréquence trop basse pourrait inclure dans la comparaison des phénomènes non stationnaires ; d'une autre côté des fréquences trop hautes pourraient contenir trop de bruit et assez peu de signal utile. Pour commencer nous allons utiliser toutes les échelles de l'AMR empirique, donc considérer $j_0 = 0$.

7.2.3 Paramètres liés à l'estimateur à noyau.

Distances sur des courbes échantillonnées. Dans le paragraphe précédent, nous n'avons pas pris en compte le fait que les courbes ne sont pas observées dans leur intégralité. En effet, nous disposons seulement d'une grille de valeurs pour chaque Z_i, disons $\mathbf{Z_i} = \{Z_i(t_l), l = 1, \ldots, P_i\}$ pour $i = 1, \ldots, n$ avec $0 \leq t_1 < t_2 < \ldots < t_{P_i} \leq 1$. Pour faire simple, supposons que la grille d'échantillonnage est la même pour toutes les observations et la taille est de $P_i = P$.

Prenons comme exemple la distance L_2 entre deux courbes $Z_m(t)$ et $Z_l(t)$, $t \in [0, 1]$ définie par

$$D_2(Z_m, Z_l) = \left(\int_0^1 (Z_m(t) - Z_l(t))^2 dt \right)^{1/2}.$$

Habituellement, les courbes ne sont pas observées sur toute leur trajectoire. Dans ce sens, D_2 est une distance théorique. Nous n'observons Z_m et Z_l que sur les grilles discrètes d'instants $\mathbf{Z_m}$ et $\mathbf{Z_l}$ respectivement. Dans l'ensemble nous utilisons une distance conditionnelle $\widetilde{D_2}(Z_m, Z_l)$ donnée par

$$\widetilde{D_2}(Z_m, Z_l) = \left\{ \mathbb{E}\left[D_2^2(Z_m, Z_l) | \mathbf{Z_m}, \mathbf{Z_l} \right] \right\}^{1/2}.$$

Cette variable aléatoire $\widetilde{D_2}(.,.)$ vérifié les propriétés suivantes :

1. $\widetilde{D_2}(Z_m, Z_l) \geq 0$
2. $\widetilde{D_2}(Z_m, Z_m) = 0$ et pour $m \neq l$, $P(\widetilde{D_2}(Z_m, Z_l) > 0) = 1$
3. $\widetilde{D_2}(Z_m, Z_l) = \widetilde{D_2}(Z_l, Z_m)$
4. $\widetilde{D_2}(Z_m, Z_l) \leq \widetilde{D_2}(Z_m, Z_k) + \widetilde{D_2}(Z_k, Z_l)$

Par définition de $\widetilde{D_2}$ les 3 premières propriétés sont évidentes. Pour la dernière, nous utilisons la propriété d'inégalité triangulaire de la distance D_2 puis en prenant le carré

$$D_2^2(Z_m, Z_l) \leq D_2(Z_m, Z_k)^2 + D_2(Z_k, Z_l)^2 + 2D_2(Z_m, Z_k) D_2(Z_k, Z_l).$$

Nous prenons l'espérance conditionnelle suivante $\mathbb{E}[.|\mathbf{Z_m}, \mathbf{Z_l}, \mathbf{Z_k}]$ pour obtenir

$$\widetilde{D}_2^2(Z_m, Z_l) \leq \widetilde{D}_2^2(Z_m, Z_k) + \widetilde{D}_2^2(Z_k, Z_l) + 2\mathbb{E}(D_2(Z_m, Z_k) D_2(Z_k, Z_l) | \mathbf{Z_k}, \mathbf{Z_l}, \mathbf{Z_m}), \quad (7.1)$$

car $\mathbb{E}[D_2^2(Z_m, Z_l)|\mathbf{Z_m}, \mathbf{Z_l}, \mathbf{Z_k}] = \mathbb{E}[D_2^2(Z_m, Z_l)|\mathbf{Z_m}, \mathbf{Z_l}] = \widetilde{D}_2^2(Z_m, Z_l)$. Finalement, pour le terme restant l'application de Cauchy-Schwartz donne

$$\begin{aligned}\mathbb{E}[D_2(Z_m, Z_k)D_2(Z_k, Z_l)|\mathbf{Z_k}, \mathbf{Z_m}, \mathbf{Z_l}] &\leq \mathbb{E}[D_2(Z_m, Z_k)|\mathbf{Z_k}, \mathbf{Z_m}])^{1/2}\mathbb{E}[D_2(Z_k, Z_l)|\mathbf{Z_k}, \mathbf{Z_l}])^{1/2} \\ &\leq \widetilde{D_2}(Z_m, Z_k)\widetilde{D_2}(Z_k, Z_l)\end{aligned}$$

qui est incorporé à (7.1) pour arriver au résultat voulu.

L'utilisation d'une distance sur des courbes échantillonnées est basé sur le fait que, lorsque la grille d'échantillonnage devient de plus en plus fine, la distance conditionnelle $\widetilde{D}$ s'approche de la distance théorique D_2.

Le noyau. Le noyau est une fonction positive, symétrique et d'intégrale égale à un :

$$K(x) > 0, \qquad K(-x) = K(x) \ \ \forall x, \qquad \int_{\mathbb{R}} K(x)dx = 1.$$

En général les méthodes à noyau sont assez robustes par rapport au choix du noyau, le choix est encore une fois guidé par des aspects pratiques. Parmi les noyaux les plus utilisés on trouve

Nom	Noyau
Uniforme	$K(x) = \frac{1}{2}\mathbf{1}_{\{\lvert u\rvert \leq 1\}}$
Triangle	$K(x) = (1 - \lvert u\rvert)\mathbf{1}_{\{\lvert u\rvert \leq 1\}}$
Epanechnikov	$K(x) = \frac{3}{4}(1 - u^2)\mathbf{1}_{\{\lvert u\rvert \leq 1\}}$
Gaussien	$K(x) = \frac{1}{\sqrt{2\pi}}e^{-x^2/2}$

Les trois premiers noyaux ont un support compact mais avec différent degrés de régularité. Le quatrième, le plus utilisé, a pour support toute la droite réelle et conduit à des estimateurs réguliers.

La largeur de fenêtre. Le paramètre h_n joue un rôle crucial pour calibrer la méthode à noyau. Habituellement appelé "largeur de fenêtre" ou simplement "fenêtre", son réglage contrôle le compromis entre le biais et la variance du prédicteur.

Dans l'article original, les auteurs proposent de calculer h_n par *validation croisée* qui est souvent la technique utilisée. Toutefois, elle ne repose que sur des bases empiriques. Un autre article des mêmes auteurs proposent une autre forme du calcul, celle-là fondé sur des éléments théoriques (cf. Antoniadis et al. (2009)). Le calcul se fait par minimisation d'une fonction de *risque empirique* sur un échantillon d'apprentissage sur une grille de valeurs possibles. Si la grille couvre la vraie valeur du paramètre, alors cette stratégie ne peut pas conduire à un choix très éloigné de l'optimal.

Plusieurs questions se sont posées sur ce paramètre et nous les traiterons au fur et à mesure de leur apparition. Dans la méthode de base, les questions ont porté sur deux points.

Le premier concerne le paramètre à minimiser. D'une part nous pouvons minimiser directement la valeur de h_n. D'une autre nous pouvons écrire $h_n = c_n\sigma$, où σ est une

mesure de la variabilité des distances entre segments. Si σ est inconnu, il peut être estimé et sa valeur utilisée dans le prédicteur. Ainsi, la seule valeur à régler est la constante c_n que nous choisissons par minimisation de la fonction de risque empirique. L'intérêt de cette variante est simplement de réduire le temps de calcul.

Le deuxième point est lié à la fréquence de calcul de h_n quand le modèle de prévision est utilisé tout au long d'une période d'étude plutôt que concentré sur une seule date. La première option est de calculer et fixer la valeur h_n avant la période de prévision (FIX). L'option alternative consiste en mettre à jour cette valeur avant la prévision de chaque date (DYN). Ainsi, nous avons :

FIX calculer une seule valeur de h_n avant de la prévision du premier segment de la période d'étude et utiliser la même valeur pour toute la période, ou

DYN recalculer h_n avant la prévision de chaque segment de la période d'étude.

Cette problématique se pose dans tout modèle de prévision. En d'autres termes, quelle est la fréquence idéale de mise à jour du modèle ? Nous allons étudier des pistes de réponse dans la prochaine section.

8 Premières expériences numériques.

Dans un premier temps, nous illustrons la méthode KWF décrite précédemment sur un exemple simulé très simple. L'exemple est le même que celui utilisé par les auteurs pour présenter leur méthode. L'expérience est réalisée avec le logiciel `R`. Nous avons développé le package `kerwavfun` (voir annexe A) qui permet d'obtenir la prévision par la méthode de base, ainsi que pour toutes les variantes présentées par la suite. La DWT est réalisé par le biais du package `wavethresh` (Nason (2010)), qui fournit une implémentation de l'algorithme pyramidal de Mallat. Ensuite, nous appliquons la technique à la demande de consommation d'électricité française pour EDF.

8.1 Données simulées.

Nous simulons un processus continu $X = (X_t, t \in \mathbb{R})$ avec deux composantes cycliques et un bruit de structure de la forme d'une moyenne mobile :

$$\begin{aligned} X(t) &= \beta_1 m_1(t) + \beta_2 m_2(t) + \epsilon_t \\ m_1(t) &= \cos(2\pi t/64) + \sin(2\pi t/64) \\ m_2(t) &= \cos(2\pi t/6) + \sin(2\pi t/6) \\ \epsilon(t) &= \nu(t) + \theta\nu(t-1) \quad ; \quad \nu(t) \sim \mathcal{N}(0, \sigma^2) i.i.d.. \end{aligned}$$

Une des composantes domine l'évolution globale avec une période de 64 pas de temps. L'autre, moins forte, avec une période de six pas de temps peut être assimilée à une saisonnalité. Les valeurs utilisées pour les paramètres sont les suivantes : $\beta_1 = 0.8, \beta_2 = 0.18, \theta = 0.8, \sigma^2 = 0.05$. Nous simulons le processus X sur une grille de 30×64 instants. Pour obtenir un échantillon de 30 observations $\widetilde{Z}_{1,J}, \ldots, \widetilde{Z}_{30,J}$, des blocs de taille $\delta = 64$ sont découpés et donc $J = 6$ $(= \log_2 64)$. La Figure 4 représente un morceau d'une trajectoire simulé de X. Bien que nous ne disposions que des représentations

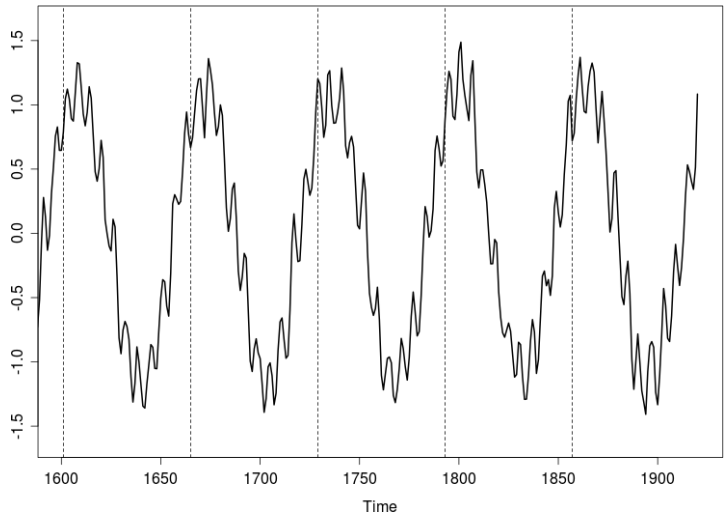


FIGURE 4 – Derniers blocs du signal simulé.

vectorielles de chaque segment, nous pouvons approcher les trajectoires en considérant le développement d'ondelettes avec une troncature appropriée (voir (6.3)). Ainsi, l'ensemble des observations est une approximation de $Z_1(t), \ldots, Z_{30}(t)$ avec $t \in [0, \delta]$.

Pour évaluer la performance prédictive, l'ensemble de 30 observations est divisé en deux. Une première partie, formé par les 25 premières observations, est destinée à calculer la largeur de fenêtre h_n. Pour ce faire, nous suivons Antoniadis et al. (2009). Les auteurs démontrent que la valeur optimale peut se trouver par la minimisation d'une fonction de risque empirique. Nous utilisons la fonction de risque quadratique suivante

$$R_n(h) = \sum_{i=i_0}^{i_1} \sum_{j=1}^{N} (\widehat{Z_{i,h}}(t_j) - Z_i(t_j))^2,$$

où $\widehat{Z_{i,h}}$ est la prévision de Z_i à partir des données $Z_1, \ldots, Z_{i-1}$ et en utilisant la largeur de fenêtre $h_n = h$. Pour une grille de valeurs possibles, nous présentons dans la Figure 5 la fonction $R_n(h)$ calculée à partir des données simulées pour $i_0 = 15$ et $i_1 = 24$. L'allure obtenue met en évidence qu'un choix de h trop petit ou trop grand produira des erreurs notablement plus grandes. La valeur qui minimise R_n est $h = 0.652$.

Nous évaluons la qualité de la prévision à horizon $n + 1$ pour les segments Z_{26} à Z_{30}. Les critères de qualité de la prévision sont classiques est donnés par :

RMSE : la racine de la moyenne des erreurs de prédictions au carré :

$$\text{RMSE} = \left\{ \frac{1}{N} \sum_{i=1}^{N} (\widehat{Z_{l_0}(t_i)} - Z_{l_0}(t_i))^2 \right\}^{1/2},$$

MAPE : la moyenne des valeurs absolues des erreurs de prédictions relatives :

$$\text{MAPE} = \frac{1}{N} \sum_{i=1}^{N} \left| \frac{\widehat{Z_{l_0}(t_i)} - Z_{l_0}(t_i)}{Z_{l_0}(t_i)} \right|,$$

où $\widehat{Z_{l_0}}$ est la prévision de Z_{l_0} avec les observations $\{Z_i\}$ et $i = 1, \ldots, l_0 - 1$. Le dernier bloc simulé et prévu est présenté dans la Figure 6. La prévision suit de près la forme de

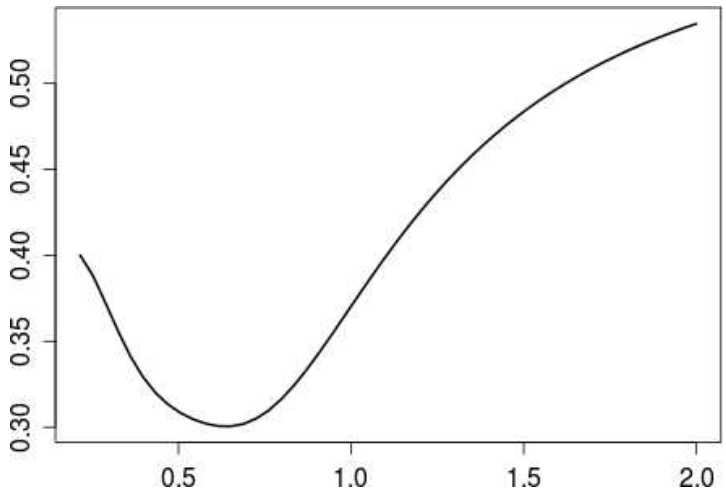


FIGURE 5 – Fonction de risque empirique pour la calibration de la fenêtre.

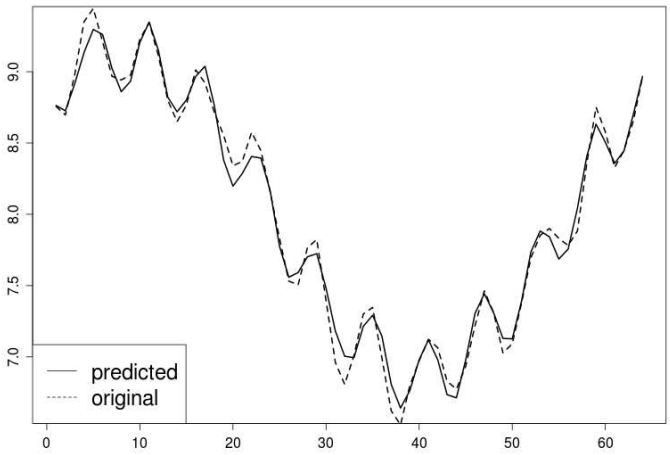


FIGURE 6 – Dernier bloc simulé et sa prévision par la méthode KWF.

Bloc	MAPE	RMSE
26	0.13	1.33
27	0.11	1.15
28	0.12	1.16
29	0.12	1.18
30	$9{\cdot}10^{-2}$	0.91
Moy	0.11	1.15

TABLE 1 – Erreurs de prévision en termes de MAPE et RMSE pour la trajectoire simulée.

la courbe autant pour la composante prédominante que pour la secondaire. Le même comportement est observé pour les autres blocs.

La qualité de prévision en termes de MAPE et RMSE des blocs prévus est présenté dans la Table 1. Les niveau d'erreur sont du même ordre de grandeur que ceux présenté dans l'article de Antoniadis et al. (2006). Plus de comparaisons avec d'autres méthodes de prévision non paramétrique ou paramétrique peuvent se trouver dans cet article. Finalement, nous voudrons remarquer que malgré l'historique relativement court, les résultats de prévision sont très intéressants. Nous avons répété l'expérience avec un historique plus long (environ 300 blocs) avec lequel on a obtenu un niveau moyen de MAPE sensiblement inférieur à 1%.

8.2 Données réelles de consommation.

8.2.1 Brève description des données.

La demande d'électricité dépend de la structure socio-économiques ainsi que des conditions climatiques. Les données contiennent des structures saisonnières correspondant à la structure du calendrier ainsi qu'aux variations annuelles du climat.

Nous commençons par énumérer les caractéristiques indépendantes des conditions climatiques. Dans le premier graphique de la Figure 7 nous pouvons observer l'évolution sur le long terme de la demande nationale d'électricité en France. Malgré la crise économique des derniers ans, nous remarquons une tendance linéaire ascendante. Le cycle annuel est aussi clairement marqué, présentant les plus importants niveaux de consommation d'électricité pendant l'hiver.

Lorsque nous faisons un focus, nous pouvons distinguer une périodicité hebdomadaire comme en témoigne le graphique au milieu de la Figure 7. Le profil économique des jours ouvrés et des week-ends est reproduit par la demande avec une forte hausse durant les jours ouvrés. Il existe d'autres artefacts de l'activité socio-économique. À titre d'exemple, durant la période estivale, nous observons deux semaines durant lesquelles la demande en électricité est extrêmement basse, correspondant aux vacances d'été. Il est à noter aussi que le profil de la demande d'électricité en hiver est plus complexe à cause d'une grande variation de la demande. L'impact de ce fait sur la prévision est direct, produisant une net dégradation de la performance de prévision avec des les niveaux d'erreur les plus grandes sur l'hiver. Malheureusement, cela arrive durant la période où les erreurs de

prédiction ont un coût plus élevé pour les fournisseurs d'électricité.

Le dernier graphique de la Figure 7 représente une courbe journalière. Notons que même à cette résolution temporelle, nous pouvons identifier des motifs : la consommation d'électricité est plus faible la nuit, elle augmente entre 5 heures et 9 heures du matin, il existe un pic que consommation d'électricité en fin d'après-midi, etc. Ces caractéristiques sont identifiables sur chaque courbe journalière. En effet, les jours pour lesquels il est difficile d'avoir une bonne prédiction de la consommation/demande d'électricité sont ceux avec des caractéristiques différents. De manière générale, ces jours sont aussi ceux qui coûtent le plus en termes d'erreurs de prédiction.

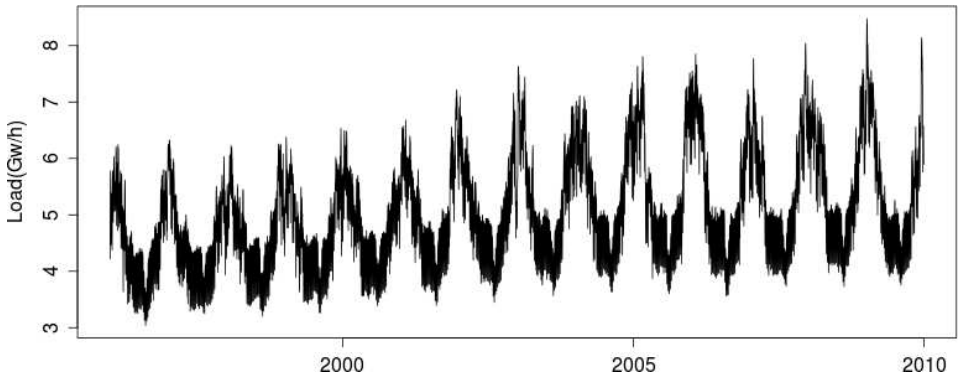
Nous décrivons à présent la partie de la consommation électrique qui dépend des conditions climatiques. En effet, nous nous centrons uniquement sur la dépendance vis-à-vis la température. Cependant, d'autres phénomènes météorologiques -tels la couverture nuageuse- ont aussi une incidence sur la demande d'électricité. Dans le cas français, cette demande est connue pour être hautement thermosensible (*cf.*. 8). Deux faits sont à remarquer. D'une part, la température utilisée dans le graphique est construite de manière artificielle en faisant une moyenne des températures records de quelques stations météorologiques françaises. Rappelons que la sensibilité de la demande d'électricité à la température s'exprime notamment dans le chauffage des bâtiments. Ainsi, un obstacle additionnel s'y rajoute, la température à utiliser doit incorporer l'inertie thermique des bâtiments. D'autre part, la dépendance est très complexe et a certainement un comportement non linéaire et asymétrique par rapport aux hautes et basses températures.

Une très précise quantification de la part des experts d'EDF montre qu'entre deux seuils de température (situés à 14 et de 23°Celsius), la demande d'électricité est insensible aux changements de température. En ce qui concerne les températures inférieures à 14°C ou supérieures à 23°C, les systèmes de chauffage et climatisation respectivement se mettent en marche.

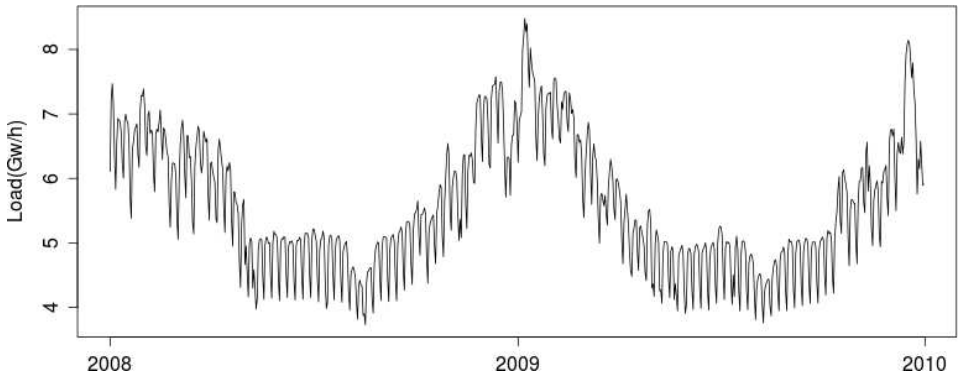
A mode synthèse, disons que ces caractéristiques expliquent la plus part de la variabilité de la consommation d'un jour typique. En d'autres mots, tous modèle de prédiction qui prenne en compte ces éléments aura un performance moyen acceptable. La difficulté réel se trouve dans la prévision de jours de consommation atypique comme pour exemple les jours fériés, les jours de pont associées à un long week-end, des jours de tarifications spéciale, etc. D'autres phénomènes externes peuvent modifier la distribution ou la demande d'électricité, e.g. une crise économique ou des damages produits par des oranges. Les modèles opérationnels à EDF sont capables de gérer la prévision de jours typiques et la plus part de jours atypiques. Ils sont pour la plus part d'entre eux de modèles paramétriques incorporant une régression non linéaire pour quantifier la sensibilité de la consommation à la température ressentie. Le niveau d'erreur moyen dans une année est d'environ 1.5% de MAPE.

8.2.2 Prévision par la méthode KWF.

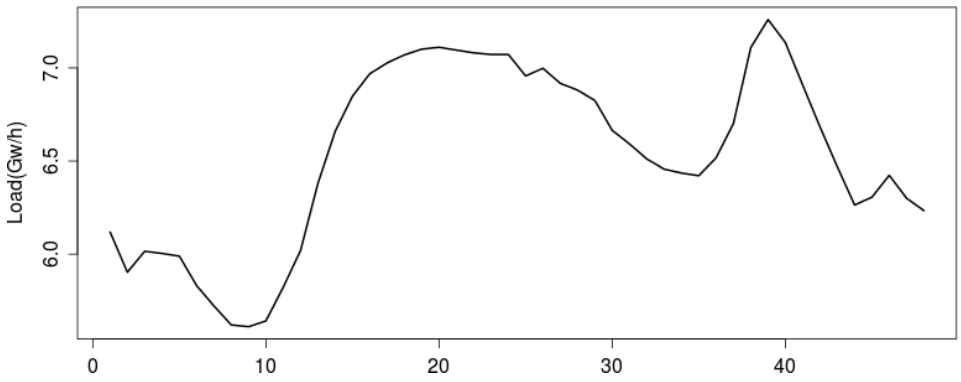
Dans l'article original, la technique proposée a été testée sur la consommation d'électricité française pendant l'été de l'année 1991. C'est-à-dire, sans effet de l'aléa climatique car il n'y a pas de chauffage et quasiment pas de climatisation à l'époque. Les auteurs montrent que leur méthode est meilleure que des modèles SARIMA, le lissage par le filtre Holt-Winters ou une modélisation par des processus autorégressifs hilbertiens



(a) Le cycle annuel.



(b) La périodicité hebdomadaire.



(c) La courbe de charge journalière.

FIGURE 7 – Caractéristiques remarquables de la consommation d'électricité indépendantes des conditions climatiques. Dans les deux premiers graphiques la consommations est représenté par de moyennes journalières, dans le dernier la courbe représenté une consommation d'automne échantillonnée chaque 30 minutes.

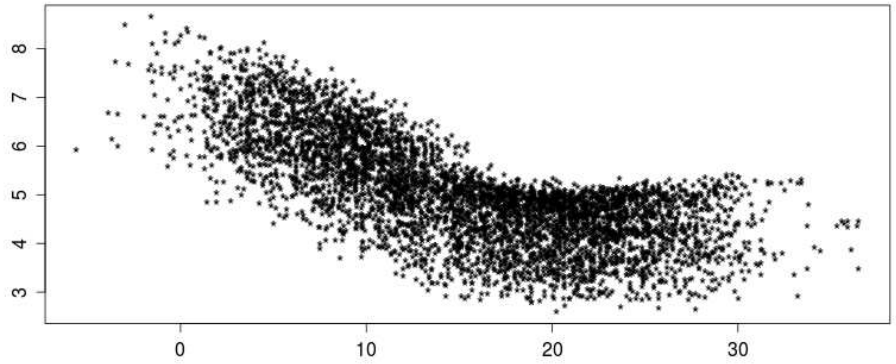


FIGURE 8 – Electricity demand (in Gw/h) as function of temperature (in ° Celsius).

d'ordre 1 (ARH(1)). Dans le but de tester notre propre méthode, nous avons commencé par étudier la performance sur un été plus récent (été 2007-08). Les résultats en performance sur la prévision à pas journalier sont très similaires à ceux des auteurs (une valeur de MAPE d'environ 2%).

Le deuxième pas a été d'élargir la période d'essai à toute une année. Pendant l'été le comportement de la consommation d'électricité est assez stable. La haute thermosensibilité de la demande d'électricité ne s'exprime clairement que quand la température descend au niveau du seuil de déclenchement des systèmes de chauffage. Ce sont justement pour des changements brusques de température, ou pour les jours fériés que la méthode devra montrer son potentiel.

Pour l'expérience, nous avons utilisé un signal contenant des mesures en temps réel postérieurement consolidées de la consommation électrique. Les données sont échantillonnées chaque 30 minutes depuis le 1 septembre 1996 jusqu'au 31 août 2006. Avec ces données, nous obtenons des blocs de taille $\delta = 48$ points représentant les courbes de charge journalière entre minuit et 23h30. Nous divisons la base en deux parties : une pour estimer la largeur de fenêtre h (jusqu'au 31 août 2005), l'autre pour mesurer la performance en prévision à pas journalier (du 1 septembre 2005 jusqu'au 31 août 2006). Chaque vecteur de longueur 48 est interpolé à 2^6 points et transformé à l'aide de la DWT pour un niveau de résolution de l'approximation de $j_0 = 0$ en utilisant l'ondelette *Symmlet 6* pour l'AMR. Quant au calcul de h, nous avons testé deux options introduites précédemment : calculer un seul h_n une seul fois (FIX) ou la mettre-à-jour avant chaque prévision (DYN). Le noyau utilisé est le gaussien.

Les résultats des performances de prévision sont présentes dans le tableau 2 en termes de RMSE (en MW/H) et MAPE (en %) pour les variantes FIX et DYN concernant le calcul de la fenêtre. Nous sommes intéressés par la qualité de la prévision globale mais aussi par la qualité par type de jour et la distribution des erreurs de prévision tout au long de l'année. Ce tableau sera le point de départ pour la comparaison de la méthode KWF de base originale et les alternatives que nous utiliserons.

De façon générale, les niveaux d'erreur global de la technique sont plus importants que les niveau observées pendant la période estivale (environ quatre fois plus grandes). Avant d'essayer d'expliquer pourquoi, nous nous centrons sur les différences entre les variantes

Bloc	MAPE	RMSE	MAPE	RMSE
lundi	7.1	4 323	7.48	4 556
mardi	8.11	5 241	8.36	5 393
mercredi	7.63	4 913	7.94	5 097
jeudi	8.03	5 140	8.28	5 294
vendredi	8.72	5 590	8.87	5 676
samedi	7.62	4 290	7.58	4 288
dimanche	8.51	4 522	8.66	4 604
férié	12.9	6 370	12.85	6 332
Global	8.11	4 907	8.31	5 028

TABLE 2 – MAPE et RMSE par type de jour et global pour la période 1/9/05 - 31/08/06. Le paramètre h_n à été réglé selon la méthode FIX (à gauche) avec $h_n = 2155$, et la méthode DYN où h_n est calculée jour après jour (à droite).

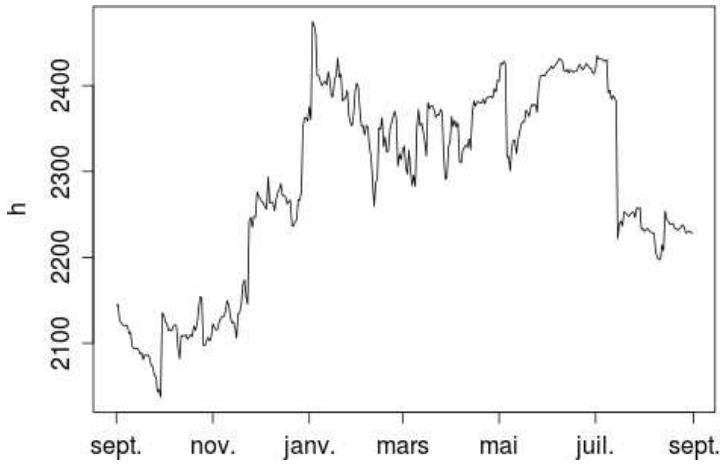


FIGURE 9 – Évolution du paramètre h_n.

pour le calcul de h. Nous pouvons noter que la méthode FIX a une performance supérieure pour chaque type de jour (sauf les samedi) que la méthode DYN pour la moyenne annuelle. L'évolution de la fenêtre mobile calculée par la méthode DYN est affichée dans la figure 9. Nous pouvons noter une forte et rapide augmentation du paramètre à partir du mois de novembre, après un saut à fin octobre dû au changement de l'horaire d'hiver. La fenêtre s'ouvre pour calibrer l'effet des jours fériés et les premiers jours froids qui ont tous les deux une dynamique de consommation très différente aux autres types de jours. Cette introduction provoque des fortes perturbations qui altèrent de manière importante la performance de la méthode. Nous continuerons à examiner cette dichotomie (fenêtre fixe - fenêtre mobile) dans les prochains chapitres.

En ce qui concerne les hauts niveaux d'erreur dans la prévision, nous rappelons que nous n'avons introduit aucune information exogène. La prévision est faite par régression sur les blocs observés du passé. Nous n'avons pas d'informations sur l'effet sur la consommation d'électricité provoquées par la structure calendaire ou l'effet des variables météorologiques (e.g. la température ou la nébulosité). Toutefois il est souhaitable d'avoir des niveaux d'erreurs globalement faibles et d'analyser sa distribution dans l'année. Pour approfondir cette analyse, nous présentons dans la Figure 10 l'évolution du MAPE par

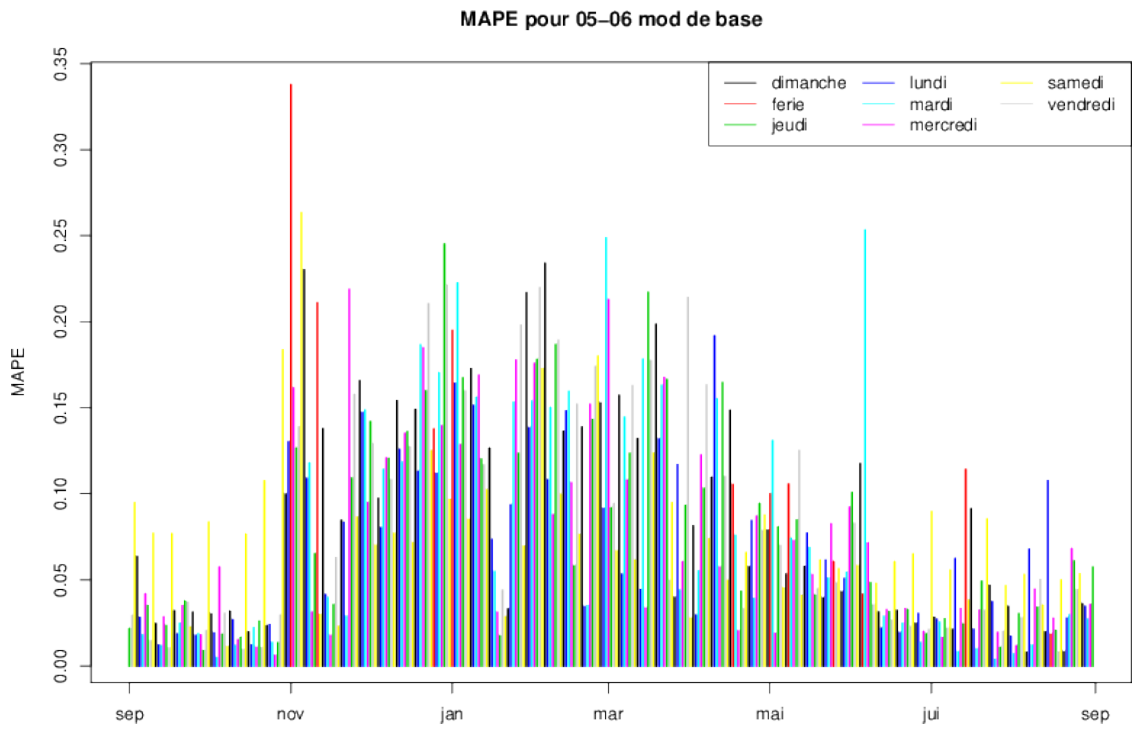


FIGURE 10 – MAPE par jour selon type de jour pour la méthode de BASE.

jour selon type de jour. À partir de ce graphique on s'aperçoit de deux grands problèmes qui concernent la stationnarité du processus Z :

- la méthode conduit à la pire des performances quand le niveau est plus difficile à prévoir (pendant l'hiver), et
- quand le niveau est assez stable (voir entre juin et octobre), les barres qui représentent les samedis sont sensiblement plus hautes.

Analysons ces deux points. Pour le premier, il est assez clair que l'hypothèse de stationnarité du processus n'est pas valide pour les données de consommation d'électricité. La méthode de base est fondée sur un processus de moyenne nulle. C'est justement quand le niveau varie de manière plus importante que nous observons les erreurs les plus grandes. Au sujet du deuxième problème, la méthode ne peut pas faire la distinction à partir des données réelles de consommation entre les différents types de jours qui sont déterminées par la structure du calendrier. Les prévisions sont donc contaminées par la structure de la consommation de tous les jours. A titre d'exemple, si on doit prévoir un samedi à partir d'un vendredi, la ressemblance des vendredis aux lundis, mardis, mercredis et aux autres vendredis fait que le samedi est prédit par un mélange de jours de la semaine et de jours du week-end. Il serait souhaitable de ne prévoir un samedi qu'à partir de samedis.

Dans la prochaine section nous essayons de mettre en oeuvre des corrections au modèle de base KWF tenant compte des points ci-dessus : un niveau moyen qui évolue tout au long de l'année et la présence de groupes de jours.

9 Gérer la non stationnarité.

Le but de ce chapitre est d'explorer des pistes pour corriger les problèmes trouvés dans l'application de la méthode de prévision fonctionnelle de base KWF que nous venons d'explorer. D'abord, nous traitons le problème d'un niveau moyen non constant par un centrage de courbes. Ensuite, nous travaillerons sur l'existence de groupes de segments.

9.1 Centrage de courbes

Les deux phases de la méthode ont un rôle différent face au centrage de courbes. Dans la première phase de la méthode, la distance proposée consiste à centrer les données de façon implicite, car elle est n'est calculée que sur l'ensemble des coefficients d'ondelettes. En fait, les effets d'échelles ne sont pas pris en compte ce qui conduit à un centrage implicite.

Cependant, dans la deuxième phase le centrage doit être fait explicitement. La deuxième phase consiste à construire la prévision par une moyenne pondérée des blocs du passé. Le fait de mélanger les blocs qui présentent de niveaux moyens très différents peut introduire des effets non souhaitables. De ce fait nous allons procéder comme suit : nous allons utiliser la méthode de base pour prévoir les détails de la courbe de charge ; puis nous allons prévoir la partie d'approximation. Une fois les deux éléments déterminés (détails et approximation) la transformée inverse en ondelettes permettra d'obtenir la prédiction du bloc futur.

Pour faciliter la discussion nous allons noter les parties approximation et détails de chaque fonction comme $\mathcal{S}_i(t)$ et $\mathcal{D}_i(t)$ respectivement, donc

$$Z_i(t) = \underbrace{\sum_k c_{j_0,k}^{(i)} \phi_{j_0,k}(t)}_{\mathcal{S}_i(t)} + \underbrace{\sum_{j \geq j_0} \sum_k d_{j,k}^{(i)} \psi_{j,k}(t)}_{\mathcal{D}_i(t)} .$$

Par exemple, si le processus Z est centré on a $Z(t) = \mathcal{D}(t)$ et nous sommes sous l'hypothèse de la méthode de base.

Nous sommes intéressés ici par le cas où les fonctions $\mathcal{S}_i(t)$ ne sont pas nulles. En particulier, nous allons travailler principalement dans le cas où $j_0 = 0$ (e.g. l'approximation à la résolution la grossière.). Dans ce cas, les fonctions d'approximation sont constantes et proportionnelles au niveau moyen de chaque fonction. De plus, nous pouvons regarder la suite $\{c_{j_0,k}^{(i)}\} = \{c_{j_0}^{(i)}\}$ avec $i = 1, \ldots, n$ comme une série temporelle unidimensionnelle.

En définitive, la prévision pour le segment $n+1$ s'écrit

$$\widehat{Z_{n+1}(t)} = \widehat{\mathcal{S}_{n+1}(t)} + \widehat{\mathcal{D}_{n+1}(t)}.$$

La processus fonctionnel $\mathcal{D}_{n+1}(t)$ est centré. De ce fait nous pouvons utiliser la méthode de base pour obtenir une prévision

$$\widehat{\mathcal{D}_{n+1}(t)} = \sum_{m=1}^{n-1} w_{m,n} \mathcal{D}_{n+1}(t).$$

Ensuite, pour la prévision de $\mathcal{S}_{n+1}(t)$ nous allons explorer ces pistes :

BASE La première variante utilise les mêmes poids calculés à partir de la ressemblance entre les détails des courbes. Le niveau se prévoit par $\widehat{\mathcal{S}_{n+1}(t)} = \sum_{m=1}^{n-1} w_{m,n}\mathcal{S}_{m+1}(t)$. Cette méthode a l'avantage d'être simple et d'être une extension naturelle de la méthode originale. En fait, elle revient à ne pas centrer lors de la deuxième phase de la méthode. Nous écrivons la prévision globale

$$\boxed{\widehat{Z_{n+1}(t)} = \sum_{m=1}^{n-1} w_{m,n}\mathcal{S}_{m+1}(t) + \sum_{m=1}^{n-1} w_{m,n}\mathcal{D}_{m+1}(t)}.$$

PRST Une alternative simple est de considérer un modèle de persistance. Si nous supposons que le niveau moyen pour le lendemain est le même que celui d'aujourd'hui on peut écrire $\widehat{\mathcal{S}_{n+1}(t)} = \mathcal{S}_n(t)$. Finalement la prévision pour le segment Z est

$$\boxed{\widehat{Z_{n+1}(t)} = \mathcal{S}_n(t) + \sum_{m=1}^{n-1} w_{m,n}\mathcal{D}_{m+1}(t)}.$$

Après utilisation de cette alternative nous avons constaté que ce n'est pas forcement le niveau de la veille la meilleure référence pour corriger la moyenne. Bien que ce soit vrai pour les jours de la semaine, pour les jours de week-end la meilleure référence est celle de la semaine antérieure. Nous utilisons donc, la stratégie hybride de choisir pour les jours de la semaine le niveau de la veille et pour les jours week-end le niveau de la dernière semaine.

DIFF Dans l'article de Poggi (1994) sur la prévision non paramétrique de la consommation électrique en tant que processus multivarié, l'auteur propose de corriger le niveau moyen de la prévision en considérant la transition des niveaux moyens entre un jour et son lendemain. Dans notre notation, cela devient à prévoir le niveau moyen à partir du niveau du jour témoin plus une correction par différences premières des niveaux du passé : $\widehat{\mathcal{S}_{n+1}(t)} = \mathcal{S}_n(t) + \sum_{m=1}^{n-1} w_{m,n}\Delta(\mathcal{S}_n)(t)$. Nous pouvons alors écrire la prévision de Z

$$\boxed{\widehat{Z_{n+1}(t)} = \underbrace{\mathcal{S}_n(t) + \sum_{m=2}^{n-1} w_{m,n}\Delta(\mathcal{S}_n)(t)}_{\widehat{\mathcal{S}_{n+1}(t)}} + \sum_{m=1}^{n-1} w_{m,n}\mathcal{D}_m(t)}.$$

SAR Finalement nous exploitons le fait d'avoir dans la série des niveaux moyens $\{c_{j_0}^{(n)}\}_n$ est unidimensionnelle. Grâce aux ondelettes cela est plus simple que sur la série originale. Nous allons utiliser une modélisation classique des séries temporelles du type Box-Jenkins par exemple. Cette voie apporte une alternative paramétrique mais aussi rend plus complexe la prévision car il faut régler deux modèles qui ne "communiquent pas entre eux". Nous avons testé plusieurs variantes. Le modèle retenu est assez simple. Il consiste en une modélisation SAR avec 2 coefficients saisonniers (hebdomadaires) et quatre journaliers.

9.1.1 Résultats de la correction par niveau

Nous reprenons les données de la consommation d'électricité pour tester les corrections que nous venons de décrire. Dans un premier temps nous nous concentrons sur la

Bloc	BASE	PRST	DIFF	SAR
lundi	7.1	4.75	2.07	3.89
mardi	8.11	4.49	2.11	3.63
mercredi	7.63	2.56	2.1	2.51
jeudi	8.03	2.06	1.97	2.43
vendredi	8.72	2.59	2.06	2.84
samedi	7.62	5.88	6.94	4.18
dimanche	8.51	5.14	1.51	2.26
férié	12.9	14.74	9.85	13.54
Global	8.11	4.24	2.91	3.42

TABLE 3 – MAPE par type de jour et global pour la méthode de base BASE, et les variantes de centrage : PRST le niveau moyen est mis au niveau du jour témoin, DIFF le niveau moyen est une moyenne pondérée des premiers différences des niveaux) et SAR modèle autorégressif saisonnier. La valeur optimale de h_n pour l'estimation FIX de la fenêtre est 2173.

Bloc	FIX	DYN
lundi	2.07	2.09
mardi	2.11	2.12
mercredi	2.1	2.13
jeudi	1.97	1.94
vendredi	2.06	2.09
samedi	6.94	7.05
dimanche	1.51	1.52
férié	9.85	10
Global	2.91	2.94

TABLE 4 – MAPE par type de jour et global pour la méthode DIFF. La valeur de h_n est calculée par la variante FIX à gauche et par DYN à droite.

comparaison des différentes corrections par centrage en ne regardant que les résultats à fenêtre fixe. Le Tableau 3 contient ces résultats.

Les résultats montrent une amélioration avec l'option globale (corriger par niveau moyen) vis-à-vis l'option de ne pas corriger (méthode de BASE). Parmi les options présentées, la correction par différences premières DIFF s'avère la plus performante pour tous les types de jour sauf les jours samedis. Comme avant, nous sommes aussi intéressés par la distribution des erreurs dans l'année. L'évolution du MAPE par jour selon type de jour est présentée dans la Figure 11. Par rapport à la figure 10 nous nous apercevons de deux faits marquants : d'une partie les plus hauts niveaux d'erreurs pendant l'hiver se sont vu réduits, d'une autre partie les jours fériés (en rouge) et les samedis (en jaune) présentent les erreurs les plus importantes.

Pour comparer les performances entre l'utilisation de la fenêtre FIX et la fenêtre DYN, le tableau 4 montre que l'utilisation d'une largeur de fenêtre qui est recalculée pour chaque jour entraîne une très légère dégradation globale de la méthode qui pourrait

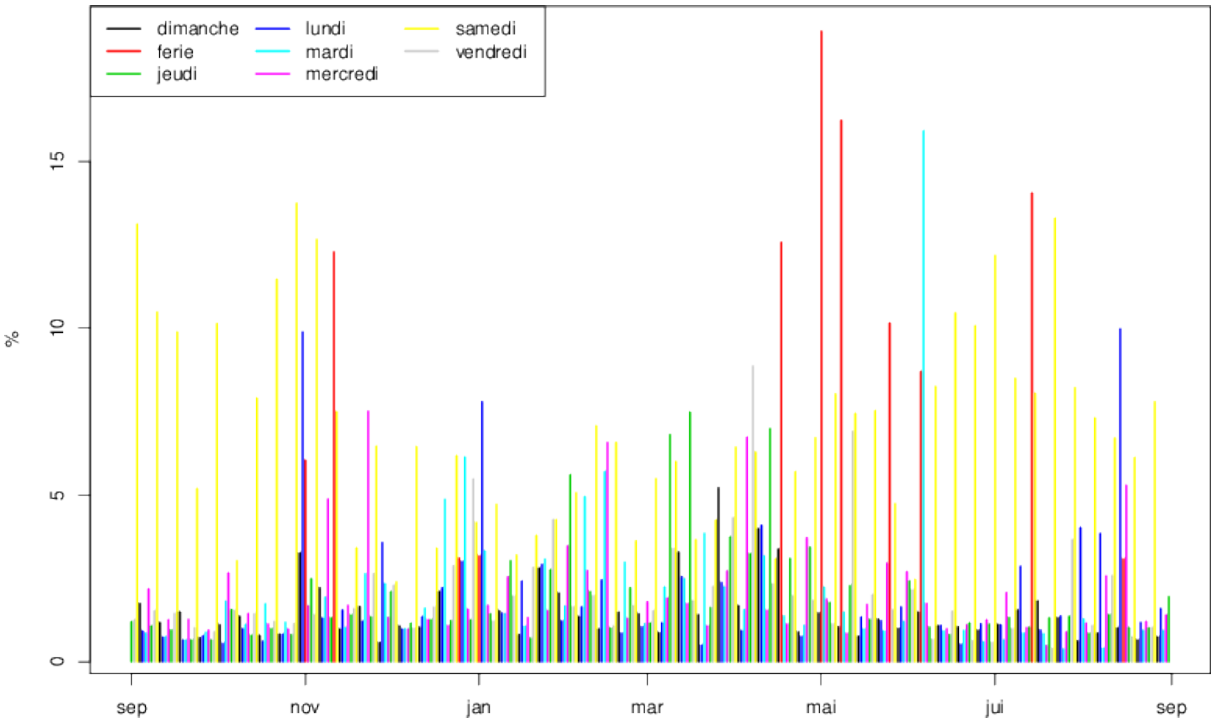


FIGURE 11 – MAPE par jour selon type de jour pour le modèle avec centrage DIFF

.

ne pas être significative. En plus, la dégradation n'est pas constante. En effet, pour les premiers deux mois de l'étude, la méthode à fenêtre mobile a un MAPE moyen de 2,40% contre 2,42% pour la fenêtre fixe. Nous observons donc la même structure que celle que nous avons vue pour la prévision sans correction par niveau. Lorsque des jours avec une dynamique de consommation très différente (comme les jours fériés et les premiers jours de froid) entrent dans l'historique récent du prédicteur, la performance de la prévision affiche une dégradation plus importante pour la méthode qui s'adapte plus rapidement aux changements.

En conclusion, nous avons réussi à contrôler le problème du niveau variable. Le centrage de chaque courbe dans la construction du prédicteur plus la prévision du niveau par différences premières s'avère une solution prometteuse. Nous avons gagné en performance globale. En effet nous sommes déjà dans des niveaux d'erreurs de prévision qui ne sont pas très éloignés de ceux que l'on trouve dans les modèles opérationnels (particulièrement pour les dimanches).

Il reste encore à travailler sur le problème des samedis. Comme nous avons déjà dit, la transition de jours de semaine à un jour du weekend est une information que nous n'introduisons pas dans le modèle. Néanmoins, c'est une information que nous avons au moment de faire la prévision et que nous pouvons incorporer à notre prédicteur. Nous allons aborder ce problème. Désormais nous présenterons les prévisions uniquement avec la correction par niveau DIFF.

9.2 Correction par groupes.

L'éventuelle existence de groupes de jours est le deuxième point à traiter pour rendre plus raisonnable l'hypothèse de stationnarité. Nous avons abordé le problème des niveaux différents dans la suite d'approximations $\{\mathcal{S}_n\}_n$. Nous nous concentrons maintenant dans la série de détails $\{\mathcal{D}_n\}_n$. L'ensemble des détails contribuent à l'allure de la courbe. Il est bien connu que la forme de la courbe journalière dépend du calendrier : les jours de week-end présentent des courbes très différentes de celles des jours de semaine car l'activité des consommateurs change. La forme de la courbe dépend aussi de variables climatiques comme la température ou la nébulosité entre autres.

L'idée sous-jacente est que nous pensons pouvoir renforcer l'hypothèse de stationnarité de la suite une fois que nous aurons conditionné par une variable de groupe de jours. Le reste de ce chapitre est destiné à la recherche de ces groupes de jours.

Groupes de jours.

Nous avons décidé de commencer par l'utilisation de groupes déterministes. L'option la plus simple est d'utiliser les jours de la semaine comme groupes. La connaissance acquise par les experts d'EDF sur la courbe de charge nous amène aux groupes suivants : {lundi}, {mardi, mercredi, jeudi}, {vendredi}, {samedi}, {dimanche} et {fériés}. En effet, les jours du milieu de la semaine ont une forme très similaire. Ceci ce différencie de la forme de jours lundi notamment dans la montée du matin, et de la forme de jours vendredi dans la forme de l'après midi.

Bien que ce groupement soit simple, il ne prend en compte que la classification du jour témoin. Or, les jours fériés ne seront jamais bien prédits car rien dans la veille n'informe la méthode que le jour à prévoir est particulier. D'ailleurs, nous avons remarqué l'importance d'inclure l'information des transitions entre jours quand nous avons développé la variante de centrage par différences premières. Nous voulons continuer à explorer cette voie, ici par l'incorporation du type de transition entre les jours à prévoir et leur veille. La classification de jours selon les transitions possibles est l'objet de la Figure 12.

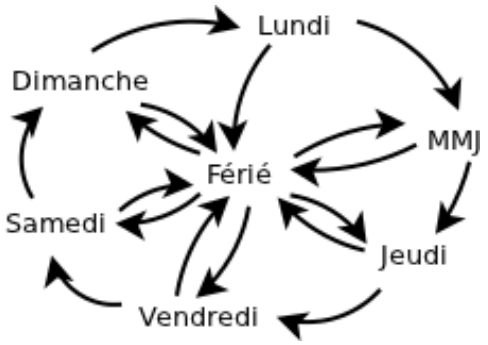


FIGURE 12 – Les possibles transitions entre types de jours. Le sigle MMJ est utilisé pour indiquer le groupe formé par les jours mardi, mercredi et jeudi.

L'incorporation de l'information des groupes dans l'étape de la prédiction se fait avec un traitement dans le calcul de la ressemblance. Nous redéfinissons notre vecteur poids

$w_{m,n}$ entre les jours n et m $\forall m = 1, \ldots, n-1$

$$\tilde{w}_{m,n} = \begin{Bmatrix} w_{w,m} & \text{si} & gr(m) = gr(n) \\ 0 & \text{sinon} & \end{Bmatrix},$$

où $gr(n)$ indique le groupe du n-ème jour. Donc nous mettons à zéro la valeur de l'indice pour tous les jours m qui n'appartiennent pas au même groupe que le jour n.

9.2.1 Résultats de la correction par niveau et par groupes de jours.

Nous affichons les résultats des performances de prévision pour les données réelles de la consommation d'électricité dans le tableau 5. La prévision est faite par correction de niveau avec le centrage DIFF et avec correction de groupes par le traitement que nous venons d'exposer. Pour le calcul de la largeur de fenêtre, nous nous centrons dans un premier temps sur les résultats de performance de prévision pour la fenêtre estimée par la méthode FIX. Remarquons que l'introduction de groupes de jours a rendu plus complexe le modèle et maintenant nous avons une fenêtre par type de groupe.

Nous observons une amélioration globale pour chacune des méthodes de correction par groupes. Cependant, l'amélioration n'est pas observée pour tous les type de jour. La méthode qui utilise les groupes issus du classement calendrier (CALEN) présente de dégradations pour les jours lundi, dimanche et fériés. Toutefois, il faut remarquer que nous avons réussi à corriger les hauts niveaux d'erreur pour les jours samedi. Nous pouvons expliquer ceci par la remarque de la section précédente. Quand la méthode tient compte du fait que le jour témoin est un vendredi, elle empêche d'autres jours de semaine dont la forme ressemble au vendredi d'entrer dans la prévision et ne prévoit le samedi qu'avec des jours samedi.

Cependant, l'utilisation de groupes a réduit le nombre effectif d'observations par type de jour. Ceci produit un dégradation de la performance de prévision pour certain types de jours, en particulier ceux qui sont peu nombreux dans l'historique (e.g. les jours fériés). L'utilisation de groupes basés sur les transitions du calendrier ne font que réduire certains de ces groupes, mais en isolant les types de jours correctement nous pouvons attendre une amélioration dans la prévision. C'est en effet le cas quand nous utilisons les groupes issues des transitions du calendrier (CALEN-TR). En fait, l'amélioration provient du fait que malgré l'augmentation de la variance du prédicteur, la réduction du biais fait plus que compenser cet effet.

Une conséquence latérale de l'introduction de groupes a été la réduction du temps de calcul. La méthode de base KWF et les variantes issues de la correction par niveau ont pris environ 1.5 heure pour obtenir les prévision de toute l'année. Ce temps de calcul est réduit à moins de 5 minutes quand des groupes sont introduits.

Nous passons maintenant à la comparaison avec le tableau à droite (Table 5). Ce dernier est la version avec la fenêtre DYN. Nous observons ici une performance supérieure de la version à fenêtre mobile par rapport à l'alternative FIX. Dans la Figure 13 nous montrons l'évolution de la fenêtre selon les groupes de jours. Les lignes correspondent à des trajectoires lissées de la fenêtre en fonction du temps par jour de la semaine. La fenêtre calibrée par la méthode dynamique paraît suivre une évolution stable tout au long de l'année pour la plupart des types de jours. Cependant, les jours lundi affichent

Bloc	DIFF	CALEN	CALEN-TR	CALEN	CALEN-TR
lundi	2.07	2.87	2.11	2.05	1.95
mardi	2.11	1.66	1.66	1.80	1.67
mercredi	2.1	1.66	1.6	1.68	1.54
jeudi	1.97	1.38	1.24	1.28	1.23
vendredi	2.06	1.83	1.66	1.91	1.66
samedi	6.94	1.83	1.57	1.88	1.57
dimanche	1.51	1.6	1.6	1.55	1.55
férié	9.85	10.49	2.59	10.80	3.33
Global	2.91	2.08	1.66	2.00	1.64

TABLE 5 – MAPE par type de jour et global pour la méthode de base BASE, la méthode corrigées par centrage DIFF : sans groupes, avec groupes calendaire CALEN et avec groupes de transition du calendaire. A gauche on utilise la méthode FIX pour calculer h_n par groupe, à droite la méthode DYN.

une augmentation importante autour l'hiver, souligné par une bosse dans le graphique, qui est due à des lendemains des jours fériés d'hiver. Les jours fériés présentent un comportement très différent selon que le jour férié est froid ou chaud. Cela indique qu'il pourrait être pertinent d'inclure deux types de jours fériés. Aussi, dans la trajectoire des jours MMJ nous pouvons remarquer une sous-suite que se différencie à partir de mois d'avril : ce sont les jours mardi qui présentent une évolution différente, la fenêtre s'ouvre un peu plus pour ce type de jours.

L'évolution du MAPE par type de jour pour la prévision avec correction de centrage DIFF et groupes CALEN-TR est présenté dans la Figure 14. Cette évolution est plus homogène dans l'année que celle de la méthode de base (*cf.* Figure 10). Les problèmes qui ont été signalés : niveau moyen qui n'est pas constant et la présence de groupes de jours, semblent être résolus. La plupart des jours présentent encore de hauts niveaux d'erreur sont des jours dans les alentours des jours fériés. Certainement, un traitement plus fin de ce type de jours pourrait conduire à de moindres erreurs.

Nous pouvons nous interroger sur le nombre et le type de blocs que la méthode trouve pour construire la prévision. Le premier indice que nous analysons est la répartition des poids dans le passé. Nous regardons l'exemple de la Figure 15 où nous cherchons les jours qui ressemblent au dimanche 2 juillet 2006. La forme de la répartition montre que les poids des jours croissent à proximité du jour en question. En plus, les poids non nuls sont attribués seulement aux jours qui correspondent à la même position du calendrier que le jour de référence.

Le nombre de jours qui ont un poids non nul dans la prévision, dépend du type de jour et de la position dans l'année comme le montre la Figure 16. Dans le graphique, nous pouvons remarquer la forte décroissance du nombre de segments voisins pendant l'hiver suivie d'une rapide augmentation pour les jours d'été. Ceci reflète la plus importante hétérogénéité des jours d'hiver où l'effet de la température sur la consommation produit des trajectoires plus particulières.

Remarquons dans la fin de cette section que le modèle de prévision utilisé n'incorpore aucune information exogène, sauf la structure du calendrier. Les niveaux d'erreurs dans

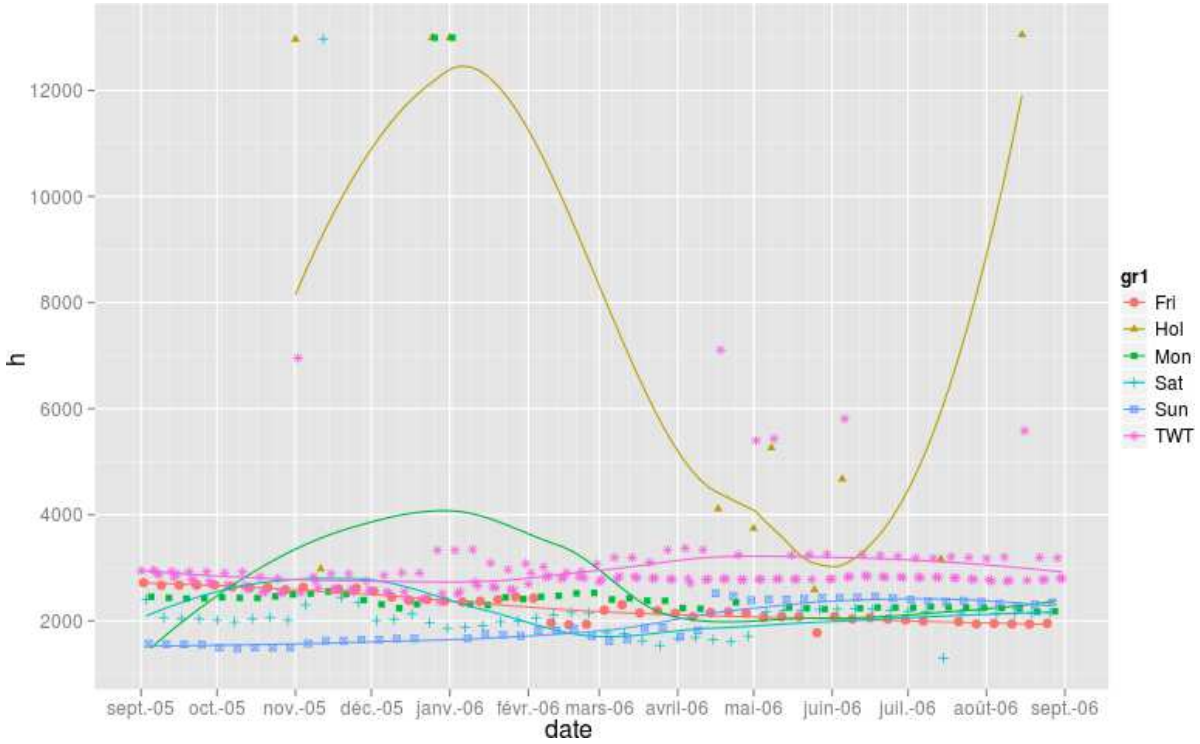


FIGURE 13 – Valeur de fenêtre calibrée par la variante DYN dans la prévision avec centrage DIFF et groupes CALEN-TR.

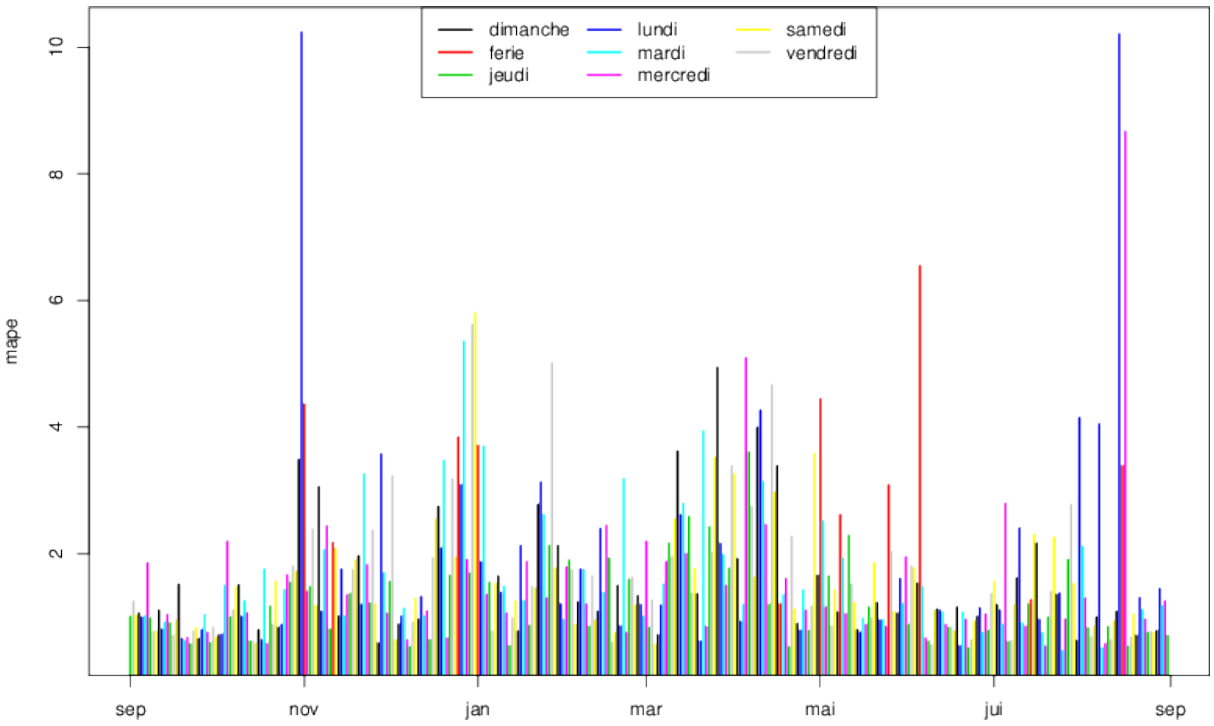


FIGURE 14 – MAPE pour la méthode avec correction par centrage DIFF et correction par groupes CALEN-TR et la fenêtre par DYN.

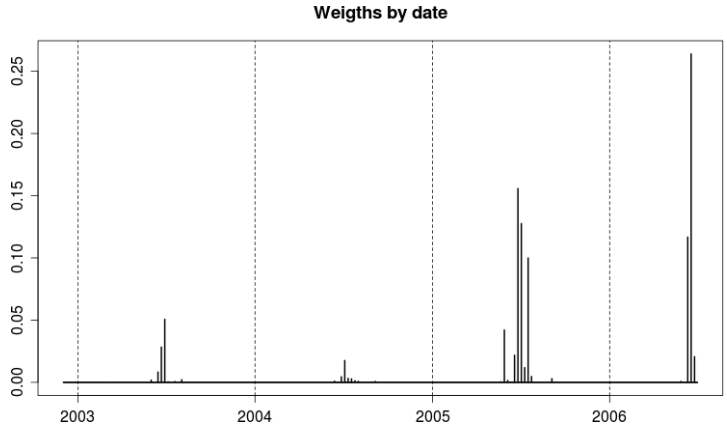


FIGURE 15 – Vecteur de poids pour le dimanche 2 juillet 2006.

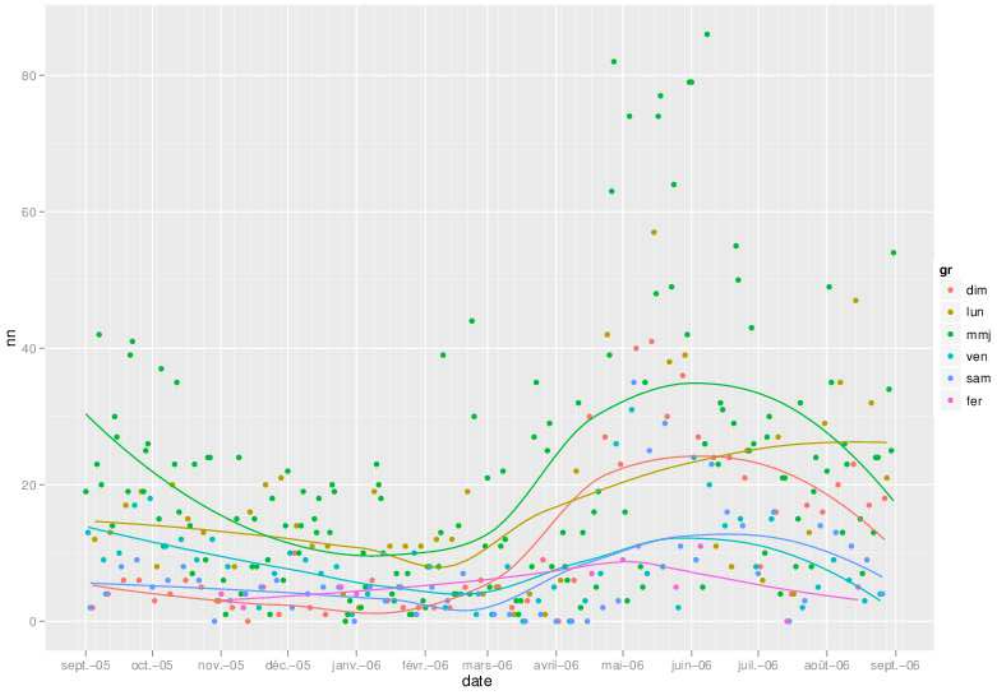


FIGURE 16 – Nombre de jours avec un poids non nul par type de jour pour la plage de prévision.

l'application de la consommation d'électricité sont acceptables notamment pour les jours fériés.

10 Remarques sur la sensibilité du prédicteur aux choix des paramètres de réglage.

Dans la présente section nous présentons des études complémentaires concernant le rôle que jouent les différents paramètres de réglage du prédicteur. L'objectif est d'aller un peu plus loin dans la compréhension de la méthode de prévision.

10.1 Incidence de la DWT.

L'ondelette. Pour tester l'effet du choix de la famille d'ondelettes et la taille du filtre, nous avons utilisé les familles d'ondelettes de Daubechies de phase extrême (DaubExPhase) et le moins asymétriques (DaubLeAsymm). La table suivante résume les tailles des filtres ainsi que l'erreur moyenne de prévision sur l'année 2005-2006.

Famille	Filtre	MAPE	Famille	Filtre	MAPE
DaubExPhase	1	1.677	DaubLeAsymm	4	1.639
	2	1.650		6	1.644
	4	1.677		8	1.649
	6	1.670		10	1.676
	8	1.691			
	10	1.670			

L'incidence de l'ondelette est presque nulle dans la moyenne annuelle (en couleur le choix initial). Toutefois, il semblerait que les ondelettes de phase extrême ont moins de mal pour prévoir de jours fériés. Une remarque similaire peut être faite par rapport à la taille du filtre : en cas de doute, une taille de filtre petite est préférable.

L'interpolation. Les vecteurs de consommation d'électricité ne peuvent pas être utilisés directement dans l'algorithme pyramidal de Mallat pour la DWT car la taille de ($N = 48$) n'est pas une puissance de 2. Pour résoudre le problème nous avons interpolé les données à 2^J points par segment pour $J = 6$. Maintenant, nous utilisons d'autres résolutions d'interpolation. Nous testons la méthode de prévision en utilisant des interpolations à 2^J points pour $J = 4$ et 6. Les résultats en termes de MAPE sont respectivement 1.685 et 1.729 tous le deux supérieurs au niveau moyen annuel trouvé originellement. Cependant, il faut remarquer que la dégradation du niveau d'erreur est plus importante quand nous utilisons une grille plus fine que dans le cas contraire, ce qui n'est pas surprenant car on crée de l'information artificielle.

10.2 Incidence des paramètres de l'estimateur à noyau.

Le noyau. Pour mesurer l'incidence du choix d'un noyau en particulier, nous avons utilisé la méthode de prévision KWF avec centrage DIFF et groupes CALEN-TR et nous avons mis en concurrence différents noyaux. Les noyaux, tous implémentés dans le package `kerwavfun`, sont : l'uniforme, triangulaire, biweight, triweight, Epanechnikov, Gaussien et Cauchy. Ils correspondent à différent degrés de régularité (triés de manière croissante). Habituellement, c'est le noyau Gaussien qui est utilisé.

Les résultats en qualité de prévision moyenne annuelle sont résumés dans la table suivante

Jour	Uniforme	Triangul.	Biweight	Triweight	Epanech.	Gaussian	Cauchy
lundi	1.89	1.7	1.68	1.68	1.89	1.74	2.22
mardi	1.52	1.46	1.41	1.4	1.52	1.41	2.08
mercredi	1.36	1.26	1.24	1.24	1.36	1.25	1.91
jeudi	1.85	1.71	1.69	1.68	1.85	1.64	2.12
vendredi	1.94	1.83	1.8	1.77	1.94	1.55	2.17
samedi	1.97	1.88	1.84	1.8	1.97	1.62	2.17
dimanche	2.37	2.19	2.13	2.11	2.37	2.05	2.71
férié	4.41	3.41	3.25	3.4	4.41	2.59	2.96
Global	1.92	1.77	1.74	1.72	1.92	1.64	2.25

Nous pouvons remarquer que le noyau gaussien est meilleur que les noyaux à support compact et que le noyau à queues épaisses de Cauchy. Ce remarque est aussi valable pour chaque type de jour.

Échelles informatives. L'indice de similarité calculé dans la première phase de la méthode est basé sur les coefficients d'ondelettes. La distance proposée (basée sur la DWT) entre deux blocs $Z_m(t)$ et $Z_{m'}(t)$ est

$$D(m,m') = \sum_{j=j_0+1}^{J} \left\{ 2^{-j} \sum_{k=0}^{2^j-1} (d_{j,k}^m - d_{j,k}^{m'})^2 \right\}^{1/2}.$$

Le niveau j_0 est l'échelle à partir de laquelle nous séparons approximation et détails de la trajectoire temporelle en concentrant dans l'approximation l'essentiel des comportements non stationnaires. Pour mieux comprendre cette distance, nous affichons dans le graphique 17 les contributions de chaque échelle de la DWT à l'énergie globale pour deux semaines d'historique de la consommation d'électricité.

Il est facile à voir qu'il n'y a que trois échelles qui sont informatives : elles correspondent aux résolutions le plus grandes. En autre, les échelles plus fines n'apportent pas d'information significative (des contributions à l'énergie globale faibles et constantes) pour comprendre la dynamique de la consommation. Donc, elles ne sont pas informatives dans le calcul de la distance. Nous voudrions confirmer cette piste. Nous allons faire la

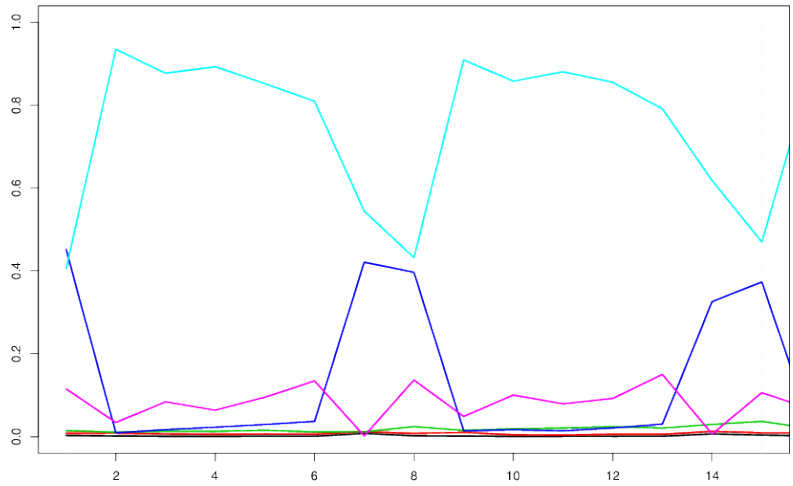


FIGURE 17 – Contributions à l'énergie globale de la courbe par échelle pour deux semaines de la consommation d'électricité.

.
.

Bloc	CALEN-TR	j^*	KLS
lundi	1.95	2.06	2.63
mardi	1.67	1.68	2.03
mercredi	1.54	1.56	1.9
jeudi	1.23	1.34	1.75
vendredi	1.66	1.7	1.83
samedi	1.57	1.48	1.79
dimanche	1.55	1.6	1.74
férié	3.33	2.7	3.23
Global	1.64	1.66	1.99

TABLE 6 – MAPE par type de jour et global pour les méthode avec centrage DIFF et groupes calendaire CALEN-TR, la version $j^* = 2$ et la version KLS. La fenêtre h_n est réglé par DYN.

prévision avec la méthode retenue (centrage DIFF et groupes CALEN-TR). Dans l'expérience nous calculons les distances seulement à partir de l'échelle j^* avec $j^* = 1, 2 \ldots$, et nous faisons la comparaison avec le calcul par défaut ($j^* = 0$).

Les résultats (*cf.* Table 6) montrent que les deux échelles les plus fines n'apportent effectivement pas d'information significative. Autrement dit, le niveau de qualité dans la prévision ne se voit pas modifié quand nous enlevons ces échelles.

Distance sur la répartition de l'énergie. Avec l'objectif d'explorer la capacité de la méthode dans le cadre d'une autre distance, nous proposons d'essayer une distance basée sur la répartition de l'énergie.

Chaque jour peut être caractérisé par l'ensemble des contributions des échelles à l'énergie globale de chaque fonction. Nous avons donc une autre représentation des jours : chaque jour est représenté par un vecteur de probabilité. Nous allons calculer la similarité

Bloc	2001	2002	2003	2004	2005	2006	2007	2008	2009
lundi	1.72	1.55	1.79	1.6	1.66	1.61	1.66	1.71	1.95
mardi	1.54	1.6	1.79	1.65	1.34	1.63	1.46	1.52	1.84
mercredi	1.51	1.33	1.76	1.56	1.23	1.33	1.55	1.53	1.6
jeudi	1.72	1.39	1.65	1.71	1.61	1.56	1.54	1.62	1.7
vendredi	1.83	1.65	1.83	1.58	1.52	1.59	1.46	1.77	2.19
samedi	1.65	1.56	1.82	1.54	1.59	1.64	1.75	1.59	1.88
dimanche	2.47	2.17	2.01	2.1	2	2.19	2.52	2.17	2.58
férié	3.83	4.87	2.5	4.23	3.17	2.94	4.33	2.76	3.3
Global	1.84	1.7	1.83	1.75	1.61	1.69	1.79	1.73	2

Table 7 – MAPE par type de jour et par année pour la prévision avec centrage DIFF et groupes CALEN-TR du 2001 au 2009.

entre jours à partir de cette représentation en utilisant une distance entre fonctions de probabilité. Nous allons utiliser la mesure de divergence de Kullback-Leibler. Pour deux distributions de probabilité r et s, on défini la divergence de s par rapport à r comme

$$D_{\mathrm{KL}}(r\|s) = \sum_i r(i) \log \frac{r(i)}{s(i)}.$$

Cette mesure n'est pas symétrique, mais nous pouvons la rendre symétrique en faisant $D_{\mathrm{KLS}}(r,s) = D_{\mathrm{KL}}(r\|s) + D_{\mathrm{KL}}(s\|r)$. Nous utilisons cette variante pour obtenir des prévision. Les résultats sont présentés dans la Table 6.

Le résultats de prévision ne sont pas meilleurs que pour la distance originale. Malgré la dégradation de la performance vis-à-vis de la distance originale, le résultat n'est pas mauvais d'un point de vue absolu.

10.3 Incidence de la période de test.

Jusqu'à présent, la performance de la méthode de prévision a été toujours évaluée sur le même type de données et sur la même plage temporelle. Nous pouvons nous demander si les performances obtenues ne sont pas liées à la période d'étude. Pour répondre à cette question, nous utilisons une historique plus long et plus récent (du 1 janvier 1996 jusqu'au 31 décembre 2009). Nous prévoyons depuis le 1 janvier 2001 jusqu'au 31 décembre 2009 jour par jour en utilisant la méthode de prévision avec centrage DIFF et groupes issues des transitions de calendrier CALEN-TR.

Les erreurs de prévision par type de jour et par année sont présentées dans la Table 7. La performance a une variation naturelle due aux différences des conditions météorologiques de chaque hiver ou encore à la différence de structure du calendrier en relations aux jours fériés. Par exemple, la canicule de 2003 explique que le mois de juillet de cette année a la plus grande erreur en prévision. De la même façon, les vagues de grand froid de 2009 produisent un niveau d'erreur en décembre qui est extraordinairement haut. D'autant plus important que le modèle n'incorpore aucune information sur des variables météorologiques.

Nous avons testé la méthode de prévision sur un autre type de données d'une nature un peu différente car ce sont des données de consommation d'électricité de clients professionnels (des entreprises). Ces données sont moins sensibles à la température. Ce test a été dans le cadre du projet SIGMA du département OSIRIS. Le projet a pour but de développer un système informatique appelé `StreamBase` que recevra de grands volumes de données de consommation. Nous avons produit une variante du package `kerwavfun` capable de travailler au sein de la maquette `StreamBase`. La méthode KWF avec centrage DIFF et une structure de groupes du type CALEN-TR simplifié a été mise en compétition avec une modélisation du type GAM conçue par des ingénieurs d'EDF. La méthode s'est avère concurrente de la méthode GAM.

10.4 Classification pour incorporer de l'information exogène.

Jusqu'à présent les groupes avec lesquels nous avons travaillé étaient basés sur l'information (déterministe) du calendrier. Nous allons essayer d'améliorer ces groupes : nous allons faire le premier essai d'incorporation d'information exogène dans le modèle.

Il est bien connu que la courbe de charge française a une très importante thermosensibilité.Nous allons donc incorporer l'information de la température dans le groupement. La structure calendaire ne peut pas être identifiée par la température, donc nous allons croiser les groupements de la température et des transitions du calendaire. Pour ne pas aboutir à des groupes avec trop peu d'observations, nous réduisons le croisement de cas au groupes de jours qui n'ont pas un jour férié dans la transition.

L'obtention de groupes basés sur la température sera fait par la méthode de classification fonctionnelle que nous expliquons dans la partie suivante.

Part II

Clustering functional data with wavelets.

Summary

Abstract

We present two methods for detecting patterns and clusters in high dimensional time-dependent functional data. Our methods are based on wavelet-based similarity measures, since wavelets are well suited for identifying highly discriminant local time and scale features. The multiresolution aspect of the wavelet transform provides a time-scale decomposition of the signals allowing to visualize and to cluster the functional data into homogeneous groups. For each input function, through its empirical orthogonal wavelet transform the first method uses the distribution of energy across scales to generate a representation that can be sufficient to make the signals well distinguishable. Our new similarity measure combined with an efficient feature selection technique in the wavelet domain is then used within more or less classical clustering algorithms to effectively differentiate among high dimensional populations. The second method uses similarity measures between the whole time-scale representations that are based on wavelet-coherence tools. The clustering is then performed using a k-centroid algorithm starting from these similarities. Practical performance of these methods that jointly design both the feature selection in the wavelet domain and the classification distance is illustrated through simulations as well as daily profiles of the French electricity power demand.

11 Introduction.

In different fields of applications, explanatory variables are not standard multivariate observations, but are functions observed either discretely or continuously. Ramsay and Dalzell (1991) gave the name "functional data analysis" to the analysis of data of this kind. As evidenced in the work by Ramsay and Silverman (1997, 2002) (see also Ferraty and Vieu (2006)), a growing interest is notable in investigating the dependence relationships between complex functional data such as curves, spectra, time series or more generally signals. Functional data often arise from measurements on fine time grids, and if the sampling grid is sufficiently dense, the resulting data may be viewed as a sample of curves. These curves may vary in shape, both in amplitude and phase. Typical examples involving functional data can be found when studying the forecasting of electricity consumption, temporal gene expression analysis or ozone concentration in environmental studies to cite only a few.

Given a sample of curves, an important task is to search for homogeneous subgroups of curves using clustering and classification. Clustering is one of the most frequently used data mining techniques, which is an unsupervised learning process for partitioning a data set into sub-groups so that the instances within a group are similar to each other and are very dissimilar to the instances of other groups. In a functional context clustering helps to identify representative curve patterns and individuals who are very likely involved in the same or similar processes. Recently, several functional clustering methods have been developed such as variants of the k-means method (Tarpey and Kinateder (2003); Tarpey (2007); Cuesta-Albertos and Fraiman (2007)) and clustering after transformation and smoothing (CATS) (Serban and Wasserman (2004)) to model-based procedures, such as clustering sparsely sampled functional data (James and Sugar (2003)) or mixed

effects modeling approach using B-splines (Luan and Li (2003)) that mostly concentrate on curves exhibiting a regular behaviour.

Our interest in time series of curves is motivated by an application in forecasting a functional time series when the most recent curve is observed. This situation arises frequently when a seasonal univariate continuous time series is segmented into consecutive segments, for example days, and treated as a discrete time series of functions. The idea of forming a functional valued discrete time series from segmentation of a seasonal univariate time series has been introduced by Bosq (1991). Suppose one observes a square integrable continuous time stochastic process $X = (X(t), t \in \mathbb{R})$ over the interval $[0, T]$, $T > 0$ at a relatively high sampling frequency. The commonly used approach is to divide the interval $[0, T]$ into sub-intervals $[(l-1)\delta, l\delta]$, $l = 1, \ldots, n$ with $\delta = T/n$, and to consider the functional-valued discrete time stochastic process $Z = (Z_i, i \in \mathbb{N})$, defined by

$$Z_i(t) = X(t + (i-1)\delta), \qquad i \in \mathbb{N} \quad \forall t \in [0, \delta). \tag{11.1}$$

The random functions Z_i thus obtained, while exhibiting a possibly nonstationary behavior within each continuous time subinterval, form a functional discrete times series that is usually assumed to be stationary. Such a procedure allows to handle seasonal variation of size δ in a natural way. This set-up has been used for prediction when ones consider a Hilbert-valued discrete time stationary autoregressive processes (see Bosq (1991); Besse and Cardot (1996); Pumo (1992); Antoniadis and Sapatinas (2003)) or for more general continuous-time processes (see Antoniadis *et al.* (2006)). However, as already noticed above, for many functional data the segmentation into subintervals of length δ may not suffice to make reasonable the stationary hypothesis of the resulting segments, that is the key for efficient prediction. For instance, in modeling the electrical power demand process the seasonal effect of temperature and the calendar configuration strongly affects the mean level and the shape of the daily load demand profile. *Recognizing this, our aim is therefore to propose a clustering technique that clusters the functional valued (discrete) times series segments into groups that may be considered as stationary so that in each group more or less standard functional prediction procedures such as the one cited above can be applied.*

We will apply the methodology focusing on EDF's (*Électricité de France*[1]) national power demand for a year. This is essentially a continuous process even though we only count with discrete records sampled at 30 minutes for the whole year. Some of the facts associated to the electricity power demand induce to think that the process is not stationary. We will construct a functional data set by splitting the continuous process as in equation (11.1) where the parameter δ will be a day.

Although slicing an univariate time series produces functional data, we do not observe the whole segments but a sample of the values at some time points. One could then use a vector representation of each observation. For example Wang *et al.* (2008) proposed to measure the distance between observations through the high dimensional multivariate distribution of all sampled time points along each curve. This approach does not exploit the eventually potential information of correlations between points of a single curve. To avoid this, many authors have clustered the coefficients of a suitable basis representation of functions. Since the analyzed curves are infinite-dimensional and temporal-structured,

1. `http://www.edf.com`

one projects each curve over an appropriate basis of the functional space to extract specific features form the data which are then used as inputs for clustering or classification. One may cite for example Abraham *et al.* (2003) where the authors proposed to cluster the spline fit coefficients of the curves using $k-$means, or James and Sugar (2003) that use a spline decomposition specially adapted for sparsely sample functional data. Nevertheless, attention must be paid to the chosen basis because this operation involves linear transformation of data that may not be invariant for the clustering technique (see Tarpey (2007)). Splines are often used to describe functions with a certain degree of regularity. However, we will be working with curves like in Figure 24 that may present quite irregular paths. We chose to work with wavelets because of their good approximations properties on sample paths that might be rather irregular. One may note here that similar methodologies relying upon wavelet decompositions for clustering or classifying time series have been developed in the literature (e.g. see Pittner *et al.* (1999) and Ray & Mallick (2006)).

Wavelets offer an excellent framework when data are not stationary. For example, in Gurley *et al.* (2003) the wavelet transform is used to develop the concept of wavelet-coherence that describes the local correlation of the time-scale representation of two functions. Grinsted *et al.* (2004) proved that this concept is convenient for clustering geophysical time series. Another example supporting such a fact is the work by Quiroga *et al.* (2004) which uses wavelets to detect and cluster spikes on neural activity. Motivated by this and the fact that the wavelet transform has the property of time-frequency localization of the time series, we propose hereafter a time-series feature extraction algorithm using orthogonal wavelets for automatically choosing feature dimensionality for clustering. We also study some more complex variants using the wavelet-coherence concept in sake of better exploiting the well localized information of the wavelet transform.

The rest of the paper is organized as follows. Section 12 is a reminder on multiresolution analysis and introduces the basis supporting our feature extraction algorithm by means of the energy operator. Following wavelet analysis we cluster the functional data using the extracted features in Section 13. Our first clustering algorithm uses k-means as unsupervised learning routine. We test the proposed method in Section 14 on simulated and real data. Section 15 presents a more sophisticated method for clustering functional data using a more specific dissimilarity measure. Finally, we conclude the paper by summarizing the main contributions and perspectives in Section 16.

12 Feature extraction with wavelets.

In this section we first introduce some basic ideas of the wavelet analysis before introducing more specific material: the energy contributions of the scale levels of the wavelet transform which are the key tools for future clustering. More details about wavelets and wavelet transforms can be found for example in Mallat (1999).

We will consider a probability space $(\Omega, \mathcal{F}, P)$ where we define a function-valued random variable $Z : \Omega \to H$, where H is a (real) separable Hilbert space (e.g. $H = L^2(\mathcal{T})$ the space of squared-integrable functions on $\mathcal{T} = [0, 1)$ (finite energy signals) or $H = W_2^s(\mathcal{T})$ the Sobolev space of s-smooth function on $\mathcal{T}$, with integer regularity index $s \geq 1$) endowed with the Hilbert inner product $< ., . >_H$ and the Hilbert norm $\|.\|_H$.

12.1 Wavelet transform.

A wavelet transform (WT for short) is a domain transform technique for hierarchical decomposing finite energy signals. It allows a real valued function to be described in terms of an approximation of the original function, plus a set of details that range from coarse to fine. The property of wavelets is that the broad trend of the input function is captured in the approximation part, while the localized changes are kept in the detail components. For short, a wavelet is a smooth and quickly vanishing oscillating function with good localization properties in both frequency and time. This is suitable for approximating curves that contain localized structures. A compactly supported WT uses a orthonormal basis of waveforms derived from scaling (i.e. dilating or stretching) and translating a compactly supported scaling function $\widetilde{\phi}$ and a compactly supported mother wavelet $\widetilde{\psi}$. We consider periodized wavelets in order to work over the interval $[0,1]$, denoting by

$$\phi(t) = \sum_{l\in\mathbb{Z}} \widetilde{\phi}(t-l) \quad \text{and} \quad \psi(t) = \sum_{l\in\mathbb{Z}} \widetilde{\psi}(t-l), \quad \text{for} \quad t \in [0,1],$$

the periodized scaling function and wavelet, that we dilate or stretch and translate

$$\phi_{j,k}(t) = 2^{j/2}\phi(2^j t - k), \quad \psi_{j,k}(t) = 2^{j/2}\psi(2^j t - k).$$

For any $j_0 \geq 0$, the collection

$$\{\phi_{j_0,k}, k = 0, 1, \ldots, 2^{j_0} - 1; \psi_{j,k}, j \geq j_0, k = 0, 1, \ldots, 2^j - 1\},$$

is an orthonormal basis of $\mathcal{H}$. Thus, any function $z \in \mathcal{H}$ can then be decomposed in terms of this orthogonal basis as

$$z(t) = \sum_{k=0}^{2^{j_0}-1} c_{j_0,k}\phi_{j_0,k}(t) + \sum_{j=j_0}^{\infty}\sum_{k=0}^{2^j-1} d_{j,k}\psi_{j,k}(t), \tag{12.1}$$

where $c_{j,k}$ and $d_{j,k}$ are called respectively the scale and the wavelet coefficients of z at the position k of the scale j defined as

$$c_{j,k} = < z, \phi_{j,k} >_H, \quad d_{j,k} = < z, \psi_{j,k} >_H .$$

To efficiently calculate the WT, Mallat introduced the notion of mutiresolution analysis of $\mathcal{H}$ (MRA) and designed a family of fast algorithms (see Mallat (1999)).

With MRA, the first term at the right hand side of (12.1) can be viewed as a smooth approximation of the function z at a resolution level j_0. The second term is the approximation error. It is composed by the aggregation of the details at scales $j \geq j_0$. These two components, approximation and details, can be viewed as a low frequency (smooth) nonstationary part and a component that keeps the time-localized details at higher scales. The distinction between the smooth part and the details is determined by the resolution j_0, that is the scale below which the details of a signal cannot be distinguished. We will focus our attention on the finer details, i.e. on the information at the scales $\{j : j \geq j_0\}$.

From a practical of view, each function is usually observed on a fine time sampling grid of size N. In the sequel we will be interested in input signals of length $N = 2^J$ for

some integer J. If N is not a power of 2, one may interpolate data to the nearest J with $2^{J-1} < N < 2^J$. We have already seen that an advantage of the nested structure of a multiresolution analysis is that it leads to an efficient tree-structured algorithm for the decomposition of functions in V_J (Mallat (1989)) for which the coefficients $< z, \phi_{J,k} >_H$ are given and that allows to derive the coefficients of the Discrete Wavelet Transform (DWT). However, when a function z is given in sampled form there is no general method for deriving the coefficients $< z, \phi_{N,k} >_H$ and one has to approximate the projection P_{V_J} by some operator Π_J in terms of the sampled values $\mathbf{z} = \{z(t_l) : l = 0, \ldots, N-1\}$ of z. For regular enough wavelets, such an approximation is highly efficient (see Antoniadis (1994)) and justifies the following.

Denote by $\mathbf{z} = \{z(t_l) : l = 0, \ldots, N-1\}$ the finite dimensional sample of the function z. For the particular level of granularity given by the size N of the sampling grid, one rewrites the approximation Π_J of the projection P_{V_J} of z in (12.1) using the truncation imposed by the 2^J points and the coarser approximation level $j_0 = 0$, as:

$$\widetilde{z}_J(t) = c_0\phi_{0,0}(t) + \sum_{j=0}^{J-1}\sum_{k=0}^{2^j-1} d_{j,k}\psi_{j,k}(t), \tag{12.2}$$

where c_0 and $d_{j,k}$ are now denoting the empirical wavelet coefficients derived from applying the DWT on the sampled values. Hence, for a chosen, regular enough, wavelet ψ and a coarse resolution $j_0 = 0$, one may define the DWT operator:

$$W_\psi : \mathbb{R}^N \to \mathbb{R}^N, \quad \mathbf{z} \mapsto (\mathbf{d}_0, \ldots, \mathbf{d}_{J-1}, c_0)\,,$$

with $\mathbf{d}_j = \{d_{j,0}, \ldots, d_{j,2^j-1}\}$. Since the DWT operator is based on an L_2-orthonormal basis decomposition, Parseval's theorem states that the energy of a square integrable signal is preserved under the orthogonal wavelet transform:

$$\|\mathbf{z}\|_2^2 = c_0^2 + \sum_{j=0}^{J-1}\sum_{k=0}^{2^j-1} d_{j,k}^2 = c_0^2 + \sum_{j=0}^{J-1}\|\mathbf{d}_j\|_2^2. \tag{12.3}$$

Hence, the global energy $\|\mathbf{z}\|_2^2$ of $\mathbf{z}$ is broken down into a few energy components. The representation (12.3) is in fact composed by the components of the discrete wavelet scalogram as defined in Arino, Morettin and Vidakovic (2004) and may be seen as the (DWT) analogue of the well-known periodogram from the spectral analysis of time series. Just as the periodogram produces an ANOVA decomposition of the energy of a signal into different Fourier frequencies, the scalogram decomposes the energy into "level components". Since $N = 2^J$ no more than J such levels can be defined. After removing from each continuous time series slowly varying trends and eventual periodicities in time by disregarding the approximation coefficient c_0, the scalogram components indicate at which levels of resolution the energy of the observed function is concentrated. A relatively smooth function will have most of its energy concentrated in large-scale levels, yielding a scalogram that is large for small j and small for large j. A function with a lot of high frequency oscillations will have a large portion of its energy concentrated in high resolution wavelet coefficients. Therefore the way these energies components are distributed and contribute to the global energy of a signal is the key fact that we are going to exploit to generate a handy number of features that are going to be used for clustering.

The image of the DWT operator applied on the column vector $\mathbf{z}$ of dimension $N = 2^J$ may be written in matrix form as:

$$\mathbf{W} = \mathcal{W}\mathbf{z},$$

where $\mathcal{W}$ is a N by N square matrix defining the DWT and satisfying $\mathcal{W}'\mathcal{W} = I_N$ (see Percival & Walden (2006, chap. 4)), and $\mathbf{W}$ is a column vector of length N with

$$\mathbf{W} = (\mathbf{d}_0, \mathbf{d}_1, \ldots, \mathbf{d}_{J-1}, c_0)',$$

where W' denotes the transpose of W. It is easy to see that if we consider a vector $\mathbf{x} = a + b\mathbf{z}$ with $a, b \in \mathbb{R}$, then the wavelet coefficients of the DWT of $\mathbf{x}$ are obtained from those of the $\mathbf{z}$ as:

$$(b\mathbf{d}_0, b\mathbf{d}_1, \ldots, b\mathbf{d}_{J-1}, a + bc_0)'. \tag{12.4}$$

12.2 Absolute and relative contributions.

We just have seen that DWT coefficients describe properties of functions both at various locations and at various time granularities. Each time granularity here refers to the level of detail that can be captured by DWT. This is therefore the reason of choosing the DWT as a representation scheme in our previous section to compare the shapes of curves for clustering. The energy $\mathcal{E}_z = \|z\|_H^2$ of the time series z via decomposition (12.3) is equal to the sum of the energy of its wavelet coefficients distributed across scales

$$\mathcal{E}_z \approx \|\mathbf{z}\|_2^2 = c_0^2 + \sum_{j=0}^{J-1} \|\mathbf{d}_j\|_2^2, \tag{12.5}$$

the approximation (denoted informally by $\approx$) holding because of the truncation at scale J for the wavelet expansion of z, discarding finer scales. If we consider $\mathbf{z}$ as the difference between two sampled curves, (12.5) justifies using the energy decomposition of wavelet coefficients for computing squared Euclidean distances between two series. However, when interested to see how the energy of wavelet coefficients is distributed across scales, other distance functions on DWT decompositions may be more appropriate for measuring the dissimilarity between two series.

In what follows, we define for $j = 0, \ldots, J-1$ the absolute and relative contributions of the scale j to the global energy of the centred function respectively as

$$\text{cont}_j = \|\mathbf{d}_j\|_2^2, \qquad \text{rel}_j = \frac{\text{cont}_j}{\sum_{j=0}^{J-1} \text{cont}_j}, \qquad \forall j = 0, \ldots, J-1. \tag{12.6}$$

We call these representations: the absolute contribution (AC) and the relative contribution (RC). We will therefore characterize each time series by the vector of its energy contributions or its relative contributions in order to define an appropriate measure of similarity that is going to be used for clustering. Note that in both of these choices of representation we leave out the eventual mean level differences of the time series since we do not make any use of the approximation term c_0 in their definition. Indeed, when within a same similarity class, the discrete time function valued processes Z are considered as stationary and therefore the scaling coefficient c_0 does not have any discriminative power, that's why we use only details after 0 in defining our distance. Now in order to compare

two paths and since they are zero-mean, our distance is more relevant to measure a difference on how their energies differ across scales. In our ultimate goal to predict the future behaviour of the discrete time functional valued process Z, collecting the series of scaling coefficients for each segment Z_i, $i = 1, \ldots, n$ one ends up with a real valued process of approximation coefficients. For a sequence of such univariate approximation coefficients, say $\{c_{0,1}, \ldots, c_{0,n}\}$, more or less classical time series models can be used to predict the next coefficient $c_{0,n+1}$. Using this fact, the considered dissimilarities computed throughout the representations will be invariant under vertical shifts of the curves. Moreover, using RC implies fixing the energy of the curves to one. Hence no difference in amplitude can be detected.

13 A $k-$means like functional clustering procedure.

We have presented a way to represent the infinite-dimensional original objects in J features that summarize the time evolution of the curves at different scales. We will now see how we use the information that we have coded to effectively cluster it.

This section starts with a brief review of some recent and highly efficient developments on feature selection and on the choice of the number of cluster.

13.1 Feature selection.

Nothing warranties that all the extracted features are relevant to discover the cluster structure. Analogously with regression analysis, a feature selection step can be performed to detect the significant ones.

Feature extraction and features selection have really different aims. Whether the former creates some new information from existing objects, the latter only selects a subset of existing features. This selection reduces the computational time of the algorithm and helps to avoid an unsatisfactory and unstable clustering. Another important advantage of using a feature selection algorithm is that the reduced number of features helps to better understand the cluster output. In our case, the number of features depends on the number of sampling points of the acquired data. For N points, the number of features is $J = \log_2(N)$ that can be large. Moreover, since we are interested in the energy decomposition across scales, potentially several scales will not be informative for the cluster structure. Besides this, the feature selection algorithm aims to reduce (or eliminate) the presence of nonsignificant variables and a possible redundant information that could hide the cluster structure.

The absence of class labels on unsupervised learning rends particularly difficult the feature selection task. Besides, this task is intricately connected with the determination of the number of cluster. Thus, it is desirable to conduce both of them simultaneously. A recent comparative study (Steinley & Brusco (2008)) evaluate the performance of eight feature selection algorithms on simulated data sets. The compared algorithms covers model-based approaches (e.g. Law *et al.* (2004) which allow the simultaneous detection of groups and features) and nonparametric ones. The same authors proposed in Steinley & Brusco (2008) an algorithm that combines a variable transformation with a variable

selection technique. The variable transformation introduces a variance-to-ratio weighting that looks for placing the variables on the same scale while preserving their ability to reveal cluster structures. Then, the transformed variables are used to construct an index of clusterability that serves to screen the variables that do not reveal information about the cluster structure (which is useful when working with large data sets with many masking features). Then, for the remaining variables an exhaustive evaluation of the feasible subsets of variables is done. For each subset size s, a best set of variables is obtained in terms of the largest proportion of explained variation VAF(s) from the clustering. Note that VAF decreases monotonically as a function of s. Finally, the solution of the algorithm is the subset of variables that maximizes the following ratio

$$\frac{\text{VAF}(s) - \text{VAF}(s+1)}{\text{VAF}(s-1) - \text{VAF}(s)},$$

where the heuristic is that for the right number of features, say s^*, adding one extra variable produces a larger degradation on the VAF than the one produced when passing from $s^* - 1$ to s^*.

13.2 Determination of the number of clusters.

One of the most difficult task in clustering is the determination of the number of clusters K. Even if some statistical support can be given to achieve this task, usually the knowledge on the particular application helps on the choice. In the classical case, i.e. not the functional one, a lot of data-driven strategies can be defined. The first one by inspecting basically the within-cluster dissimilarity as a function of K. Many heuristics have been proposed trying to find a "kink" in the corresponding plot.

A more formal argument has been proposed by Tibshirani *et al.* (2001) by comparing, using the gap statistic, the logarithm of the empirical within-cluster dissimilarity and the corresponding one for uniformly distributed data. Slight modifications have been proposed to the original argument, see for instance Ye (2007).

Another point of view useful to determine the number of clusters comes from model-based clustering. The idea is to fit a Gaussian mixture model to data and identify clusters as mixture components. The number of clusters is usually obtained using the well-known BIC criterion. All the above mentioned strategies seem to perform well when data do come from a mixture model but can perform poorly when the situation is more confused and fuzzy.

For determining the number of clusters James and Sugar (2003) proposed an information theoretic approach. They consider the transformed distortion curve $d_K^{-p/2}$, a kind of average Mahalanobis distance between data and the set of cluster centers C as a function of K. Jumps in the associated plot allow to select sensible values for K while the largest one can be the best choice for a mixture of p-dimensional distributions with common covariance. An asymptotic analysis (as p goes to infinity) states that, when the number of clusters used is smaller than the correct number (when any), then the transformed distortion remains close to zero, before jumping suddenly and increasing linearly. Then, detecting a jump in the transformed distortion curve is equivalent to detecting the number of clusters K.

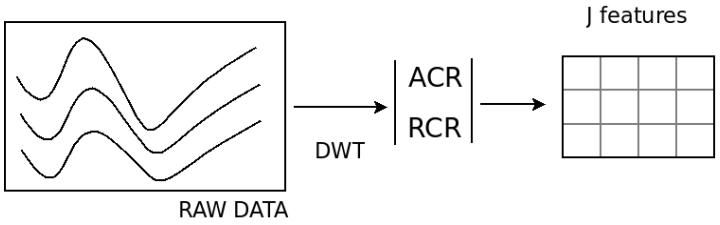


Figure 18: Diagram of the actual clustering strategy. Using the absolute and relative contribution representations (respectively ACR and RCR), the infinite dimension of the curves is reduced in J features per curve.

13.3 The actual procedure.

We can now sum up the actual strategy for clustering that we will use in the first part of the paper in the following steps:

0. **Data preprocessing.** Approximate sample paths of $z_1(t), \ldots, z_n(t)$ by the truncated wavelet series (see (12.2)) at the scale J from sampled data $\mathbf{z}_1, \ldots, \mathbf{z}_n$.
1. **Feature extraction.** Compute either the energetic components using absolute contribution (AC) or relative contribution (RC). If using the latter, transform the obtained vector using the logit transformation.
2. **Feature selection.** Use a feature selection algorithm for screening irrelevant variables.
3. **Determine the number of clusters K.**
4. **Clustering.** Obtain by k-means algorithm the K clusters using the selected features.

The preprocessing step is mandatory when working with functional data that is only observed on a finite grid. Figure 18 represents the obtained sample paths. Step 1 extracts from infinite dimensional curves a set of finite dimensional features. This allows us to employ multidimensional data analysis techniques. For the AC we have a vector of positive components that sums up the global energy on the details $\sum_j \|\mathbf{d}_j\|_2^2$. Meanwhile for the RC the vector, all its positive components sum up one, in other words we have a probability vector. This is a constraint if we want to use the $k-$means like algorithms because nothing warrants that the resulting clusters will be probability vectors. Therefore we transform the vector by using the well known logit transformation.

The next step is to select an adequate subset of features. The Steinley and Brusco (2008) algorithm is used performing directly the exhaustive evaluation of all the feasible subsets of variables. For the selected subset, we detect the number of clusters using the jump distortion approach of Sugar and James (2003). A last element to help on the determination of the number of clusters are the validation tools. Once a clustering has been achieved one can use diagnostic tools in order to asses the quality of the clustering. Usually, clustering analysis is not a linear procedure. The practitioner must try several solutions so it will iterate between steps three and four. The final quality of a clustering output will be assessed by means of diagnostic displays that will be presented later.

14 Numerical illustration.

We study the empirical performance of our clustering strategy on two functional data sets. The first one has been simulated using functional-valued processes and the second one is issued from the electricity power demand in France.

14.1 Simulated example.

We simulated a functional data set structured in three ($K = 3$) clusters. Each cluster is generated by a different continuous time process. The first one, inspired from Vidakovic *et al.* (2004), is a simple superposition of two sinusoids and a white noise: $f(x) = \sin(5\pi x/1024) + \sin(2\pi x/1024) + \epsilon$ with $x \in [1, 1024 \times 25]$. For the second and third clusters, we use two functional autoregressive (FAR) processes (see Bosq (2000)) defined by

$$g_n = \rho g_{n-1} + \epsilon_n.$$

Here, for each $n \in \mathbb{N}$ a functional valued random value g_n is observed, ρ is a linear bounded operator and ϵ_n is a functional valued strong white noise. We use the R package `far` (see Damon and Guillas (2007)) to generate them. The input parameters are inspired from Damon and Guillas (2005). The matrix associated to the autocorrelation operator (`d.rho` in the software) is in both cases diagonal with elements (0.45, 0.9, 0.34, 0.45). The only difference is on the `alpha` (on the software) matrix associated to the auto-covariance operator. We consider the following full matrix for the first process

$$\begin{pmatrix} 0.672 & -0.134 & 0 & 0 \\ 0.364 & 0.228 & 0 & 0 \\ 0 & 0 & 0.9 & 0 \\ 0 & 0 & 0 & 0.34 \end{pmatrix}$$

and only its diagonal for the second one.

As mentioned in the introduction, we produce 25 segments of 1024 points for each simulated trajectory. Doing this, we cope the seasonality of $f(x)$ rendering reasonable the hypothesis of stationarity for the obtained functional set. Note however that within each segment, no assumption of stationarity is made.

On the right of Figure 19 we plot one trajectory for each cluster. Note that the first model, dominated by a low frequency trend, is clearly distinguished from the two others whose differences are more intricate.

We apply to the sampled curves the DWT using a *Symmlet 6* wavelet (see Nason (2008)) and we extract the absolute contribution (AC) of energy for each detail level $j = 1, \ldots, 10$. We calculate the mean AC by cluster. The results are plotted on the left of Figure 19. Note how for a wide range of levels (from 1 to 6) there is no clear separation between cluster of the mean values of the AC. However, for the highest levels the extracted features show a better discrimination.

To effectively detect which are the informative scales we use the Steinley and Brusco's algorithm for variable selection for unsupervised learning. It needs as input the number of clusters so we perform the feature selection sequentially over a likely number of clusters

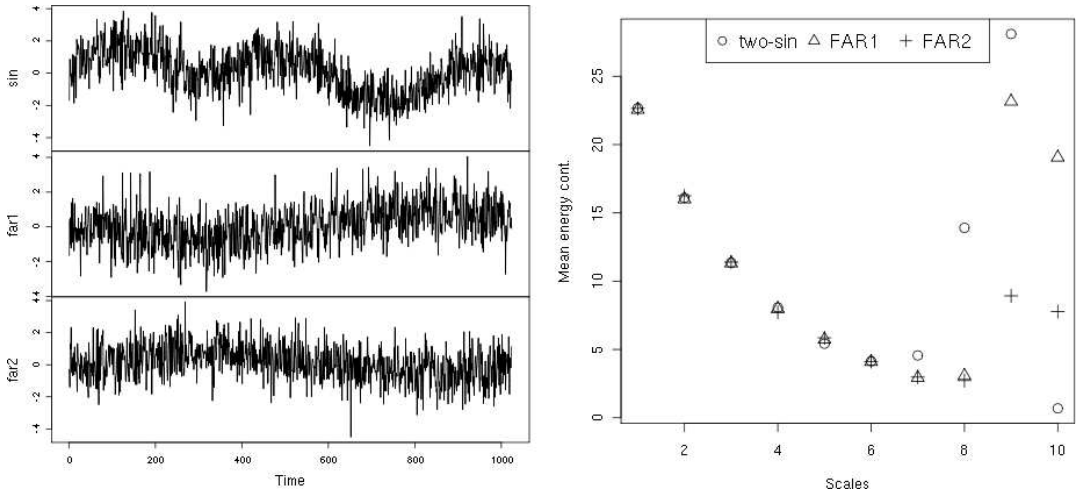


Figure 19: On the left, some typical simulated trajectories of the sinus model (top panel), the FAR1 model (middle), and the FAR2 model (bottom). On the right, the mean scales' energy absolute contribution by cluster.

Clustering	Feature Extraction	Raw curves
Abbreviation	RC	RAW
Mean Global error	21.05 (4.125)	25.32 (4.834)
Mean Rand Index	0.414 (0.092)	0.335 (0.109)

Table 8: Indicators of the clustering quality. Mean values over the 100 replicates with standard deviation between parenthesis.

ranging from two to twenty. We retain scales 8 to 10 which are associated with the lowest frequencies. We use the $k-$means algorithm (that we initialize many times, retaining the minimum within cluster distance solution) where the input data are the selected features and the number of clusters is the true one. We call this strategy RC. We then compare our clustering strategy with the one consisting in clustering the raw points directly, i.e. clustering the discretized trajectories as if they were vectors of dimension 1024 (we call it RAW). We obtain so the predicted membership of each instance (for both RC and RAW strategies) that we compare with the true ones. We repeat the process of generation of the three clusters and clustering 100 times to eliminate the effect of the data generation.

Table 8 contains two quality indicators of the clustering while Figure 20 presents the boxplots of the quality indicators across replications. We measure the global quality of the clustering by counting the number of observation misclassified and the Rand

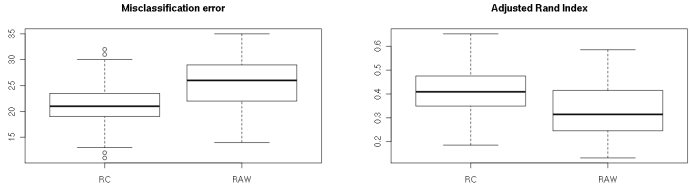


Figure 20: Boxplots of the misclassification error (left) and the Adjusted Rand Index (right) for the 100 replicates of the simulated data set.

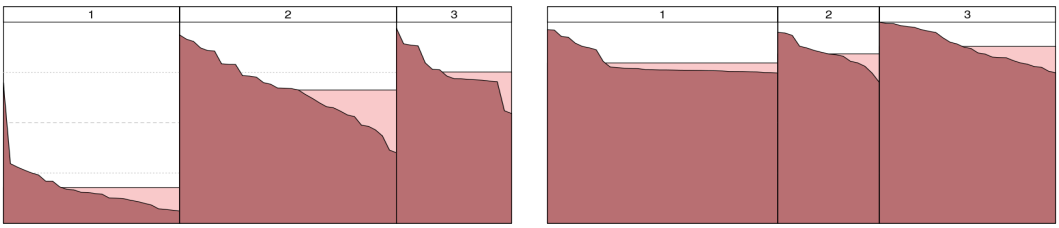


Figure 21: Shadow plots of resulting clustering for the simulated data set using $k-$means on the extracted feature (left) and on the raw curves (right). Class 1 corresponds to the sinus model, Class 2 to the FAR1 model and Class 3 to FAR2 model.

Index introduced by Rand (1971). This index measures the agreement between two clustering partitions. The index computes a ratio between the number of agreements between the partitions and the total number of comparisons. Hubert and Arabie (1985) the Adjusted Rand Index (ARI) corrects the agreement by change of the original index (usually produced when the size of partition is unequal). ARI varies between -1 and 1. Values close to 0 show partitions that are does not agree, while if it is close to 1 in absolute value the partitions are highly correlated.

The mean difference of the global error between RC and RAW clustering is significantly less than zero ($p-$value: $1e^{-10}$). In the same way, we obtain a significant greater value of the Rand Index for RC clustering ($p-$value: $3.7e^{-8}$). We can so deduce that our strategy provides a significant gain in the mean performance of the clustering when comparing with the RAW. Moreover, as shows the length of the boxplots RC produces less variable results that RAW. The ARI is computed between the clustering output of each strategy, RAW and RC, and the real classes simulated for this example. We will then talk of recovery capacity instead of agreement. As the ARI is larger, the mean recovery of RC is shown to be better than the recovery from RAW.

Even with a lower global rate error, one may be interested in the quality of each cluster. We will help us by using some recent validity tools. The shadow plot (Leisch (2010)) helps to examine the intraclass quality. The shadow of an observation measures it distance to its centroid and to the second nearest centroid. If the observation is close to its centroid the shadow value is near 0, while if it is at the same distance between the first and second nearest centroids it gives values near to one. The shadow plot is a graphical representation of all shadow values arranged by cluster and sorted in decreasing order within the cluster. We represent the shadow plots of the resulting clustering of a randomly selected replicate in Figure 21. The unequal width of each box is due to the different number of observations in each cluster. We see that globally the shadow values of the RC clustering (on the left) are lower than those of the RAW clustering (on the left) which shows that the former clustering provides more compact clusters than the latter one. Remark how the first class (the one that coincides with the sinus model) reveal a very compact cluster for the RC clustering. If the first cluster has such a good separation, most of the misclassification error must be made by confusing classes 2 and 3. Indeed, class 2 is formed by almost all the trajectories issued from the FAR1 process and some of the FAR2 process. The relative sizes can be compared in terms of width of the respectives boxes on the shadow plot.

Another useful diagnostic tool to assess the clustering quality of some high dimensional data is the neighborhood graph (see Leisch (2006) and its R package `flexclust`). The

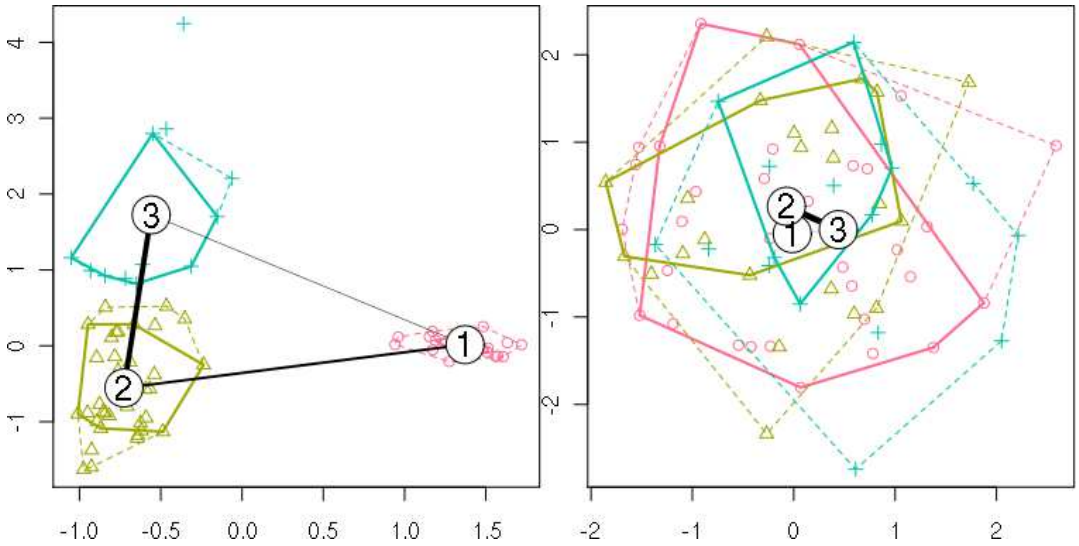


Figure 22: Neighborhood graph of resulting clustering for the simulated data set using $k-$means on the extracted feature (left) and on the raw curves (right).

main idea is to obtain some representation of the relative position of the clusters in some low dimensional space. As the projection to a plane may yield to a misinformation of how well two clusters are separated, the author proposes to combine it with a graph that reveals this information on the thickness of its edges. By this way, one constructs a graph where the nodes are the centroids and the thickness of the edges is proportional to the shadow mean value of the points belonging to the respective clusters of the relied centroids. To help the interpretation, one last element is added to the graphics: two convex hulls per cluster that will play an analogous role of the box plot for unidimensional data. The hulls formed by the thick lines correspond to the "inner 50%". For each cluster $k = 1, \ldots, K$, consider the distances from the centroid to each point of the cluster. Let be med_k be the median distance of the cluster k. The "inner 50%" of the cluster is defined as all the points with distance from the centroid is less or equal med_k. The second hull, the dashed one, contains all the points which distance from the centroid is less or equal $2.5 * med_k$.

The neighborhood graph of the RC and RAW clustering are shown on the left and right hand side respectively of Figure 22. For each clustering, the points are projected over the plane spanned by the two first principal directions of their respective spaces. The topology of both clusters is very different. While all the three convex hulls of the clusters on the RAW clustering overlaps in the principal plane, the separation of the clusters on the RC clustering is quite good. Thanks to the feature extraction, the cluster 1 (formed almost only by the sinus model observations) is completely separated from the rest of the observations on the principal plane. While the principal plane may lead to a reasonable representation of the feature space for the RC clustering (remember that we have selected 3 features), it seems some restrictive to give an accurate idea of the feature space on the RAW clustering (which dimension is 1024). However, the projected neighborhood graph will help us. Actually, both graphs are similar for RAW and RC. They show a thick line between classes 2 and 3 indicating that their members are more difficult to separate (they share more first and second centroids than with class 1). It seems to be easier to explain how classes are separated in RC clustering by just watching at the principal plane.

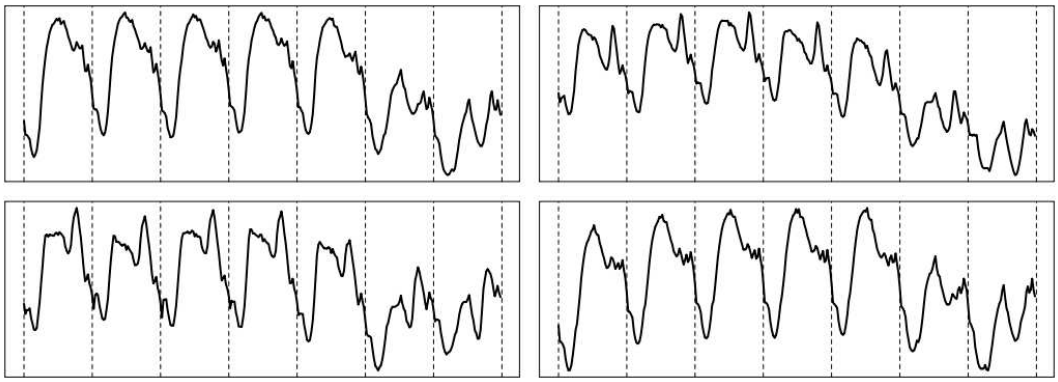


Figure 23: Four weeks of the French electricity power demand: autumn (top left), winter (bottom left), spring (top right) and summer (bottom right). Divisions on the week trajectories mark the daily load curves. Weeks start on Mondays.

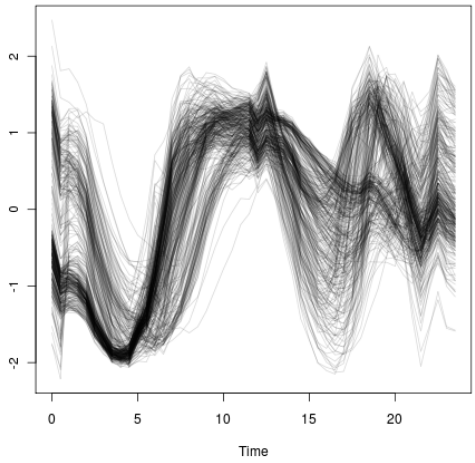


Figure 24: Standardized data from French electrical power. Each curve corresponds to a daily profile of 2006.

14.2 Electricity power demand data.

Our aim here is to discover clusters of daily load curves for a whole commercial year (September through August). Before we start let us state some well known facts for EDF's experts from this data set. We will help us by using Figure 23 where a sample of one week per season is presented. One important issue is the highly dependence on the meteorological conditions that is translated by a seasonal cycle. Also social and economic phenomenons (like holidays or the dichotomy between working-days and weekends) are present in the power demand. The structure of the weeks forms a weekly cycle where the profiles vary not only in mean level but also in the shape. It is also usual to found intraweekly differences, usually on the days next to weekends.

The centered profiles highlight the variation in shape mainly due to calendar structure (or to some special events like a few days when some industries are encouraged to reduce their demand). But the variability in the curves is not only reduced to first and second order moments. The standardized version of daily profiles shows that higher order

moments contribute also to the variability in the dynamic of the curves (see Figure 24). The objective of this empirical evaluation is to discover groups that reflects this heterogeneity to better understand the underlying structure.

Data preprocessing. From the practical point of view we count have a discrete equidistant grid of 17520 (= 365x48) time points of an underlying continuous process. After splitting the process as in (11.1) with $\delta = 1$ day, the corresponding discrete versions of (Z_n) are 48-length vectors $z_{i,J}, i = 1, \ldots, 365$. We use spline interpolation over each function in order to obtain $N = 64$ points ($J = 6$) to be able to use Mallat's pyramidal algorithm for the DWT.

14.2.1 Feature extraction and feature selection.

We proceed as before: for each $z_{i,6}, i = 1, \ldots, 365$ we compute the wavelet coefficients via the DWT. Then we calculate both the absolute and relative energy contributions of the scales $j = 1, \ldots, 6$ to the global energy (as in as in (12.6)). We will called them AC and RC respectively. For the RC we compute the logit transformations. We arrange the coefficients in two matrices of 365 rows and six columns.

For each data matrix the Steinley-Brusco's feature selection algorithm is used. As it needs as input the number of clusters we test it for a wide range of possibles number of clusters $k = 1, \ldots, K_{\max}$ for some large positive $K_{\max}$. For our application we used $K_{\max} = 20$).

The algorithms returns which variables are significant for each k. The results of the algorithm show that

- The significant scales for revealing the cluster structure are independent of the number of clusters used on the feature selection algorithm.
- As expected the significant scales are those associated with the mid-frequencies. On one hand, too large scales represent slow varying frequencies that are associated with the common day-night structure inherent to every day. On the other, too small scales capture very high frequency activity usually noise and thus no structure should be found neither.
- Finally, the scales that are significant may parametrize the cycle they represent. For the AC the scales that have been retained represent the cycles of 1.5, 3 and 6 hours. For the RC the scales retained are those associated to the cycles of 30 minutes, 1.5 and 3 hours.

14.2.2 Clustering results.

The next step is to determine the number of clusters in data for both representations. We use the Sugar and James' own implementation of their algorithm to detect a jump in the transformed distortion curve. The algorithm has a graphical output, presented in Figure 25. For the AC and RC the number of detected clusters is 8 and 5 respectively (right panel for each row in Figure 25). The difference in the number of clusters is explained by the fact that while AC is vertical shift and scale invariant, the RC is only invariant by vertical shifts. Hence it founds less of variability in curves.

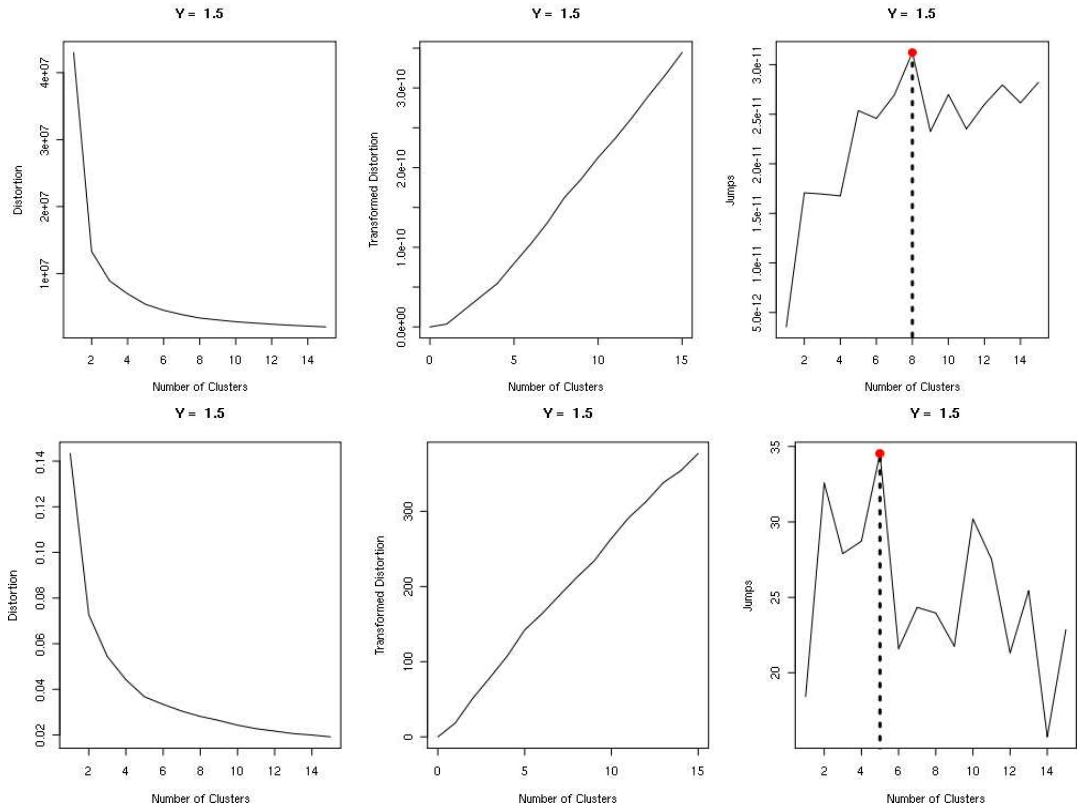


Figure 25: Detecting the number of clusters by feature extraction of the AC (top) and on the RC (bottom). From left to right we have the distortion curve, the transformed distortion curve and the first difference on the transformed distortion curve.

	AC	RC		AC	RC
Summer workdays	1, 5, 7	1, 5	Saturday	3, 8	2
Cold weekdays	2, 4, 6	4	Sundays and bank holiday	3, 8	3

Table 9: Resulting clustering for AC and RC variants.

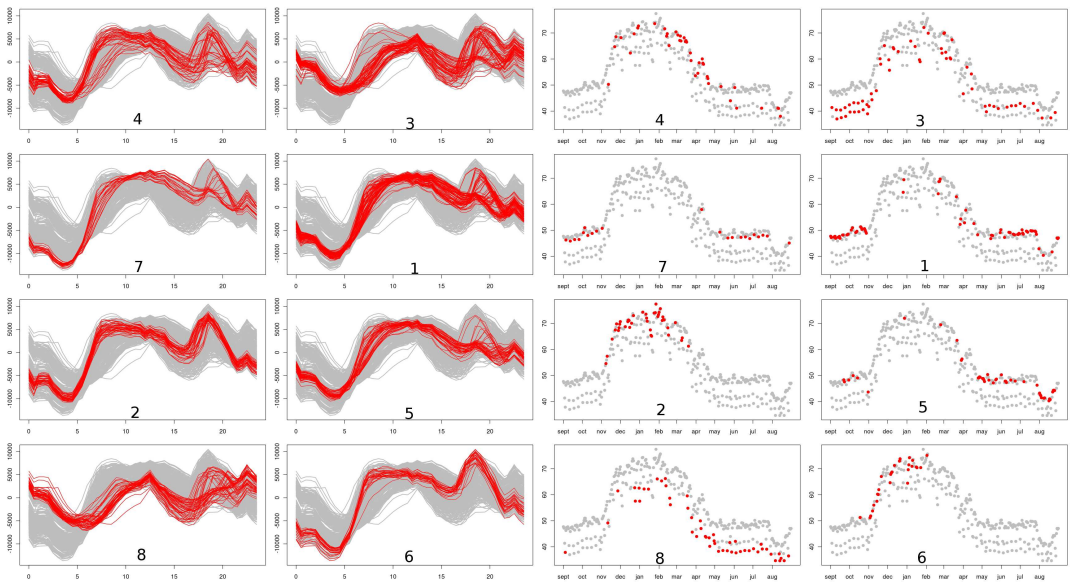


Figure 26: Curves membership (left) and calendar positioning (right) of the clustering using AC feature extraction. Numbers corresponds to labels of Table 9.

Then, we perform the detection of groups using the $k-$means algorithm. The input are the selected variables and the number of clusters detected by a significant jump in the distortion curve. The algorithm is randomly initialized 20 times and only the best result (in terms of minimum sum-of-squares-of-errors (SSE)) is retained.

Let us sketch some interesting facts of the clustering results obtained from both these variants. We are helped using a visual representation of the classes of daily curves that takes into account both the functional nature of data and the position in the year. Figure 26 shows the clustering result when using AC. The figure is composed by two groups of 8 graphics (one for each cluster).

Each graphic on the left hand side shows the curves that assigned to a specific class, while each one on the right hand side provides information about the position on the year of the instances of that class. Numbers were assigned to classes. The curves plots (left hand side) have a grey shade formed by all the (standardized) daily curves drawn on the background in order to help the visual comparison. On the calendar plots (right hand side), each day in the year is represented by the mean average load in GW/h arranged chronologically (ranging from September through August). Again a grey shade represents the set of all the days and only those corresponding to each cluster are highlighted.

Using the calendar information, we can interpret the results of the clustering. The resulting classes for AC and RC are enumerated in Table 9. First, note that we find the well known structure of the electrical power demand: two well defined periods covering the summer and winter seasons (for both AC and RC). Moreover, the clusterings reflects also the dichotomy of working days and weekends.

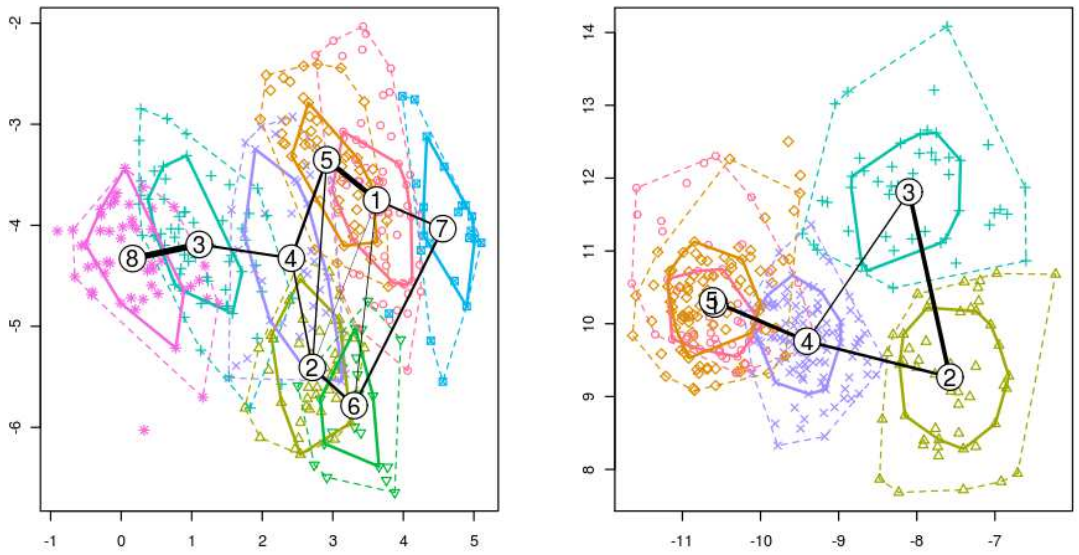


Figure 27: Neighborhood graphs of the two variants: AC (left) and RC (right) for the EDF data.

A second issue is the difference between the AC and RC. Basically when passing from AC to RC one eliminates some of the heterogeneity in classes. See for example how the three different classes for cold workdays 6, 2 and 4 of AC (corresponding to early, medium and late winter respectively) collapsed into one only class for the RC. This heterogeneity is consistent with the existence of two transition periods corresponding to the months of October-November and April-May. These transition days are no longer different from the winter workdays as soon as we do not authorize changes in scale (as mentioned before, RC is equivalent to standardized curves). So we can conclude that the transition days differ to the winter workdays by amplitude variation.

To better understand the topology of the clustering, Figure 27 presents the neighborhood graphs of the resulting outputs. For the AC clustering, the neighborhood graphs on the principal plane shows the summer classes 1, 5 and 7 (corresponding to middle of the workdays, Fridays and summer break days and Mondays respectively) forming a conglomerate. Then, the winter classes 2, 6 (resp. early and medium term winter) forming another. Late winter class (number 4) does the linkage between working days and the weekend days (classes 3 and 8). This is also observed for the RC clustering, where the only winter class (number 4) links workdays (classes 1 and 5) with weekend days (classes 3 and 2).

Results are quite satisfactory but at this step one could ask itself whether the feature extraction suffices to well cluster a set of curves. Moreover, time aggregation on the computation of the energy contribution deletes all time location information. We will try in the next section to develop some dissimilarity measure that better exploit the power of the wavelet transform.

15 Using the wavelet spectrums.

The success of any clustering algorithm depends on the adopted dissimilarity measure. Direct similarity measures such as L_p norms match two functional objects in their original representations without explicit feature extraction. When $p = 2$, this reduces to commonly used Euclidean distance. L_p norms are straightforward and easy to compute. However, in many cases such as in shifting and scaling, the distance of two sequences cannot reflect the desired (dis)similarity between them. Furthermore, L_p distance has meaning only in the relative sense when used to measure (dis)similarity.

In the previous section we have proposed instead the usage of the discrete wavelet transform of two curves of equal length to define a weighted normalized Euclidean like distance between them as a measure of their similarity. Indeed this was supported by two facts. First, the similarity between curves should be based on certain characteristics of the underlying sample path rather than on the raw data itself. Second, these characteristics describe the concentration of most of the energy in a small region of the scale-frequency domain. However the feature extraction procedure that we have used loses the location information. This is due to the aggregation on the time domain.

We may ask ourselves whether this loss is meaningful. To answer this question we present in this section two other direct and intuitive similarity measures that can be used for matching sequential patterns.

The adopted similarity measures are based on the wavelet coherence between two time series (here considered as curves) and in a principal components analysis of the common covariance structure measured in terms of the cross wavelet transform. These concepts provide a way of analyzing local correlation or covariance of time series both in the time domain and in the frequency domain. In this, they fundamentally differ from Fourier coherence that relies upon the correlation of the two series in the frequency domain only. In addition to locality, the continuous wavelet transform on which the wavelet coherence is based, possesses the very desirable ability of filtering the polynomial behavior to some predefined degree and therefore is invariant to vertical or scale shifts. Therefore, correct characterization of time series is possible, in particular in the presence of non stationarities like global or local trends or biases.

In what follows, we first recall some facts on the continuous wavelet transform and the concept of wavelet coherence between two time series. After that, we will use the maximum covariance analysis (MCA) for measuring the common time-frequency patterns and deduct a similarity measure between them.

15.1 Continuous WT.

Although Fourier analysis is well suited to quantifying constant periodic components in a time-series, it is not able to characterize signals whose frequency content changes with time. On the other hand, a Fourier decomposition may determine all the spectral components embedded in a signal and does not provide any information about when they are present. To overcome this problem, the wavelet transform decomposes a signal using functions (wavelets) that narrow when high-frequency features are present and widen on low-frequency structures (Daubechies (1992)). This decomposition yields a

good localization in both time and frequency and is well suited for investigating the temporal evolution of aperiodic and transient signals. Indeed, wavelet analysis is the time-frequency decomposition with the optimal trade-off between time and frequency resolution (Mallat (1999)).

Starting with a mother wavelet ψ, we consider the analyzing functions $\psi_{a,\tau}$ generated by simply scaling ψ by $a > 0$ and translating it by $\tau \in \mathbb{R}$

$$\psi_{a,\tau}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-\tau}{a}\right). \tag{15.1}$$

The parameter a is a scaling or dilation factor that controls the length of the wavelet (the factor $1/\sqrt{a}$ being introduced to guarantee preservation of the unit energy, $\|\psi_{a,\tau}\|_2 = 1$) and τ is a location parameter that indicates where the wavelet is centered. Scaling a wavelet simply means stretching it (if $a > 1$), or compressing it (if $a < 1$). Note that the choice of the wavelet function ψ is not arbitrary. This function verifies also the moment condition $\int \psi(t)dt = 0$.

Given a function $z \in L_2(\mathbb{R})$, its continuous wavelet transform (CWT), with respect to the wavelet ψ, is a function $W_z(a,\tau)$ defined as

$$W_z(a,\tau) = \int_{-\infty}^{\infty} z(t)\psi^*_{a,\tau}(t)dt, \tag{15.2}$$

where '$*$' denotes the complex conjugate. The wavelet decomposition is a linear representation of the signal z where the variance is preserved (Daubechies (1992)). Briefly, the continuous wavelet transform yields a redundant decomposition (the information extracted from a given scale slightly overlaps that extracted from neighbor scales) but it is generally more robust to noise as compared with other decomposition schemes (Poularikas (2009)).

In some sense, the wavelet transform can be regarded as a generalization of the Fourier transform and by analogy with spectral approaches, one can compute the local wavelet energy spectrum of z defined by $S_z(a,\tau) = |W_z(a,\tau)|^2$ where $|\cdot|$ denotes the modulus.

It is often desirable to quantify statistical relationships between two non-stationary signals. In Fourier analysis, the coherency is used to determine the association between two square-integrable signals z and x. The coherence function is a direct measure of the correlation between the spectra of two time-series (Chatfield (1989)). To quantify the relationships between two non-stationary signals, the following quantities can be computed: the wavelet cross-spectrum and the wavelet coherence.

The cross-wavelet transform is given by $\mathcal{W}_{z,x}(a,\tau) = W_z(a,\tau)W^*_x(a,\tau)$. As in the Fourier spectral approaches, the cross wavelet coherence can be defined as ratio of the cross-wavelet spectrum to the product of the spectrum of each series, and can be thought of as the local correlation between two CWTs. Here, again, we follow Grinsted *et al.* (2004) and define the wavelet coherence between two time series z and x as follows:

$$R_{z,x}(a,\tau) = \frac{|S(\mathcal{W}_{z,x}(a,\tau))|}{|S(\mathcal{W}_{z,z}(a,\tau))|^{1/2}|S(\mathcal{W}_{x,x}(a,\tau))|^{1/2}}, \tag{15.3}$$

where S denotes a smoothing operator composed by a time convolution by means of a Gaussian window and a scale convolution performed via a rectangular window

(see Grinsted *et al.* (2004) for details). Smoothing is necessary since without that step, coherence is identically equal to 1 for all (a, τ) (see for example Torrence and Compo (1998)). In Fourier analysis a similar problem is overcame by smoothing the cross-spectrum before normalization.

We will now explain our choice of wavelet coherence based distances motivated by the coefficient of determination R^2 from the linear regression framework.

15.2 Extended coefficient of determination.

Consider a single linear regression between a response variable Z and its regressor X. Given a set of sample data $(\mathbf{z}, \mathbf{x}) = (z_i, x_i)_{i=1,\dots,N}$ of length N, the model can be estimated in the least squares sense and a fitted line can be obtained. To measure how well does the adjusted regression line $\hat{\mathbf{z}}$ fits the data one should recall the definition of the coefficient of determination R^2:

$$R^2 = R^2(\mathbf{z}, \mathbf{x}) = \frac{\left[\sum_{i=1}^{N}(z_i - \bar{z})(x_i - \bar{x})\right]^2}{\sum_{i=1}^{N}(z_i - \bar{z})^2 \sum_{i=1}^{N}(x_i - \bar{x})^2},$$

where $\bar{z}$ indicates the average of z. Some simple calculations show that the coefficient of determination in simple linear regression is the square of Pearson's coefficient of linear correlation between $\mathbf{z}$ and $\mathbf{x}$. Therefore the coefficient R^2 measures the goodness of fit between the observed and the fitted values since it appears as the cosine of the angle θ of the vectors $\mathbf{z}$ and $\hat{\mathbf{z}}$. It takes its values between 0 and 1 where the closer the value is to 1, the better the regression line fits the data points.

The coefficient of determination R^2 has some desirable properties as reflexivity ($R^2(\mathbf{z}, \mathbf{z}) = 1$) or symmetry ($R^2(\mathbf{z}, \mathbf{x}) = R^2(\mathbf{x}, \mathbf{z})$). Thanks to these properties, R^2 appears to be a good similarity measure. Moreover R^2 has an intrinsic relation with the normalized Euclidean distance by mean-deviation. Indeed if $\nu(\mathbf{z})$ denotes the centered and standardized vector $\mathbf{x}$ and if $D_2(\nu(\mathbf{z}), \nu(\mathbf{x}))$ denotes the L_2 Euclidean distance between these two transformed vectors then it is easy to show that

$$D_2(\nu(\mathbf{z}), \nu(\mathbf{x}) = \sqrt{2N(1 - R^2(\mathbf{z}, \mathbf{x}))}. \tag{15.4}$$

Unfortunately, R^2 is applicable only to 1-dimensional sequences, but in our application, the sequences we will consider will be multidimensional (indexed by the scales in the wavelet coherence vector). To match a pair of J-dimensional sequences $\mathbf{Z}$ and $\mathbf{Y}$, we will therefore use the extended coefficient of determination ER^2 defined as:

$$ER^2 = ER^2(\mathbf{Z}, \mathbf{X}) = \frac{\left[\sum_{j=1}^{J}\sum_{i=1}^{N}(z_{ji} - \bar{z_j})(x_{ji} - \bar{x_j})\right]^2}{\sum_{j=1}^{J}\sum_{i=1}^{N}(z_{ji} - \bar{z_j})^2 \sum_{i=1}^{N}(x_{ji} - \bar{x}_j)^2},$$

where $\mathbf{z}_j$ and $\mathbf{y}_j$ are the $j-$th scale portions of $\mathbf{Z}$ and $\mathbf{Y}$. It is easy to see that ER^2 has properties similar to R^2.

15.3 Scale-specific ER^2.

Direct assessment of relationships between time series at specific scales has remained a challenging problem. Using the concept of wavelet coherence, we now show that

traditional statistical measures, such as the extended multiple coefficient of determination analyzed in the previous subsection, can be adapted to address scale-specific questions in non stationary time series data. Scale-specific global relationships between two time series $z = (z(t); t \in \mathbb{R})$ and $x = (x(t); t \in \mathbb{R})$ can be quantified by computing first an averaged over time squared coherence

$$R^2_{z,x}(a) = \frac{\int_{-\infty}^{\infty} |S(\mathcal{W}_{z,x}(a,\tau))|^2 d\tau}{\int_{-\infty}^{\infty} |S(\mathcal{W}_{z,z}(a,\tau))| d\tau \int_{-\infty}^{\infty} |S(\mathcal{W}_{x,x}(a,\tau))| d\tau}. \tag{15.5}$$

By definition (15.3) the values of $R^2_{z,x}(a)$ are bounded by $0 \leq R^2_{z,x}(a) \leq 1$ exactly as it holds for the coefficient of determination R^2. Moreover, the wavelet integrated over time squared coherency is equal to 1 when there is a strong linear association at a particular scale between the two signals, and equal to 0 if z and x are non correlated. We may therefore use this integrated squared wavelet coherence to define a scale-specific similarity measure WER^2 that looks like the familiar extended ER^2 based distance, say:

$$WER^2_{z,x} = \frac{\int_0^{\infty} \left(\int_{-\infty}^{\infty} |S(\mathcal{W}_{z,x}(a,\tau))| d\tau \right)^2 da}{\int_0^{\infty} \left(\int_{-\infty}^{\infty} |S(\mathcal{W}_{z,z}(a,\tau))| d\tau \int_{-\infty}^{\infty} |S(\mathcal{W}_{x,x}(a,\tau))| d\tau \right) da}. \tag{15.6}$$

Expression (15.6) must be approximated in practice because we do not observe the continuous sample paths $(z(t), x(t))$. Instead, we have sampled values $\widetilde{z} = \{z(t_i)\}$ and $\widetilde{x} = \{x(t_i)\}$ for $i = 1, \ldots, N$. Hence, we must approximate the integral operation by summations over the N time points. So in practice the CWT is computed only for $\tau = 1, \ldots, N$ and for an arbitrary set of scale values $a = \{a_j, j = 1, \ldots, J\}$. The smallest scale and the greatest scale are usually chosen as a power of two depending on the minimum detail resolution and the length of the time grid respectively. The rest of the values corresponds to a linear interpolation on a logarithmic scale with base 2.

These considerations yield in a $J \times N$ matrix W_z whose $(k, j)-$th element is

$$W_z(k, j) = \frac{1}{a_j} \sum_{i=0}^{N-1} z(t_i) \psi^* \left(\frac{i-k}{a_j} \right) \qquad k = 0, \ldots, N-1, j = 0, \ldots, J-1.$$

The derived similarity measure, mimicking (15.4) over the J scales is then given by

$$d(z, x) = \sqrt{JN \left(1 - \widetilde{WER}^2_{z,x} \right)}, \tag{15.7}$$

where $\widetilde{WER}_{z,y}$ is the analogous to (15.6) calculated over a discrete grid where we have replaced integrals by summations over the scale set and the time-points set.

15.4 MCA over the wavelet covariance.

We will explore a more sophisticated alternative to measure non linear relationship between two non stationary time series. This approach uses a Maximum Covariance Analysis (MCA) over a localized covariance matrix based on the CWT. This way we can obtain time-frequency patterns that explains the principal covariation of the time series. We will introduce a way of quantifying the similarity of these patterns. Using

MCA on the wavelet power spectrum of two series was originally proposed by Rouyer *et al.* (2008). .

As before, consider two time series $z = (z(t), t \in \mathbb{R})$ and $z = (z(t), t \in \mathbb{R})$ and their CWT $W_z = W_z(a, \tau)$ and $W_x = W_x(a, \tau)$ computed from the sampled sample path $\tilde{z}$ and $\tilde{x}$. We first define the time-frequency local covariance matrix by

$$Q_{zx} = W_z W_x^H,$$

where W_y^H is the conjugate transpose and Q_{zx} is a $J \times J$ symmetric matrix with possibly complex values. Performing a SVD of Q_{xy} gives the following decomposition

$$Q_{zx} = U \Gamma V^H,$$

where the columns of U and V are the orthonormal singular vectors of W_z and W_x respectively, and Γ is a diagonal matrix with the positive real numbers $\lambda_1 \geq \ldots \geq \lambda_J \geq 0$ that we arrange in decreasing order. These numbers, known as the singular values of the decomposition give important information about Q_{zx}. For example the zero-norm gives the rank of the matrix. We have also that using the Frobenius norm $\|Q_{zx}\|^2 = \|\Gamma\|^2 = \sum_{j=1}^J \lambda_j^2$ so the total inertia of the covariance matrix is decomposed into the sum of squared singular values. By this way, the quantity $\lambda_j^2 / \sum_j \lambda_j^2$ can be seen as the portion of explained covariance associated to the direction of the pair of the $j-$th singular vectors of U and V that we write u_j and v_j respectively.

We define also the $j-$th leading pattern as the projections of the CWT of z and x over their respective $j-$th singular vectors

$$L_z^j(t) = u_j^H W_z \quad \text{and} \quad L_x^j(t) = v_j^H W_x.$$

The leading patterns show how the wavelet scales evolve over time for the time series z and x in the orthogonal directions that maximize their common covariance. We can then decompose the CWT of z and x in terms of the singular vectors and the leading patterns obtaining

$$W_z = U L_z \qquad \text{and} \qquad W_x = V L_x,$$

where L_z and L_x are matrices that have the leading patterns of z and x respectively in their rows. As in PCA we chose the first D directions for the smallest D such that

$$\frac{\sum_{j=1}^D \lambda_j^2}{\sum_{j=1}^J \lambda_j^2} \geq \theta,$$

with θ is a prefixed threshold. Thus, we obtain approximations of the CWT for both z and x using

$$W_z \approx \sum_{j=1}^D u_j L_z^j \quad \text{and} \quad W_x \approx \sum_{j=1}^D v_j L_x^j,$$

where the informal notation $\approx$ is due to the truncation at the first D direction for the expansions. This approximation guarantees a portion θ of the inertia of the covariance matrix C.

To compare the evolution on time of each pair of leading patterns we measure how dissimilar is their shape. For the jth pair of leading pattern, take the first derivative of

the difference between them. The energy in this quantity is bigger if two leading patters presents very different evolutions. We finally measure this energy by taking the modulus

$$d_j(z,x) = |\Delta(L_z^j - L_x^j)|.$$

Finally, we aggregate all the significant directions using a weighted combinations with weights given by the explained square covariances

$$D(z,x) = \frac{\sum_{j=1}^{D} \lambda_j^2 d_j^2(z,x)}{\sum_{j=1}^{D} \lambda_j^2}.$$

In the literature more complicated proposals for measuring parallelism between curves in similar contexts have been proposed. For example Keogh and Pazzani (1998) proposed as parallelism index to measure the angles between the segments formed by each pair of consecutive points. Rouyer *et al.* (2008) use this index in the context of a MCA analysis on the wavelet power spectrum of two series. Aguiar and Soares (2009) use the MCA on a local covariation matrix based on the CWT with a complex wavelet. The parallelism index between the resulting complex valued leading patterns is measure as a complex angle of consecutive segments. Our proposal aims to measure the parallelism between complex valued vector avoiding to pass through the notion of complex angle.

15.5 Clustering electricity power data through the wavelet spectrum.

We test the proposed dissimilarities to cluster functional data on the the electricity power data. These dissimilarities impose a new challenge for the clustering algorithm: it is not clear why when using a distance other than the euclidean we should still calculate the centroids of the clusters by the average mean. We could for example argue that, as a consequence of the linearity of the WT the mean of elements is the mean element of a group. Otherwise, some deepness-based notion can be used to find some median-like element (e.g. see Cuevas *et al.* (2006); Febrero *et al.*(2008)). We will instead use the `cluster` R package to perform a partitioning around medoids (PAM). This technique admits a general dissimilarity matrix as input and is known to be more robust than $k-$means. The representative element of a group (medoid) is the element of the cluster that minimizes the dissimilarity to all the points of the cluster.

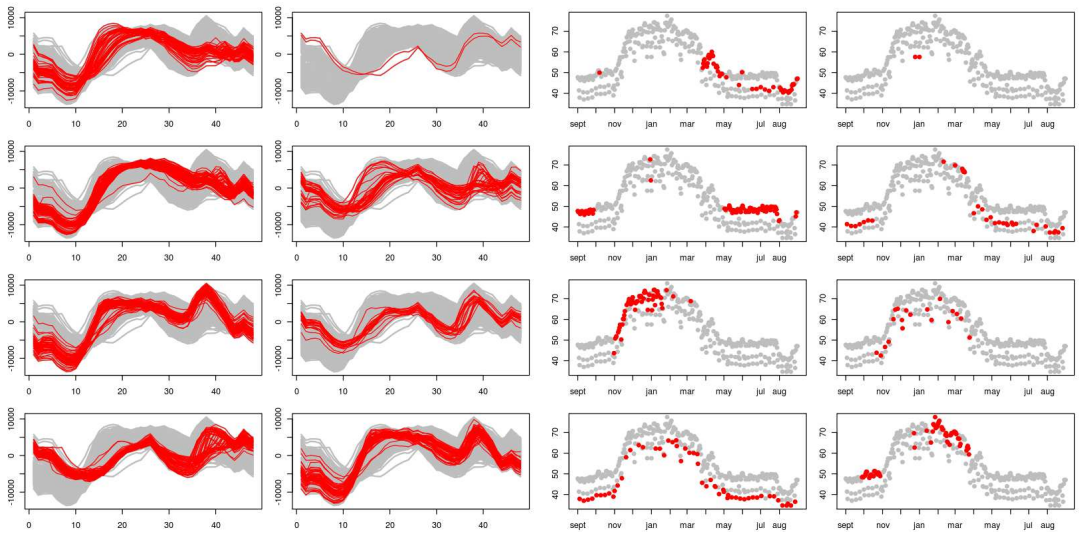
Using code from R package `sowas` (Maraun *et al.* (2004) and Maraun *et al.* (2007)), each one of the daily segments is transformed by means of the CWT using the Morlet complex wavelet (see Mallat (1999)). We use the discrete scales set $\{2^j, j = 1, \ldots, 4\}$ and 8 octaves between dyadic scales. The choice of this scales is made to filter the highest scales (above 12 hours) in order to eliminate the common low frequency pattern. The result is a 41×48 complex matrix for each segment.

Then, a dissimilarity matrix is computed for the wavelet based extended R^2 dissimilarity (WER) and another for the one based on the maximum covariance analysis over the wavelet transform (MCA). We use the PAM clustering to obtain $k = 8$ clusters with each dissimilarity matrix. The number of clusters is chosen in order to be able to compare the clustering results with the AC clustering.

With the same display used for Figure 26, Figures 28 and 29 show the clustering results founded using WER and MCA respectively. We compute the adjusted Rand index

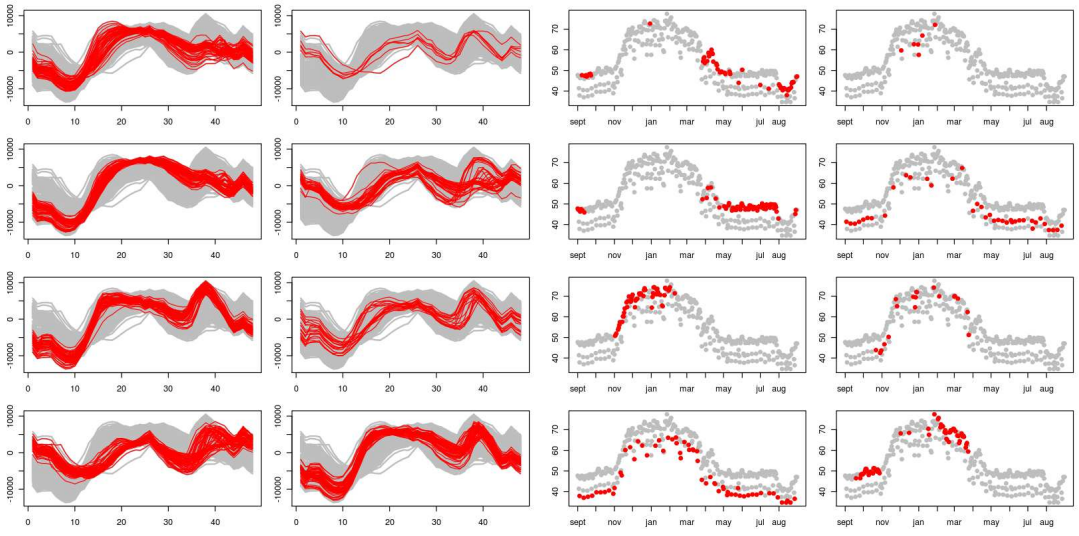
A.	Warming transition workdays	**E.**	Special days I
B.	Summer workdays	**F.**	Hot Saturdays
C.	Early winter workdays	**G.**	Special days II
D.	Sundays	**H.**	Late winter and cooling transition workdays

Table 10: Labels for the MCA and WER clustering of the electrical power demand data.



(a) Curves (b) Calendar

Figure 28: Curves membership of the clustering using WER based dissimilarity (a) and the corresponding calendar positioning (b).



(a) Curves (b) Calendar

Figure 29: Curves membership of the clustering using MCA based dissimilarity (a) and the corresponding calendar positioning (b).

between the resulting partitions. While we obtain an Index of 0.26 and 0.32 between the AC clustering and WER and MCA respectively, the rand Index between WER and MCA is 0.57. This show that the resulting outputs of WER and MCA are quite different from that obtained with AC, and more consistent between them. This is why we can use the same labels for the resulting clusters on both WER and MCA. Helping us with the calendar information we labeled the clusters as shown in Table 10. We leave two clusters with a generic names (special days I and II) to make a deeper analysis.

When we watch one year of daily curves, there is a large proportion of days where the dynamic of the demand is well known. This is the case of the clusters A, B, C, D, F and H where it is quite easy to find why they are together. Actually the graphics of the curves show that they are quite homogeneous groups with maybe some exceptions. However, important enhancement on the modeling on very few special tricky days can be made by founding this tricky days.

We focus our attention on the groups named "Special days" I and II (labels E and G respectively). These groups are not founded when using simpler clustering alternatives based on feature extraction. They are formed by only a few observations and they are not the same between the WER and MCA.

WER's cluster E only has two bank holiday (Christmas and New Year days) while for MCA clustering the cluster is formed by the eves of Christmas and New Year days and the New Year day as well as other three other cold weekend days. Cluster G has more elements (19 for both dissimilarities). For the WER dissimilarity it mainly formed by winter's Saturdays (16 out of 19). While for MCA the proportion of Saturdays is largely smaller (7 out 19) other winter tricky days are founded: the days of the one hour clock's shift due to the daylight saving time or the whole week between Christmas and New Year's day.

16 Concluding remarks.

The initial interest on clustering nonstationary functional time series was guided by the need for determining big structures of behavior of the daily power demand curves. The alternative would be to work directly over the vectors that constrains the sampled daily records using the L_2 metric. However, this turn out to be useless results in terms of clustering because one obtains groups that differ mainly in mean level and no information on the shape of the curves is recovered.

Our proposals for clustering functional data can be arranged into two types. While the first one is based on a wavelet-based feature extraction in order to use classical multidimensional clustering tools, the second way is to cluster using a dissimilarity measure between curves. Both approaches are based on the wavelet transform. Our choice is guided by the interesting theoretical properties of wavelets. They are particularly fruitful in describing functional objects with localized structures.

The feature extracted clustering is extremely fast thanks to the fast implementation of the DWT and the $k-$means algorithm. We believe that the feature selection step is particularly useful. On one hand, it eliminates non informative features that could induce an unsatisfactory clustering. On the other hand, it gives some adaptability to the

technique with respect to the particular application and also gives interpretation ability to the clustering technique.

In spite of the fact that the results of the first clustering strategy are useful in practice, more refined ones can be obtained using the dissimilarity based clustering proposals. On the daily load curves application we shown that very particular shapes of days can be founded with an extra computational burden. If for the WER dissimilarity this computational cost remains considerably low, the extra computational burden of the singular value decomposition for the MCA dissimilarity and the lack of meaningful different results from the WER dissimilarity drive us to think that it would not be useful in practice.

Part III

Introducing exogenous variables by Conditional Autoregressive Hilbertian Process.

Summary

Abstract

When considering the prediction problem of a continuous-time stochastic process on an entire time-interval in terms of its recent past, the notion of Autoregressive Hilbert processes (ARH) arises. This representation is a generalization of the classical autoregressive processes to random variables with values in a Hilbert space.

Many authors have proposed estimation strategies for this kind of process. We propose an extension by introducing a conditioning process on the ARH. By this way, the intrinsic linearity of ARH is overwhelm.

We propose estimators that use appropriate strategies available in the recent literature. Consistency results of the resulting prediction estimators are obtained. We illustrate the performance of the proposed methods by means of a simulated example.

17 Introduction

We focus ourselves on the problem of predicting a functional valued process $Z = (Z_k, k \in \mathbb{Z})$ where Z_k belongs to some functional space F. That is, given the discrete sequence $Z_1, \ldots, Z_n$, we want to obtain some information about the future value Z_{n+1}. Note that at each time point k the variable Z_k is a random element of a functional space F with some definition domain $\mathcal{T} \subset \mathbb{R}$. Separable Hilbert spaces appear to be an appropriate choice since they allow enough generality providing a rich geometrical structure that naturally extends the notion of Euclidean space (e.g. unique orthogonal projection over convex sets, parallelogram identity, countable bases, ...). Thus we concentrate on $F = H$ for H a (real) separable Hilbert space. For example, H can be set equal to the space of square integrable functions $L_2(\mathcal{T})$ defined over a compact $\mathcal{T}$. If one wants to work with some intrinsic smoothness in data, $H = W_2^s(\mathcal{T})$ the Sobolev space of functions with s square integrable derivatives, or $H = \mathcal{C}^m$ the space of functions with m continuous derivatives may be appropriate.

Given a sampling $Z_1, \ldots, Z_n$ from Z, the best predictor (in the quadratic mean loss function sense) of the future observation Z_{n+1} is its conditional mean given the past

$$\tilde{Z}_{n+1} = \mathbb{E}(Z_{n+1}|Z_n, \ldots,). \tag{17.1}$$

Usually $\tilde{Z}_{n+1}$ depends on the unknown distribution of Z. To deal with this prediction problem when Z is stationary, Bosq (1991) introduces the Autoregressive Hilbertian process of order 1 (ARH(1)) defined by

$$Z_{n+1} = \rho Z_{n-1} + \epsilon_n, \qquad n \in \mathbb{Z}, \tag{17.2}$$

with ρ a bounded linear operator over H and $\epsilon = (\epsilon_n, n \in \mathbb{Z})$ a strong H-valued white noise. For this process, the best predictor of Z_{n+1} given the past observations is $\tilde{Z}_{n+1} = \rho Z_n$. Notice that ρ is unknown. Two strategies can be followed here. The first one is to first estimate ρ and then apply it to the last observation Z_n to obtain a prediction of Z_{n+1}. The second one is to directly estimate the element $\hat{Z}_{n+1} = \widehat{\rho Z_n}$. This dichotomy mimics the problem of estimating a function say f at a point x_0. One can either estimate the whole function and then apply it to x_0 or estimate directly the value

of $f(x_0)$. The former approach is used by Bosq (1991); Besse and Cardot (1996); Pumo (1998), while the latter is studied by Antoniadis and Sapatinas (2003) and Kargin and Onatski (2008).

We will study the ARH model in the next section. Then, in Section 19 we propose an extension consisting on the introduction of some exogenous information that produces a new process called Conditional Autoregressive Hilbertian that we abbreviate CARH. We study the empirical performance of ARH and CARH models via a simulated example in Section 20.

18 Autoregressive Hilbert process

We begin this section by stating notation and recalling some relevant facts about linear operators on Hilbert space (see Kato (1976, Ch. 5) for details). We consider a discrete-time stochastic process $Z = (Z_i, i \in \mathbb{Z})$ defined over $(\Omega, \mathcal{F}, \mathbb{P})$ and taking values on a real separable Hilbert space $(H, \mathcal{H})$ with the associated Hilbert norm $\|.\|_H$ and the Hilbert inner product $< ., . >_H$. In other words, for each $i \in \mathbb{Z}$ the random variable Z_i is a measurable map from the probability space $(\Omega, \mathcal{F}, \mathbb{P})$ on the real separable Hilbert space H endowed with its Borel σ-algebra of subsets of $\mathcal{H}$.

We note $\mathcal{L}$ the space of bounded linear operators from H to H equipped with the uniform norm

$$\|\rho\|_{\mathcal{L}} = \sup_{\|z\|_H \leq 1} \|\rho(z)\|_H, \qquad \rho \in \mathcal{L}, z \in H.$$

This space seems to be a too large space, so one usually consider the subspace of compact operators $\mathcal{K}$ that is easier to deal with (see Mas (2007)). For instance, if the operator ρ is compact then it admits a unique spectral decomposition, i.e. for two bases $(\phi_j)_{j\in\mathbb{N}}$ and $(\psi_j)_{j\in\mathbb{N}}$ and a sequence of numbers $(\lambda_j)_{j\in\mathbb{N}}$ that we can choose to be non-negative (choosing the sign of ψ_j) we have

$$\rho = \sum_{j\in\mathbb{N}} \lambda_j \psi_j \otimes \phi_j,$$

where we use the tensor product notation $(u \otimes v)(z) =< u, x >_H v$ for any elements $z, u, v \in H$. We say that a operator ρ is self-adjoint if $< \rho u, v >_H =< u, \rho v >_H$ for all $u, v \in H$. If ρ is symmetric the decomposition becomes $\rho = \sum_{j\in\mathbb{N}} \lambda_j \phi_j \otimes \phi_j$ with eigen-elements $(\lambda_j, \phi_j)_{j\in\mathbb{N}}$. If ρ is not self-adjoint, we call ρ^* its adjoint. Finally we say that ρ is positive-definite if it satisfies $< \rho z, z >_H \geq 0$ for all $z \in H$. Two subspaces of $\mathcal{K}$ will be of our interest: the space of Hilbert-Schmidt operators $\mathcal{K}_2$ and the space of trace class (or nuclear) operators $\mathcal{K}_1$ defined respectively as

$$\mathcal{K}_2 = \{A \in \mathcal{K} : \sum_{j\in\mathbb{N}} \lambda_j^2 < \infty\}, \qquad \mathcal{K}_1 = \{A \in \mathcal{K} : \sum_{j\in\mathbb{N}} |\lambda_j| < \infty\}.$$

The Hilbert-Schmidt operators form a separable Hilbert space with inner product $< \rho, \tau >_{\mathcal{K}_2} = \sum_{j\in\mathbb{N}} < \rho\psi_j, \tau\psi_j >$ with $(\psi_j)_j$ an orthonormal basis and $\rho, \tau \in \mathcal{K}_2$ (the product does not depends on the choice of the basis, see Kato (1976, p. 262)). The associated norm yields from $\|\rho\|^2_{\mathcal{K}_2} = \sum_{j\in\mathbb{N}} \|\rho\psi_j\|^2_H = \sum_{j\in\mathbb{N}} \lambda_j^2$. On the other hand the space of trace-class operator endowed with the norm $\|.\|_{\mathcal{K}_1}$ defined as $\|\rho\|_{\mathcal{K}_1} = \sum_j |\lambda_j|$ is a

separable Banach space. Finally, from the continuity of the inclusions $\mathcal{K}_1 \subset \mathcal{K}_2 \subset \mathcal{K} \subset \mathcal{L}$ we have that

$$\|.\|_{\mathcal{K}_1} \geq \|.\|_{\mathcal{K}_2} \geq \|.\|_{\mathcal{L}}.$$

18.1 The ARH(1) model.

The Autoregressive Hilbertian process of order 1 (ARH(1)) states that at each n we can write

$$Z_n - a = \rho(Z_{n-1} - a) + \epsilon_n \tag{18.1}$$

where $\rho \in \mathcal{L}$, a is the mean level of the process and $\epsilon = (\epsilon_n)_{n\in\mathbb{Z}}$ is an $H-$valued strong white noise, i.e. ϵ is an independent and identical distributed sequence of zero-mean H-random variables with $0 < \mathbb{E}\|\epsilon_n\|_H^2 = \sigma^2 < \infty$ (see Bosq (2000, ch. 3)).

If it is exists some integer j_0 such that $\|\rho^{j0}\|_{\mathcal{L}} < 1$ then (18.1) has a unique solution which is a strictly stationary process with innovation ϵ.

The above model naturally generalizes the classical autoregressive model for scalar or multivariate time series. It has one only infinite dimensional parameter, namely the autoregressive operator ρ, that belong to the space of linear operators $\mathcal{L}$. We assume ρ to be bounded (and thus continuous) to work in a more comfortable framework with yet enough of generality.

Example 18.1. Let $H = L_2([0,1])$ with $< z, x >_H = \int_0^1 z(t)x(t)dt$ and $\|z\|_H^2 = \int_0^1 z^2(t)dt$ for all $z, x \in H$. Consider $\rho = r$ a kernel operator associated to some continuous, square-integrable kernel $R(.,.)$ defined by

$$r(z)(t) = \int_0^1 K(s,t)z(s)ds, t \in [0,1], \qquad z \in H.$$

The boundedness of r yields from the square integrability of R. Moreover, consider the following decomposition using an orthonormal basis of H, $\{e_j\}_{j\in\mathbb{N}}$,

$$\begin{aligned}\int_0^1 R^2(s,t)dsdt &= \sum_{i,j}\left(\int_0^1 R(s,t)e_i(s)e_j(t)dsdt\right)^2 \\ &= \sum_{i,j}\left(\int_0^1 r(g_j)(s)e_i(s)ds\right)^2 \\ &= \sum_j \|r(e_j)\|_H^2.\end{aligned}$$

Hence, r is a Hilbert-Schmidt operator. The sequence $(Z_n, n \in \mathbb{Z})$ follows a centred ARH if for all $N \in \mathbb{Z}$, then Z_n can be written as,

$$Z_n(t) = \sum_{j=0}^{\infty} r^j(\epsilon_{n-j}),$$

where the sequence $(\epsilon_n, n \in \mathbb{Z})$ is a strong white noise.

Example 18.2. If X_t follows an Ornstein-Uhlenbeck process, then it can be written in terms of an ARH (see Bosq (2000, p.76) for details).

18.2 Associated operators.

The expectation of the process can be estimated by the empirical mean function. In consequence, unless explicitly mentioned, we assume from now on that Z is centred.

We note H^* the topological dual of H, i.e. the space of bounded linear functionals on H. If we suppose that Z is strictly stationary and the $\mathbb{P}-$forth-order moment of Z exists, $\mathbb{E}\|Z_0\|_H^4 < \infty$, then we can introduce two linear operators defined from H^* to H associated to the ARH process Z. Thanks to the Riesz representation, H^* can be identified with H, and the following operators can be defined

$$\Gamma = \mathbb{E}[(Z_0 - a) \otimes (Z_0 - a)], \tag{18.2}$$

$$\Delta = \mathbb{E}[(Z_0 - a) \otimes (Z_1 - a)]. \tag{18.3}$$

Γ and Δ are called the *covariance* and *cross covariance operators* respectively. The operator Γ is positive-definite and self-adjoint. Notice that the cross-covariance operator Δ is not necessarily symmetric. We then consider its adjoint $\Delta^* = \mathbb{E}[(Z_1 - a) \otimes (Z_0 - a)]$. The defined operators are trace-class and hence Hilbert-Schmidt. We will just show it for the covariance operator since a similar reasoning can be used for the other operators. Let $Y_n = Z_n - a$ for all $n \in \mathbb{Z}$ a centred version of Z_n. Then, for an orthonormal basis $\{e_j\}_{j\in\mathbb{N}}$ of H, it suffices to to show that $\sum_j \|\Gamma(e_j)\|_H^2 < \infty$. Using the fact that expectation and bounded operators commute then we have for all $z \in H$ that,

$$\begin{aligned}
\|\Gamma\|_{\mathcal{K}_2}^2 &= \mathbb{E}\left[\sum_{j=1}^{\infty} \| < Y_0, e_j > Y_0\|_H^2\right] \\
&= \mathbb{E}\left[\|Z_0\|_H^2 \sum_{j=1}^{\infty} < Y_0, e_j >_H^2\right] \\
&= \mathbb{E}\|Y_0\|_H^4 < \infty.
\end{aligned}$$

The last expression is true since we assumed the existence of the forth moment of Z. We may then write down the spectral decomposition of Γ

$$\Gamma = \sum_{j\in\mathbb{N}} \lambda_j (e_j \otimes e_j)$$

where $(\lambda_j, e_j)_{j\in\mathbb{N}}$ are the eigen-elements of Γ. The eigen-values may be arranged to form a non-negative decreasing sequence of numbers towards zero. Covariance operators are unbiasedly estimated by their empirical counterparts from a sequence of Z say $Z_1, \dots Z_n$

$$\Gamma_n = \frac{1}{n} \sum_{i=1}^{n} (Z_i - \hat{a}_n) \otimes (Z_i - \hat{a}_n) \tag{18.4}$$

$$\Delta_n = \frac{1}{n-1} \sum_{i=2}^{n} (Z_{i-1} - \hat{a}_n) \otimes (Z_i - \hat{a}_n), \tag{18.5}$$

where $\hat{a}_n = n^{-1} \sum_{i=1}^{n} Z_i$ estimates a. Notice that the rank of Γ_n and Δ_n are finite since they are a finite sum of rank-one operators. In consequence, they are are nuclear and thus Hilbert-Schmidt. Moreover, Γ_n is positive-definite and symmetric by construction. Consistency results for $\hat{a}_n, \Gamma_n, \Delta_n$ and the empirical eigenvalues (calculated from the

spectral decomposition of Γ_n) $\lambda_{1,n} \geq \lambda_{2,n}, \ldots$ can be found in Bosq (2000, Chapters 3 and 4). Results concerning the eigenfunctions are also available in the same work. However, attention must be paid because the eigenfunctions can be estimated only up to their signs. As mentioned by the author, the parameter to be estimated is the eigenspace and not the eigenfunction. Therefore, the asymptotic results obtained do not guarantee the closeness of the empirical eigenfunctions $e_{j,n}$ to e_j (see Horváth et al. (2010)). However, in some situations one can circumvent the problem by setting arbitrarily the sign without loss of generalization (see for example Hall and Hosseini-Nasab (2006)). Practical implementations of the spectral decomposition of an empirical covariance operator may be hampered also by this issue producing sudden flips on the obtained eigenfunctions when one slightly modifies the input data (Horváth et al. (2010)).

Like in the scalar case, a Yule-Walker like relation exists between the operators Δ, Γ and ρ, namely

$$\Delta = \rho\Gamma. \tag{18.6}$$

Indeed, using the autoregression equation we have that $Y_1 = \rho Y_0 + \epsilon_1$, the moment equation yields to

$$\begin{aligned}\Delta &= \mathbb{E}[Y_0 \otimes Y_1] \\ &= \mathbb{E}[Y_0 \otimes \rho(Y_0)] + \mathbb{E}[Y_0 \otimes \epsilon_1] \\ &= \rho(\mathbb{E}[Y_0 \otimes Y_0]) \\ &= \rho\Gamma.\end{aligned}$$

Using the property of the adjoint and the symmetry of Γ, we obtain from (18.6) the following relation

$$\Delta^* = \Gamma\rho^*. \tag{18.7}$$

18.3 Estimation and prediction for an ARH(1) process.

If H is finite-dimensional, the equation (18.6) provides a way of estimating ρ by the inversion of the operator Γ. Using the fact that we count with well behaved estimators of the covariance operators, one may plug-in the empirical counterparts of the covariance operators and solve the equation on ρ. However, the inversion of Γ is a problem when H has infinite dimension because the operator is not bounded and may not be defined over the whole space H. To well identify ρ from (18.6) the eigenvalues of Γ need to be strictly positive. An analogous assumption is to demand the kernel of Γ to be null (see Mas and Pumo (2011)). In this case, a linear measurable mapping Γ^{-1} can be defined as $\Gamma^{-1} = \sum_{j\in\mathbb{N}} \lambda_j^{-1}(e_j \otimes e_j)$ with domain

$$\mathcal{D}_{\Gamma^{-1}} = \left\{ z = \sum_{j\in\mathbb{N}} < e_j, z > e_j \in H : \sum_{j\in\mathbb{N}} \left(\frac{< e_j, z >_H}{\lambda_j} \right)^2 < \infty \right\},$$

that is a dense subset of H. It turns to be an unbounded operator. However, from (18.6) we obtain that

$$\rho_{|\mathcal{D}_{\Gamma^{-1}}} = \Delta\Gamma^{-1},$$

where $\rho_{|\mathcal{D}_{\Gamma^{-1}}}$ is the autoregression operator restraint to $\mathcal{D}_{\Gamma^{-1}}$ as a consequence of $\Gamma\Gamma^{-1} = I_{\mathcal{D}_{\Gamma^{-1}}}$.

On the other hand, since the adjoint of a linear operator in H with a dense domain is closed (*closed graph theorem*, see for example Kato (1976, Theorem 5.20)) and that the range of the adjoint of the cross-covariance operator, Δ^*, is included in $\mathcal{D}_{\Gamma^{-1}}$ we can deduce from (18.7) that

$$\rho^* = \Gamma^{-1}\Delta^*.$$

As pointed out by Mas (2000) one can use classical results on linear operators to extend $\rho_{|\mathcal{D}_{\Gamma^{-1}}}$ by continuity to H, in order to obtain

$$\rho = \mathrm{Ext}(\Delta\Gamma^{-1}) = (\Gamma^{-1}\Delta^*)^* = (\Delta\Gamma^{-1})^{**}.$$

Then one may focus on the estimation of ρ^*. The theoretical properties of a such estimator are applicable to ρ through the composition of ρ^* by the adjoint operator.

Mas (2000) identifies two classes of estimators for ρ^*. The first one, the class of *projection estimators*, projects the data on an appropriate subspace H_{k_n} of finite dimension k_n. Let Π_{k_n} be the projector operator over H_{k_n}. Then one inverts the linear operator defined by the matrix $\Pi_{k_n}\Gamma_n\Pi_{kn}$ and completes with the null operator on the orthogonal subspace. In Bosq (2000), H_{k_n} is set equal to the one generated by the first k_n eigenfunctions of Γ. The subspace H_{k_n} is estimated by $\widehat{H}_{k_n}$, the linear span of the first k_n empirical eigenfunctions. By this way, if P_{k_n} is the projection operator on $\widehat{H}_{k_n}$, the estimator of ρ^* can be written as

$$\rho_n^* = (P_{k_n}\Gamma_n P_{k_n})^{-1}\Delta_n^* P_{k_n}. \tag{18.8}$$

The estimation solution by projection is equivalent to approximate Γ^{-1} by a linear operator with additional regularity $\Gamma^\dagger$ defined as

$$\Gamma^\dagger = \sum_{j=1}^{k_n} b(\lambda_j)(e_j \otimes e_j),$$

where $(k_n)_n$ is an increasing sequence of integers tending to infinity and b is some smooth function converging point-wise to $x \mapsto 1/x$. Indeed, $\Gamma^\dagger \to \Gamma^{-1}$ when $k_n \to \infty$. The choice of taking $b(x) = 1/x$ yields, for a finite k_n, to set $\Gamma^\dagger$ equal to a spectral cut of Γ^{-1}. However, this is choice is not unique. Mas (2000) consider a family of functions $b_{n,p} : \mathbb{R}_+ \mapsto \mathbb{R}_+$ with $p \in \mathbb{N}$ such that

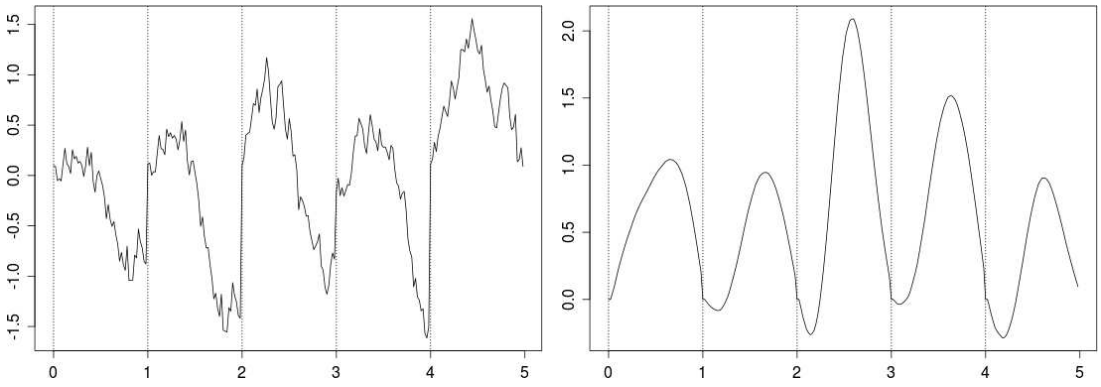
$$b_{n,p}(x) = \frac{x^p}{(x+\alpha_n)^{p+1}},$$

with α_n a strictly positive sequence that tends to 0 as $n \to +\infty$. With this, the second class of estimators for ρ^*, the *resolvent class*, is defined as

$$\rho_{n,p}^* = b_{n,p}(\Gamma_n)\Delta_n^*, \tag{18.9}$$

where we write $b_{n,p}(\Gamma_n) = (\Gamma_n + \alpha_n I)^{-(p+1)}$ with $p \geq 0$, $\alpha_n \geq 0$, $n \geq 0$. As for the projection class, the operators $b_{n,p}(\Gamma_n)$ from the resolvent class can be associated to an regularized approximation of Γ^{-1} to solve the inversion problem (see Antoniadis and Sapatinas (2003) for a discussion on this topic applied to the ARH estimation).

Both classes of estimators allow one to predict the future value Z_{n+1} from the sample $(Z_1, \ldots, Z_n)$ by first estimating the autocorrelation operator ρ^* with $(Z_1, \ldots, Z_{n-1})$, and



(a) WIEN (b) GS

Figure 30: A five blocs trajectory of ARH simulated process generated using a the WIEN variant (a) and the GS variant (b).

then applying it to the last available observation Z_n. Alternatively, one may directly predict Z_{n+1} by estimating the relevant elements of the range of ρ^*. Using a basis of the space, one may decompose ρZ_n and use the adjoint property. This second strategy is proposed in Antoniadis and Sapatinas (2003). Using wavelet basis the authors obtain considerable better prediction errors. Kargin and Onatski (2008) go further in this sense and proposed to use a basis adapted to the prediction task.

18.4 Simulation of an ARH(1) process.

Unlikely to the scalar version, on a functional context simulation may become a difficult task due to the infinite dimension of the space. Pumo (1992) uses a Wiener process to produce a functional white noise, i.e. the innovation part of the process. Then, through the Karhunen-Loève decomposition of the Wiener process he obtains the eigenfunctions associated to its covariance operator. The autocorrelation operator is defined to have the same eigenfunctions. Finally, the simulation is carry out using the equation (18.1) that defines an ARH process. A second variant is proposed in Besse and Cardot (1996) where the authors define the process through a stochastic differential equation.

More recently, Damon and Guillas (2005) proposed a generalization of the simulation procedure allowing the use of a non-Gaussian noise and a full autocorrelation operator. The simulation is performed on an approximation $m-$dimensional space H_m of the functional space H. The functional observations are projected on H_m using a basis of this space. Then an ARH structure is simulated on H_m. The procedure can be resumed as follows:

1. Choose an orthonormal basis of H_m.
2. Choose the H_m representation of ρ and Γ.
3. Compute the H_m representation of the covariance operator Γ_ϵ for the noise ϵ by $\Gamma_\epsilon = \rho\Gamma\rho^*$. Check if it is admissible as a covariance operator. If not, back to 2.
4. Simulate a H_m-white noise with covariance structure given by Γ_ϵ.
5. Use the recurrent formula (18.1) to obtain the ARH_m.

All above three strategies are implemented in the `R` package `far` (Damon and Guillas (2007)). Figure 30 shows a trajectory of an ARH process for each one of the Pumo's and Damon & Guillas variants (we abbreviate them WIEN and GH respectively). We use the same simulated strong noise process and the same input values for define the autoregression operator on H_m. The WIEN variant shows considerable rougher trajectories. This can be explained by the relative severe truncation of the covariance operator Γ used when adopting the GH variant. Indeed, when one relaxes the amount of truncation of this matrix, the obtained trajectories become similar.

19 CARH: Conditional ARH process.

While ARH processes are a natural generalization of the well known autoregressive processes in Euclidean spaces, the infinite dimension of the space H produces new challenges for their estimation and prediction. However, some of the extensions used on the scalar case has been successfully adapted to the Hilbertian framework. For example, higher order of ARH processes are studied by Pumo (1992). Mas and Pumo (2007) studies the introduction of the derivative of the lagged functional-valued variable into the autoregressive equation as an additive term. Damon and Guillas (2002) incorporates more general exogenous dependent functional-valued covariates as additive regressors modelled also by ARH.

A different approach for the introduction of exogenous information is to do it through the autoregression operator. Like in the scalar case, one may consider a more general case where the parameter ρ depends on some variable or even it may slowly vary on time. For such cases, the exogenous information is incorporated in a non-additively manner. Guillas (2002) propose to model the zero-mean functional process $(Y_k, k \in \mathbb{Z})$, setting for each $k \in \mathbb{Z}$:

$$Z_k = \rho_{V_k}(Z_{k-1}) + \epsilon_k, \tag{19.1}$$

where $V = (V_k)_{k\in\mathbb{Z}}$ is a sequence of independent identical distributed Bernoulli variables. By this way, one randomly combines two ARH processes with autoregression operators ρ_0 and ρ_1 for the two possible regimes of the dichotomic variable V_k. The resulting process admits to have one of the regimes to be explosive if it is not visited too often. If this is the case, Equation (19.1) has one unique stationary solution. The model seems appropriate to describe the trajectory of a process that may enter rarely on a nonstationary regime.

We propose a similar formulation for a new kind of process, letting V to be a multivariate random process with some continuous distribution. The process is constructed in such a way that conditionally on the exogenous variable V, the process is an ARH. We call it *Conditional Autoregressive Hilbertian process* (CARH). The model approaches the time-varying regression framework that can be founded in literature (see Hastie and Tibshirani (1993) and Wu et al. (2010) for the scalar and functional cases respectively).

We first start to define the process and the associated conditional operators of covariance and autocorelation. Then, we use the ARH classes of estimators for the conditioned autocorrelation operator. We will need to use a conditional PCA analysis for functional data.

19.1 Presentation of the model.

All the variables are defined on the same probability space $(\Omega, \mathcal{F}, \mathbb{P})$. We consider a sequence $Z = (Z_k, k \in \mathbb{Z})$ of Hilbert-random variables, i.e. each random variable Z_k is a measurable map from the probability space in an real separable Hilbert space H endowed with its Borel σ-field, $\mathcal{H}$. The space H is equipped with the scalar product $< ., . >$ and the induced norm $\|.\|_H$). Also consider the sequence of real multivariate random variables $V = \{V_k\}_{k\in\mathbb{Z}}$. Both sequences Z and V are assumed to be stationary. We will focus on the behaviour of Z conditioned to V.

The conditional expectation is characterized by the conditional distribution of Z given V, i.e. by the conditional probability $\mathbb{P}_{Z|V}$ on $\mathcal{B}$. In order this conditional probability be properly defined as a measure (in the sense that it represents a regular version of the conditional probability), it is assumed that a transition probability exists that associates to each $v \in \mathbb{R}^d$ a probability measure $\mathbb{P}^v$ on $(H, \mathcal{B})$ such that

$$\mathbb{P}^v_{Z|V}(A) = \mathbb{P}^v(A), \qquad \text{for every } A \in \mathcal{B}, v \in \mathbb{R}^d.$$

We call $\mathbb{P}^v$ the sampling measure and denote $\mathbb{E}$ the induced expectation. We restraint our attention to functions defined over a real compact interval $\mathcal{T}$ and we assume hereafter T to be $[0, 1]$ without loss of generality. More precisely, we set H to be the subspace of continuous functions on the space of classes of 4-th order $\mathbb{P}-$integrable functions.

A sequence $(Z, V) = \{(Z_k, V_k), k \in \mathbb{Z}\}$ of $H \times \mathbb{R}^d$-valued random variables is a conditional autoregressive hilbertian process (CARH) of order 1 if it is stationary and and such that, for each k

$$Z_k = a + \rho_{V_k}(Z_{k-1} - a) + \epsilon_k, \tag{19.2}$$

where the conditional mean function $a_v = \mathbb{E}^v[Z_0|V], v \in \mathbb{R}^d$, is the conditional expectation (on V) of the process, $\epsilon = \{\epsilon_k, k \in \mathbb{Z}\}$ is an Hilbertian white noise independent of V, and $\{\rho_{V_k}\}_{k\in\mathbb{Z}}$ is a sequence of random operators such that conditionally on V, ρ_V is a linear compact operator on H. Additionally, the sequence V_n is independent of the noise ϵ.

We need the following assumptions to prove the existence and uniqueness of the CARH process. Proofs are postponed until Appendix A.

Assumptions 19.1.
1. It exists a map $v \mapsto P^v$ that assigns a probability measure on $(H, \mathcal{B})$ to each value v on the support of V.
2. $\sup_n \|\rho_{V_n}\|_{\mathcal{L}} = M_\rho < 1$.

Theorem 19.2. *Under the set of Assumptions 19.1, equation* (19.2) *defines a* CARH *process with an unique stationary solution given by*

$$Z_k = a + \sum_{j=0}^{\infty} \left(\prod_{p=0}^{j-1} \rho_{V_{k-p}} \right) (\epsilon_{k-j}),$$

with the convention $\prod_{p=0}^{j-1} \rho_{V_{k-p}} = Id$ *(the identity operator) for* $j = 0$.

Let us comment the assumptions. The first point was already mentioned. The second one, certainly too strong, ensures the contraction of the conditional autoregressive operator.

In addition, we assume hereafter that finite fourth conditional moment of Z exists in order to define conditional covariance and cross-covariance operators.

The *conditional (on V) covariance* and *cross covariance operators* at the point $v \in \mathbb{R}^d$, respectively defined by

$$z \in H \mapsto \Gamma_v z = \mathbb{E}^v[(Z_0 - a) \otimes (Z_0 - a)(z)|V] \qquad \text{and}$$
$$z \in H \mapsto \Delta_v z = \mathbb{E}^v[(Z_0 - a) \otimes (Z_1 - a)(z)|V].$$

For each $v \in \mathbb{R}^d$, both operators are trace-class and hence Hilbert-Schmidt. Moreover, the operator Γ_v is positive, self-adjoint. The operators have associated kernels $\gamma(v)$ and $\delta(v)$ defined over $L^2[0,1]^2$ such that

$$\Gamma_v z(t) = \int_0^1 \gamma(v,s,t)z(t)ds,$$
$$\Delta_v z(t) = \int_0^1 \delta(v,s,t)z(t)ds, \qquad (s,t) \in [0,1]^2, v \in \mathbb{R}^d,$$

with $\gamma(v,.,.)$ continuous, symmetric and positive and $\delta(v,.,.)$ a continuous kernel. The kernels turn to be the *conditional covariance function* $\gamma(v,s,t) = \mathbb{E}^v[(Z_0(s)-a(s))(Z_0(t)-a(t))|V]$, and the *one-ahead conditional cross-covariance function* $\delta(v,s,t) = \mathbb{E}^v[(Z_0(s)-a(s))(Z_1(t)-a(t))|V]$, $(s,t) \in [0,1]^2, v \in \mathbb{R}^d$.

As for the ARH, the operators are linked by the following expression, conditionally on V at the point $v \in \mathbb{R}^d$

$$\Delta_v = \rho_v \Gamma_v.$$

To be able to use the estimation and prediction techniques that we had enumerated for the ARH processes, well behaved estimators of a_v, Γ_v and Δ_v are needed. Note that the infinite-dimension parameters to be estimated are all defined through a conditional expectation. We will use nonparametric Nadaraya-Watson like estimators. In this sense, this estimators have the same form to those proposed in Cardot (2007) for independent data. However, one important difference is that in our case neither the sequences V nor Z are assumed to be independent. We deal with their dependence through a strong mixing hypothesis. In other words, we assume each sequence to be asymptotically independent controlling the decay of the dependence. Many contexts of mixing exist in literature. In general one relies upon the decay of the dependence of the observations with respect to their time lag.

We use an α-mixing setting. Let $X = \{X_k\}_{k\in\mathbb{Z}}$ be a stationary random processes (e.g. $X = \{Z_k\}_{k\in\mathbb{Z}}$ or $\{V_k\}_{k\in\mathbb{Z}}$). Consider the $\sigma-$algebras $\mathcal{F}_{n+k} = \sigma(X_i, i > n+k)$ and $\mathcal{P}_n = \sigma(X_i, i \leq n)$ and the $\alpha-$mixing coefficients defined as

$$\alpha_X(k) = \sup_{B\in\mathcal{F}_{n+k}; C\in\mathcal{P}_n} |P(B \cup C) - P(B)P(C)|.$$

When $\lim_{k\to\infty} \alpha_X(k) = 0$ we say that X is $\alpha-$mixing. A slight weaker setting is the $2-\alpha-$mixing one, where the mixing coefficient is defined by

$$\alpha_X^{(2)}(k) = \sup_{t\in\mathbb{Z}} \alpha(\sigma(X_t), \sigma(X_{t+k})) \xrightarrow[n\to\infty]{} 0.$$

If the mixing coefficients have a geometrical decay, then the corresponding mixing process is called geometrically mixing (GSM). For example, a $2-\alpha-$mixing process verifies that $\alpha_X^{(2)}(k) \leq br^k$ for some positive b and $0 < r < 1$.

More general mixing framework adapted to nonparametric regression can be found in Nze et al. (2002). For functional data, a more restrictive setting (thus weaker dependence) based on $m-$dependence has been recently proposed on Hörmann and Kokoszka (2010) but we won't pursue this approach here.

Estimation of the mean function a**.** We want to estimate the conditional expectation function of the process $a(v,t)$ for all $t \in [0,1]$ from $\{(Z_1,V_1),\dots(Z_n,V_n)\}$.

Recall that for any $k \in \mathbb{Z}$, Z_k is a real function on $\mathcal{T} = [0,1]$. In order to properly define the framework we introduce some quantities. Fix $t \in [0,1]$ and set $Y = Z_0(t)$ and $Y_i = Z_i(t)$ for $i = 1,\dots,n$. Let us assume that V admits a density f. We define for $v \in \mathbb{R}^d$

$$g(v,t) = \mathbb{E}^v[Z_0(t)f(V)|V],$$

and provided that $f(v) > 0$ we rewrite the parameter as a regression

$$\begin{aligned} a(v,t) &= \frac{g(v,t)}{f(v)} \\ &= \frac{\mathbb{E}^v[Z_0(t)f(V)|V]}{f(v)} \\ &= \mathbb{E}^v[Y|V]. \end{aligned}$$

When $f(v) = 0$ we set $a(v,t) = \mathbb{E}[Y]$. The quantity a is called the regression parameter, i.e. the regression function of $Z_0(t)$ with respect to V. A nonparametric kernel type estimator of a can be formed by estimating f and g respectively by

$$\hat{f}_n(v) = \frac{1}{nh_a^d}\sum_{i=1}^n K(h_a^{-1}(V_i - v)) \qquad \text{and} \tag{19.3}$$

$$\hat{g}_n(v,t) = \frac{1}{nh_a^d}\sum_{i=1}^n K(h_a^{-1}(V_i - v))Y_i, \tag{19.4}$$

where $K : \mathbb{R}^d \mapsto \mathbb{R}$ is a unitary square-integrable $d-$dimensional kernel and $h_a = (h_{a,n})_{n\in\mathbb{N}}$ is a decreasing sequence of positive numbers tending to 0 called the bandwidth. Then, Nadaraya-Watson like estimator of the regression parameter is given by

$$\hat{a}_n(v,t) = \frac{\hat{g}_n(v,t)}{\hat{f}_n(v)},$$

that can be written as a weighted mean of the observed values

$$\hat{a}_n(v) = \sum_{i=1}^n w_{n,i}(v,h_a)Y_i,$$

with weights

$$w_{n,i}(v,h) = \frac{K(h^{-1}(V_i - v))}{\sum_{i=1}^n K(h^{-1}(V_i - v))}. \tag{19.5}$$

The weights are more important for those segments Z_i with closer value of V_i to the target v. If $\hat{f}_n(v) = 0$, then one usually sets the weights to be $w_{n,i}(v,h_a) = n^{-1}$ or $w_{n,i}(v) = 0$ for all $i = 1,\dots,n$ in order to define the estimator for all $v \in \mathbb{R}^d$. The bandwidth plays a key role, tuning the proximity of the scatter of $\mathbb{R}^d$ to v via the scaling

of the kernel function. Large values of h drives to a higher proximity between points, making the weights $w_{n,i}$ being no negligible for an important number of observations. Conversely, small values makes few observations to have a significant impact on the estimator. This produces the common trade-off between bias and variance of kernel regression estimators.

Additionally to Assumptions 19.1, we use the next set of assumptions to show the good pointwise asymptotic properties of the proposed estimator for a_v.

Assumptions 19.3. i. V admits a probability density function f and for each $s \neq t$, (V_s, V_t) has a density f_{V_s,V_t} such that $\sup_{|s-t|>1} \|G_{s,t}\|_\infty < \infty$ where $G_{s,t} = f_{V_s,V_t} - f \otimes f$.

ii. Both $\{Z_k\}_{k\in\mathbb{Z}}$ and $\{V_k\}_{k\in\mathbb{Z}}$ are strong mixing processes with geometrically decaying coefficients $\alpha(k) = \beta_0 e^{-\beta_1 k}$ for some $\beta_0, \beta 1 > 0$ and $k \geq 1$.

iii. $\|Z_k\|_H = M < \infty$.

iv. The kernel K is bounded symmetric density satisfying

1. $\lim_{v\to\infty} \|v\|^d_{\mathbb{R}^d} K(v) = 0$,
2. $\int_{\mathbb{R}^d} \|v\|^3_{\mathbb{R}^d} K(v)dv < \infty$,
3. $\int_{\mathbb{R}^d} |v_i||v_j| K(v)dv < \infty$ for $i, j = 1, \ldots, d$.

v. $f(.), g(., t) \in C^2_d(b)$ with $C^2_d(b)$ denoting the space of twice continuously differentiable functions z defined on $\mathbb{R}^d$ and such that

$$\left\| \frac{\partial^2 z}{\partial v_i \partial v_j} \right\|_\infty \leq b.$$

vi. $\mathbb{E}[Z_0(t)^2|V_0]f(.)$ is strictly positive, continuous and bounded at $v \in \mathbb{R}^d$.

Proposition 19.4. *Under Assumptions 19.1 and 19.3, for a bandwidth verifying* $h_{a,n} = c_n \left(\frac{\ln n}{n}\right)^{1/(d+4)}$, $c_n \to c > 0$, *when* $n \to \infty$, *we have*

1.

$$\widehat{f}_n(v) - f(v) = \mathcal{O}\left(\left(\frac{\ln n}{n}\right)^{\frac{2}{4+d}}\right) \qquad a.s., \tag{19.6}$$

2.

$$\widehat{a}_n(v,t) - a(v,t) = \mathcal{O}\left(\left(\frac{\ln n}{n}\right)^{\frac{2}{4+d}}\right) \qquad a.s.$$

Let us comment the assumption for this result. The density assumption in 19.3(**i.**) may be drop if one uses a more general framework like in Dabo-Niang and Rhomari (2009) where no density assumption is done and the observations are independent. However, similar results for dependent data are not available yet. The mixing hypothesis will allow us to control the variance of estimators. We ask on the kernel K mild conditions, that are usual for such problems. All symmetric kernels defined over a compact support verifies the hypothesis, but also more general ones like the Gaussian kernel. Assumption 19.3(**v.**) is used to control the bias terms of the estimators that is purely analytical.

Note that only the observations of V are used to estimate the density f. This is a well know result about estimation of multidimensional density functions, even for

dependent data. We include it for sake of comprehension. In particular, the consistency result for $\widehat{f}_n$ is true for each $t \in [0,1]$. However, the consistency of $\widehat{a}_n(v,t)$ was obtained point-wisely for some fixed value of $t \in [0,1]$. We can obtain a version of this result that holds true uniformly on $[0,1]$, that is conditionally on V, the mean function is consistently estimated by $\widehat{a}_n(v,.), v \in \mathbb{R}^d$.

Proposition 19.5. *Under Assumptions 19.1, 19.3**(i-iii)** and if 19.3**(iv-vi)** hold true for all $t \in [0,1]$, a bandwidth verifying $h_{a,n} = c_n \left(\frac{\ln n}{n}\right)^{1/(d+4)}$, $c_n \to c > 0$, when $n \to \infty$, yields*

$$\|\widehat{a}_n(v,.) - a(v,.)\|_H = \mathcal{O}\left(\left(\frac{\ln n}{n}\right)^{\frac{2}{4+d}}\right) \qquad a.s.$$

As usual the speed of convergence rapidly degrades with the raise of the dimension of $\mathbb{R}^d$, the space where V takes its values. This is a well known phenomena called *curse of dimensionality*, associated to the difficulty in finding close points in high dimensional spaces.

Estimation of the conditional covariance operator Γ_v. The estimation of the covariance operator is made via the estimation of the associated kernel. We follow the same principle adopted for the mean function estimation to estimate the kernel. Let us first assume that the process Z is centred, then we fix the couple $(s,t) \in [0,1]^2$ and define the real valued variables $Y = Z_0(s)Z_0(t)$ and the "observations" $Y_i = Z_i(s)Z_i(t)$ with $i = 1, \ldots, n$. Then, one defines the auxiliary quantities f and g and their estimators as before, obtaining an estimator of the conditional second order moment function $r(v)$ that can be written as

$$\hat{r}_n(v) = \sum_{i=1}^{n-1} w_{n,i}(v, h_\gamma) Y_i,$$

with the bandwidth $h_\gamma = (h_{\gamma,n})$. Besides Y and $Y_i, i = 1, \ldots, n$ being redefined, the only difference respect to the estimator of the conditional mean function is on the bandwidth used inside the weights. Moreover, f is strictly identical to the one defined before. This because it only depends on the covariate V. The bandwidths h_γ are h_a may be different.

An analogous results on the strong consistency property can be obtained for $\hat{\gamma}_n$.

Proposition 19.6. *Under Assumptions 19.1 and 19.3, and if $\mathbb{E}[\|Z\|_H^4 | V] < \infty$ then for a bandwidth verifying $h_{\gamma,n} = c_n \left(\frac{\ln n}{n}\right)^{1/(d+4)}$, $c_n \to c > 0$, when $n \to \infty$, we have*

$$\widehat{r}_n(v,t,s) - r(v,t,s) = \mathcal{O}\left(\left(\frac{\ln n}{n}\right)^{\frac{2}{4+d}}\right) \qquad a.s..$$

Again, the result is valid uniformly for $(t,s) \in [0,1]^2$. Let us come back to the more general case where Z has a non zero mean a. Through the equivalence between Hilbert-Schmidt norm and the integral operator norm (on $L_2([0,1]^2)$) one has,

$$\begin{aligned}\|\hat{\Gamma}_{v,n} - \Gamma_v\|_{\mathcal{K}_2} =& \|\hat{\gamma}_n(v,.,.) - \gamma(v,.,.)\|^2_{L^2([0,1]^2)} \\ =& \int_0^1 \int_0^1 (\hat{\gamma}_n(v,s,t) - \gamma(v,s,t))^2 ds dt\end{aligned}$$

and thus the strong consistency of Γ_n^v.

Proposition 19.7. *Under Assumptions 19.1, 19.3**(i-iii)** and if 19.3**(iv-vi)** hold true for all $t \in [0,1]$, and $\mathbb{E}[\|Z\|_H^4|V] < \infty$, then a bandwidth verifying $h_{\gamma,n} = c_n \left(\frac{\ln n}{n}\right)^{1/(d+4)}$, $c_n \to c > 0$, when $n \to \infty$, yields*

$$\|\hat{\Gamma}_{v,n} - \Gamma_v\|_{\mathcal{K}_2} = \mathcal{O}\left(\left(\frac{\ln n}{n}\right)^{\frac{2}{4+d}}\right) \qquad a.s..$$

Now, one may use the consistency properties of the empirical eigenvalues $\hat{\lambda}_{j,n}(v)$ as estimators of the true ones $\lambda_j(v), j \geq 1$, obtained by Bosq (2000) on the dependent case. Also a result concerning the eigenfunctions $\hat{e}_{j,n}(v)$ is provided.

Corollary 19.8. *Under the conditions of Proposition 19.7, we have*

1.

$$\sup_{j\geq 1} |\widehat{\lambda}_{j,n}(v) - \lambda_j(v)| = \mathcal{O}\left(\left(\frac{\ln n}{n}\right)^{\frac{2}{4+d}}\right) \qquad a.s.$$

2.

$$\|e'_{j,n}(v) - e_j(v)\|_H = \xi_j \mathcal{O}\left(\left(\frac{\ln n}{n}\right)^{\frac{2}{4+d}}\right) \qquad a.s.$$

where $e'_{j,n}(v) =< \hat{e}_{j,n}(v), e_j(v) >_H \hat{e}_{j,n}(v)$ and $\xi_1 = 2\sqrt{2}/(\lambda_1 - \lambda_2)$, $\xi_j = 2\sqrt{2}/\min(\lambda_{j-1} - \lambda_j, \lambda_j - \lambda_{j-1})$ for $j \geq 2$.

Remark that the conditional eigenfunctions are estimated up to their sign. This may cause a problem both on practice and theory. Actually, the true parameter is the eigenspace generated by the eigenfunction and not its direction. See Mas and Menneteau (2003) for a generalization of a transfer approach of limit theorem properties and modes of convergence of the estimator of an operator to the estimators of its eigenvalues or its eigenspaces.

Estimation of the conditional cross-covariance operator Δ_v. The estimation of the conditional operator is done through the estimation of its kernel, the conditional cross covariance function. Hence, fix $(s,t) \in [0,1]^2$ and take $Y^v = (Z_0(s) - a_v(s))(Z_1(t) - a_v(t))$ and the observations $Y_i^v = (Z_{i-1}(s) - \hat{a}_n(s))(Z_i(t) - \hat{a}_n(t))$ for $i = 2, \ldots, n$. Define f and g and their estimators with the same form of (19.3) and (19.4) respectively using the bandwidth h_δ and the redefined Y and Y_i.

Proposition 19.9. *Under Assumptions 19.1, 19.3**(i-iii)** and if 19.3**(iv-vi)** hold true for all $t \in [0,1]$, and $\mathbb{E}[\|Z\|_H^4|V] < \infty$, then a bandwidth verifying $h_n = c_n \left(\frac{\ln n}{n}\right)^{1/(d+4)}$, $c_n \to c > 0$, when $n \to \infty$, yields*

$$\|\hat{\Delta}_{v,n} - \Delta_v\|_{\mathcal{K}_2} = \mathcal{O}\left(\left(\frac{\ln n}{n}\right)^{\frac{2}{4+d}}\right) \qquad a.s..$$

Choice of the bandwidths. The practical performance of any kernel regression heavily depends on the appropriate calibration of the bandwidths. As mentioned before, this is associated to a trade-off between bias and variance of the estimators. Cross validation techniques are usually used on practice for independent data. As our final task is to predict, the temporal aspect on the stochastic processes (and their possible lack of stationarity) should be considered on the calibration. Antoniadis et al. (2009) proposed to chose the bandwidth for functional time series prediction. For this purpose, an empirical risk of prediction is calculated from past values of the process for a range of bandwidths within a given grid of possible values. The value of the bandwidth that minimizes the empirical risk is then the value used for the prediction.

19.2 Prediction of a CARH process.

The intrinsic infinite dimension of the space challenges the estimation of the conditional autocorrelation operator in the same way it does it for the ARH process. The same remarks mentioned above when we inverted Γ for estimate ρ on the ARH processes hold true for the inversion conditional covariance opeator Γ_v. If one follows Bosq (2000), the functional observations may be projected onto the subspace, $H_v^{k_n}$, formed by the first k_n eigenfunctions.

In our case, the eigenfunctions represents the principal direction of variation contained on the conditional covariance operator. Let us call $P_v^{k_n}$ the projection operator from H to $H_v^{k_n}$. Then, we define the *projection estimator* of ρ_v^* by

$$\widehat{\rho}_{v,n}^* = (P_v^{k_n}\widehat{\Gamma}_{v,n}P_v^{k_n})^{-1}\widehat{\Delta}_{v,n}^* P_v^{k_n}. \tag{19.7}$$

As before, be can also define a resolvent estimator by

$$\widehat{\rho}_{v,n,p}^* = b_{n,p}(\widehat{\Gamma}_{v,n})\widehat{\Delta}_{v,n}^*, \tag{19.8}$$

which is parametrized by $p \geq 0$ and $b > 0$. The same kind of remarks done for the ARH case stands here.

Any of the proposed estimators can be use to predict Z_{n+1} by applying them to Z_n. However, since Z_n was used on the construction of the estimator, it is a better approach to evaluate the predictor on the following element of the sequence. In this sense, we obtain the convergence on probability of the proposed predictors.

Assumptions 19.10.

1. $\mathbb{E}[\|Z\|_H^4|V] < \infty$,
2. Γ_v is one-to-one,
3. $\mathbb{P}(\liminf \mathcal{E}_n) = 1$, where $\mathcal{E}_n = \{\omega \in \Omega : \dim R(P_v^{k_n}\widehat{\Gamma}_{v,n}P_v^{k_n}) = k_n\}$,
4. $n\lambda_{k_n}^4(v) \to \infty$ and $(1/n)\sum_{j=1}^{k_n} \xi_k(v)/\lambda_k^2(v) \to 0$, as $n \to \infty$.

Finite fourth conditional moment of Z is needed for defining Γ_V. The second point in 19.10 is necessary for uniquely define the conditional autoregression operator ρ_v. Third point is necessary to assure the existence of the inversion of the random operator $P_v^{k_n}\widehat{\Gamma}_{v,n}P_v^{k_n}$. Controlling the decay of the eigenvalues of the conditional covariance operator is used for the consistency of the projection class operator (see Corollary 19.8

for the definition of ξ). Alternatively, one may set $\Lambda_v(k) = \lambda_k(v)$ where $\Lambda_v : \mathbb{R} \to \mathbb{R}$ is a convex function (see Mas (2007)).

The notation $\xrightarrow{P}$ on the next theorems stands for convergence in probability.

Theorem 19.11. *Under assumptions 19.1, 19.3 holding true uniformly over* $[0,1]$ *and 19.10, if* $\lambda_k(v) = c_0 c_1^k, c_0 > 0, c_1 \in (0,1)$ *and if* $k_n = o(\ln n)$ *as* $n \to \infty$, *then*

$$\|\widehat{\rho}^*_{v,n}(Z_{n+1}) - \rho^*(Z_{n+1})\|_H \xrightarrow{P} 0$$

Theorem 19.12. *Under assumptions 19.1, 19.3 holding true uniformly over* $[0,1]$ *and 19.10(**i-ii**), and if* $b_n \to 0, b_n^{p+2}\sqrt{n} \to \infty$ *for some* $p \geq 0$ *as* $n \to \infty$, *then*

$$\|\widehat{\rho}^*_{v,n,p}(Z_{n+1}) - \rho^*(Z_{n+1})\|_H \xrightarrow{P} 0$$

The above results are also valid to the best prediction $\rho(Z_{n+1})$ using the continuity extension property (see Section 18).

20 Empirical study

On this section, we carry out some numerical experiences to show the prediction performance of the proposed estimators for the CARH process. For this purpose, we first introduce a simulation scheme inspired from the ARH framework.

20.1 Simulation of a CARH.

We modified some of the routine from `far` package (Damon and Guillas (2007)) to simulate a CARH process. For the numerical experiences, we use a Beta(β_1, β_2) distribution to create an i.i.d. sequence V. Since the Beta is the unit interval, the the CARH process exists and it is stationary. The simulation uses a Wiener process for the noise and a symmetric autocorrelation operator. By this way, one can use the eigenfunctions g_j issued from the Karhunen-Loève decomposition of the noise structure to write down the autocorrelation operator (see Pumo (1992)).

We use an iterative algorithm to simulate a CARH trajectory of size n. For the step $k = 1, \ldots, n + n0$,

1. Simulate a Wiener process ϵ_k.
2. Draw V_k following a Beta(β_1, β_2) distribution. Call v_k its realization.
3. Compute the eigenvalues $\lambda_j(v_k) = \lambda_j v_k$ to define $\rho_{v_k} = \sum_j \lambda_j(v_k)(g_j \otimes g_j)$.
4. Compute $Z_k = \rho_{v_k} Z_{k-1} + \epsilon_k$.

To initiate the algorithm Z_0 is drawn from a strong white noise. At the end the first $n0$ steps are burned out.

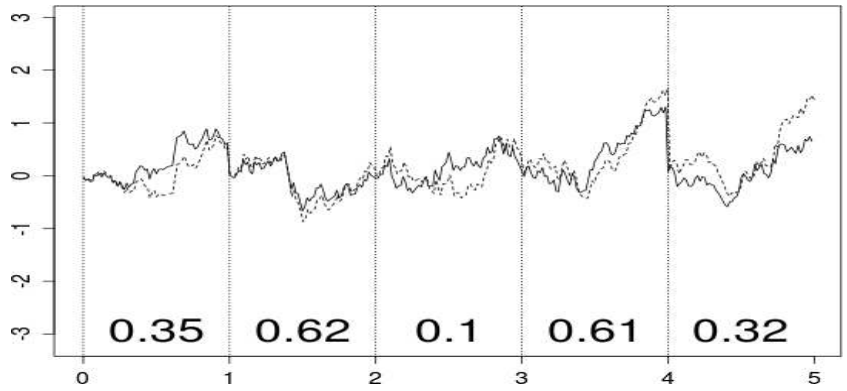


Figure 31: A 5 blocs sampling of a simulated CARH (continuous line) and ARH (dashed line) processes. Values below graphics are the realization of the covariate V of the CARH process.

20.2 Parameters used on simulation.

We simulate a $n = 100$ length trajectory of a CARH process using $m = 64$ points per function. We choose a Beta distribution with parameters $(\beta_1, \beta_2) = (4/5, 4/5)$. The operator ρ on the space of dimension $k = 4$ generated by the first eigenfunction $g_j, j = 1, \ldots, 4$ is a diagonal matrix with elements $(0.672, 0.228, 0.9, 0.34)$. The first $n_0 = 5$ elements of the simulation are burned out. Some comments about the values of (β_1, β_2) to justify our choice

- If $\beta_1, \beta_2 > 1$ the Beta distribution is unimodal and for large values of the parameters the variance may be small. Thus, the CARH process is close to an ARH.
- If $\beta_1 = \beta_2 = 1$ one has the uniform distribution.
- If $\beta_1 = \beta_2 < 1$ the density function is U shaped. The CARH process may be close to a dichotomic ARH (as in Guillas (2002)).

The result of the simulation is an object with two components: the simulated trajectory of Z and the sequence of V used to generate the CARH process. Five consecutive curves issued from the simulation are plotted on Figure 31. The figure also shows the trajectory of an ARH process simulated with the same noise as for the CARH process.

20.3 Prediction of a CARH.

We use the projection class and resolvent class predictors, respectively (19.7) and (19.8), to obtain predictions of the simulated CARH process.

In order to calibrate the parameters involved on each estimators, we separate the data in two parts. With the first part, we calibrate the bandwidth parameters h_a, h_γ, h_δ, the dimension k_n for the projection class, and the regularization parameters α, p for the resolvent class. For this, we predict at horizon 1 each one of the observations of the last fifth of the sample. A square loss function of the prediction error is computed, using a grid of possible values for the parameters.

The bandwidth parameter values that minimize the loss function are $(h_a, h_\gamma, h_\delta) = (0.01, 0.01, 0.01)$ for the projection class estimator and $(h_a, h_\gamma, h_\delta) = (0.162, 0.086, 0.124)$ for the resolvent class estimator. We found also $k_n = 4$, and 0.3775 for α and $p = 2$.

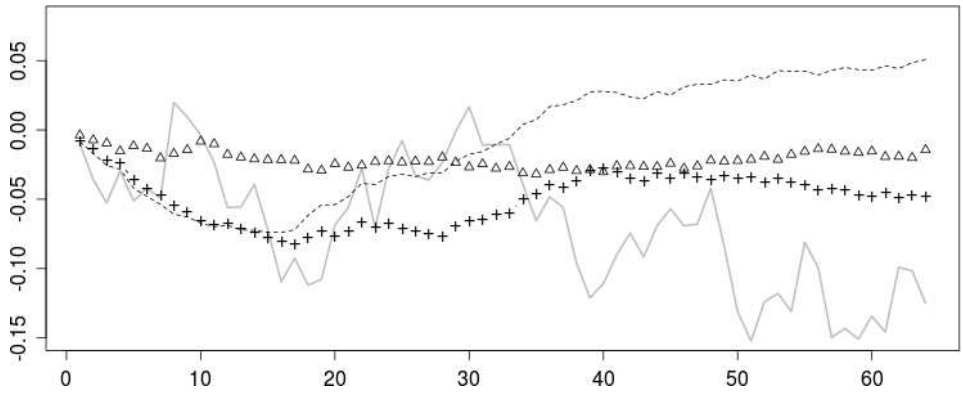


Figure 32: Prediction of one simulated curve of an CARH process (full line) using projection class (+) and resolvent class (△) predictors for CARH process, and projection class predictor for ARH process (dashed line).

The mean prediction error for the last 10 observations is slightly better for the resolvent class (0.1472 measured as root mean square error, RMSE) with a value of 0.1544 for the projection class estimator. Figure 32 shows the trajectory of one of the observations with the predictions issued from both predictors. We also include the prediction of the projection class estimator (see Equation 18.8) obtained from the simulated data without using the observations for V. That is, we neglect the fact that the observations come from a CARH process, assuming instead they come from an ARH process. Our CARH orientated predictors seems to have a better performance the ARH predictor. This is confirmed by the mean prediction error over the last 10 observations for which the ARH based predictor obtains an RMSE of 0.1714.

A Sketch of proofs.

Proof of Theorem 19.2.

We mimic the proof of Theorem 1 in Guillas (2002). To prove the existence, Let

$$\begin{aligned}
\eta_m^{m'} &= \mathbb{E}\left\|\sum_{j=m}^{m'}\left(\prod_{p=0}^{j-1}\rho_{V_{n-p}}\right)(\epsilon_{n-j})\right\|_H^2 \\
&= \sum_{j=m}^{m'}\mathbb{E}\left\|\left(\prod_{p=0}^{j-1}\rho_{V_{n-p}}\right)(\epsilon_{n-j})\right\|_H^2 \\
&\leq \sum_{j=m}^{m'}\mathbb{E}\left[\left\|\prod_{p=0}^{j-1}\rho_{V_{n-p}}\right\|_{\mathcal{L}}^2\|\epsilon_{n-j}\|_H^2\right] \\
&\leq \sum_{j=m}^{m'}\mathbb{E}\left[\prod_{p=0}^{j-1}\left\|\rho_{V_{n-p}}\right\|_{\mathcal{L}}^2\right]\underbrace{\mathbb{E}\|\epsilon_{n-j}\|_H^2}_{=\sigma^2}
\end{aligned}$$

where we used the independence between V and ϵ gives

$$\mathbb{E}\left\langle \left(\prod_{p=0}^{j-1}\rho_{V_{n-p}}\right)(\epsilon_{n-j}), \left(\prod_{p=0}^{j'-1}\rho_{V_{n-p}}\right)(\epsilon_{n-j'})\right\rangle = 0, \qquad \text{for } j \neq j'.$$

Finally, we obtain

$$\eta_m^{m'} \leq \sigma^2\mathbb{E}\left[\prod_{p=0}^{j-1}\left\|\rho_{V_{n-p}}\right\|_{\mathcal{L}}^2\right] \leq \sigma M_\rho^{2j}.$$

We have that the upper bound is the general term of a convergent series. For m, m' tending to infinity, $\eta_m^{m'}$ tend to zero and the Cauchy criterion gives the mean square convergence of the solution.

Now, consider the stationary process $W_n = a + \sum_{j=0}^{\infty}\left(\prod_{p=0}^{j-1}\rho_{V_{n-p}}\right)(\epsilon_{n-j})$. From the almost surely boundedness of ρ_{V_n} we have that it is indeed a solution of the CARH process:

$$\begin{aligned}
(W_n - a) - \rho_{V_n}(W_{n-1} - a) &= \sum_{j=0}^{\infty}\left(\prod_{p=0}^{j-1}\rho_{V_{n-p}}\right)(\epsilon_{n-j}) - \sum_{j=0}^{\infty}\rho_{V_n}\left(\prod_{p=0}^{j-1}\rho_{V_{n-1-p}}\right)(\epsilon_{n-1-j}) \\
&= \sum_{j=0}^{\infty}\left(\prod_{p=0}^{j-1}\rho_{V_{n-p}}\right)(\epsilon_{n-j}) - \sum_{j=0}^{\infty}\left(\prod_{p=0}^{j}\rho_{V_{n-p}}\right)(\epsilon_{n-1-j}) \\
&= \sum_{j=0}^{\infty}\left(\prod_{p=0}^{j-1}\rho_{V_{n-p}}\right)(\epsilon_{n-j}) - \sum_{j'=1}^{\infty}\left(\prod_{p=0}^{j'-1}\rho_{V_{n-p}}\right)(\epsilon_{n-j'}) \\
&= \epsilon_n.
\end{aligned}$$

Proof of Proposition 19.4

The proof is based on the classical decomposition in terms of bias and variance of the estimators. The bias term is purely analytical. The variance term is composed by the variance and covariance of the estimator's terms. The dependency of the data is controlled by means of the following exponential inequality (a proof can be founded in Bosq and Blanke (2007, p. 140)).

Lemma A.1. *Let $W = (W_t)$ be a zero-mean real valued stationary process with $\sup_{1\leq t\leq n}\|W_t\|_\infty = M < \infty$, $(M > 0)$. Then for $q \in [1, n/2]$, $\kappa > 0$, $\epsilon > 0$, $p = n/(2q)$,*

$$\mathbb{P}\left(\left|\sum_{i=1}^{n} W_i\right| > n\epsilon\right) < \frac{8M}{\epsilon\kappa}(1+\kappa)\alpha_X\left(\left[\frac{n}{2q}\right]\right) + 4\exp\left(-\frac{n^2\epsilon^2/q}{8(1+\kappa)\sigma(q) + \frac{2M}{3}(1+\kappa)n^2q^{-2}\epsilon}\right), \quad \text{(A.1)}$$

with $\sigma(q)$ an intricate quantity involving the pairwise covariances of W. We will only need a bound of $\sigma(q)$ that in the stationary case turns to be

$$\sigma(q) < ([p]+2)(\mathrm{Var}(W_0) + 2\sum_{l=1}^{[p]+1}|\,\mathrm{Cov}(W_0, W_l)|). \quad \text{(A.2)}$$

Proof of 1. One has

$$\mathbb{E}\widehat{f}_n(v) - f(v) = \int_{\mathbb{R}^d} K(u)(f(v - h_n u) - f(v))du$$

Using Taylor formula and the symmetry of K one gets

$$\mathbb{E}\widehat{f}_n(v) - f(v) = \frac{h_n^2}{2}\int_{\mathbb{R}^d} K(u)\left(\sum_{i,j=1}^{d} u_i u_j \frac{\partial^2 f}{\partial v_i \partial v_j}(v - \theta h_n u)\right) du$$

where $0 < \theta < 1$. Finally, Lebesgue dominated convergence theorem gives

$$h_n^{-2}|\mathbb{E}\widehat{f}_n(v) - f(v)| \to b_2(v) = 1/2\left(\sum_{i,j=1}^{d} \frac{\partial^2 f}{\partial v_i \partial v_j}(v)\int_{\mathbb{R}^d} u_i u_j K(u)du\right) \tag{A.3}$$

We use (A.1) to deal with the variance term $\widehat{f}_n(v) - \mathbb{E}\widehat{f}_n(v)$. Define $W_i = K_h(v - V_i) - \mathbb{E}K_h(v - V_i)$, with $K_h(.) = K(./h)$. Then, $M = 2h_n^{-d}\|K\|_\infty$. Let us choose $q_n = \frac{n}{2p_0 \ln n}$ for some $p_0 > 0$. Which yields on a logarithmic order for $p_n = \frac{1}{p_0 \ln n}$. This choices and the boundeness of f and $G_{s,t}$ entail on A.2,

$$\begin{aligned}\sigma(q_n) &< (p_n + 2)\operatorname{Var}(K_h(v - V_1)) + (p_n + 2)^2 \sup_{|s-t|>1}\|G_{s,t}\|_\infty \\ &< p_n h_n^{-d}\|K\|_2^2 f(v)(1 + o(1)).\end{aligned}$$

Now take $\epsilon = \eta\sqrt{\frac{\ln n}{nh_n^d}}$, for some $\eta > 0$, then

$$\begin{aligned}\mathbb{P}\left(n^{-1}\left|\sum_{i=1}^{n} W_i\right| > \eta\sqrt{\frac{\ln n}{nh_n^d}}\right) < \frac{8\beta_0 c^{d/2}}{\eta\kappa}(1 + \kappa)\|K\|_\infty \frac{n^{\frac{2+d}{4+d}} - \beta_1 p_0}{(\ln n)^{\frac{2+d}{4+d}}} \\ + 4\exp\left(-\frac{\eta^2 \ln n}{4(1 + \kappa)^2\|K\|_2^2 f(v)(1 + o(1))}\right)\end{aligned}$$

If we take $\eta > 2(1 + \kappa)\|K\|_2\sqrt{f(v)}$ and $p_0 > 2\beta$, then where both terms are $o(n^{-\lambda})$, for some $\lambda > 0$, in which case

$$\sum_n \mathbb{P}\left\{\left(\frac{n}{\ln n}\right)^{\frac{2}{4+d}}\left|n^{-1}\sum_{i=1}^{n} W_i\right| > \eta c_n^{-d/2}\right\} < \infty.$$

So Borel-Cantelli lemma implies $\limsup_{n\to+\infty}\left(\frac{n}{\ln n}\right)^{\frac{2}{4+d}}|\widehat{f}_n(v) - \mathbb{E}\widehat{f}_n(v)| \leq 2c_n^{-d/2}(1 + \kappa)\|K\|_2\sqrt{f(v)}$ almost surely for all $\kappa > 0$. We have finally

$$\limsup_{n\to+\infty}\left(\frac{n}{\ln n}\right)^{\frac{2}{4+d}}|\widehat{f}_n(v) - f_(v)| \leq 2c_n^{-d/2}\|K\|_2\sqrt{f(v)} + c^2|b_2(v)|,$$

which gives (19.6).

Proof of 2. We use the following decomposition, omitting the argument v,

$$\widehat{a}_n - a = \frac{\widehat{g}_n - a\widehat{f}_n}{\widehat{f}_n}.$$

From (19.6) we have for the denominator that $\widehat{f}_n \to f(x) \neq 0$ almost surely. We work out the numerator through the following decomposition between variance and bias terms. Let $\psi_n = (n/\ln n)^{2/(4+d)}$, then one has

$$\psi_n|\widehat{g}_n - a\widehat{f}_n| \leq \underbrace{\psi_n|\widehat{g}_n - a\widehat{f}_n - \mathbb{E}(\widehat{g}_n - a\widehat{f}_n)|}_{:=A_n} + \underbrace{\psi_n|\mathbb{E}(\widehat{g}_n - a\widehat{f}_n)|}_{:=B_n}.$$

We first study A_n using as before the exponential type inequality (A.1) with the redefined random variables

$$W_i = K_h(v - V_i)(Y_i - a_v) - \mathbb{E}\left(K_h(v - V_i)(Y - a_v)\right)$$

with the precedent choices of q_n and p_n. First, one has $|W_i| \leq 2h_n^{-d}\|K\|_\infty(1 + o(1))$. Next, using Bochner lemma (Bosq and Blanke (2007, p. 135)) we obtain

$$h_n^d \operatorname{Var}(W_1) \leq h_n^{-d}\mathbb{E}[K_h^2(v - V_1)(Y_1 - a_v)^2] \to f(v)\|K\|_2^2\Sigma(v)$$

where $\Sigma(v) = (\mathbb{E}^v[Y_0^2|V] - a_v)$ is the conditional variance parameter. The logarithmic order of p_n and the control on F gives $\sigma^2(q_n) \leq p_n h_n^{-d} f(v) 2\Sigma(v)\|K\|_2^2(1 + o(1))$. As before, taking $p_0 > 2/\beta_1$ and for a large enough η, Borel-Cantelli lemma entails

$$\limsup_{n\to\infty} A_n \leq 2c^{-d/2}\sqrt{\Sigma(v)f(v)} \qquad \text{a.s.}$$

For the bias term we write

$$\mathbb{E}(\widehat{g}_n(v) - a(v)\widehat{f}_n(v)) = h_n^{-d}\int_{\mathbb{R}^d} K_{h_n}(v - t)(g(t) - f(t)a(v))dt.$$

Then, we use the Taylor formula to expand $g(t) - f(t)a(v)$ and Assumptions 19.3(**iii-iv**) to obtain

$$\psi_n|B_n| \to b_a(v) = \frac{1}{2}\left|\sum_{i,j=1}^{d}\left\{\frac{\partial^2 g}{\partial v_i \partial v_j}(v) - a(v)\frac{\partial^2 f}{\partial v_i \partial v_j}(v)\right\}\int u_i u_j K(u)du\right|$$

Finally, putting all the elements together one obtains,

$$\limsup_{n\to\infty}\left(\frac{n}{\ln n}\right)^{\frac{2}{4+d}}|\widehat{a}_n(v) - a(v)| \leq 2c^{-d/2}\|K\|_2\sqrt{f(v)\Sigma(v)} + c^2\frac{|b_a(v)|}{f(v)} \tag{A.4}$$

from with the result is derived.

Proof of Proposition 19.5.

The only terms on equation A.4 that depends on the value fixed for t are the conditional variance parameter Σ and the bias b_a. With the new hypothesis holding uniformly, for each $v \in \mathbb{R}^d$, $\Sigma(v,t)$ and $b_a(v,t)$ are bounded uniformly on $[0,1]$. Then, recalling that

$$\|\widehat{a}_n(v,.) - a(v,.)\|_H^2 = \int_0^1 (\widehat{a}_n(v,t) - a(v,t))^2 dt,$$

we obtain the wanted result.

Proof of Proposition 19.6.

The proof follows the same guidelines that those used to show Proposition 19.4(2). In particular,

$$\widehat{r}_n - r = \frac{\widehat{g}_n - r\widehat{f}_n}{\widehat{f}_n}.$$

gives the decomposition between variance and bias terms,

$$\psi_n|\widehat{g}_n - r\widehat{f}_n| \leq \underbrace{\psi_n|\widehat{g}_n - r\widehat{f}_n - \mathbb{E}(\widehat{g}_n - r\widehat{f}_n)|}_{:=A_n} + \underbrace{\psi_n|\mathbb{E}(\widehat{g}_n - r\widehat{f}_n)|}_{:=B_n}.$$

Which yields on

$$\limsup_{n\to\infty} A_n \leq 2c^{-d/2}\sqrt{\Sigma(v)f(v)} \qquad \text{a.s.}$$

where, by the redefinition of Y, $\Sigma(v) = \mathbb{E}^v[(Z_0(s)Z_0(t))^2|V] - r(v,s,t)$.

Again using Taylor formula to expand $g(t) - f(t)r(v)$ and the precedent Assumptions we obtain

$$\psi_n|B_n| \to b_r(v) = \frac{1}{2}\left|\sum_{i,j=1}^{d}\left\{\frac{\partial^2 g}{\partial v_i \partial v_j}(v) - r(v)\frac{\partial^2 f}{\partial v_i \partial v_j}(v)\right\}\int u_i u_j K(u)du\right|$$

Finally, resembling the terms we get the equivalent of Equation (A.4) with the redefined Σ and the bias b_r, from with the result is derived.

Proof of Proposition 19.7.

First, consider the following decomposition

$$\widehat{\Gamma}_{v,n} = \widehat{R}_n(v) - \widetilde{a}_n(v)\otimes\widehat{a}_n(v) - \widehat{a}_n(v)\otimes\widetilde{a}_n(v) + \widehat{a}_n(v)\otimes\widehat{a}_n(v),$$

where $\widehat{R}_n(v) = \sum_{i=1}^n w_{n,i}(v,h_\gamma)Z_i\otimes Z_i$ is the empirical counterpart of the second order moment operator $R(v) = \mathbb{E}^v[Z_0\otimes Z_0|V]$, and $\widetilde{a}_n(v) = \sum_{i=1}^n w_{n,i}(v,h_\gamma)Z_i$. Second, we obtain that

$$\Gamma_v - \widehat{\Gamma}_{v,n} = R(v) - \widehat{R}_n(v) - a(v)\otimes a(v) + \widetilde{a}_n(v)\otimes\widehat{a}_n(v) + \widehat{a}_n(v)\otimes\widetilde{a}_n(v) - \widehat{a}_n(v)\otimes\widehat{a}_n(v).$$

Hence, we can control the estimation error regrouping the terms of the above decomposition (we drop the argument v),

$$\|\Gamma_v - \widehat{\Gamma}_{v,n}\|_{\mathcal{K}_2} = \|R - \widehat{R}_n\|_{\mathcal{K}_2} + \|\widetilde{a}_n\otimes\widehat{a}_n - a\otimes a\|_{\mathcal{K}_2} + \|\widehat{a}_n\otimes(\widetilde{a}_n - \widehat{a}_n)\|_{\mathcal{K}_2}. \qquad \text{(A.5)}$$

From Propositions 19.6 and 19.5 it follows that

$$\|R - \widehat{R}_n\|_{\mathcal{K}_2} = \mathcal{O}\left(\left(\frac{n}{\ln n}\right)^{\frac{2}{4+d}}\right) \qquad \text{a.s.}$$

The second term of the left hand side of equation (A.5) is equal to

$$\|\widetilde{a}_n\otimes(\widehat{a}_n - a) + (\widetilde{a}_n - a)\otimes a\|_{\mathcal{K}_2} \leq \|\widetilde{a}_n\|_H\|\widehat{a}_n - a\|_{\mathcal{K}_2} + \|\widetilde{a}_n - a\|_{\mathcal{K}_2}\|a\|_H.$$

Since both $\|a\|_H$ and $\|\widetilde{a}_n\|_H$ are bounded and using Proposition 19.5 successively for $\widetilde{a}_n$ and $\widehat{a}_n$ with their respective sequences of bandwidths $h_{\gamma,n}$ and $h_{a,n}$, we obtain that

$$\|\widetilde{a}_n \otimes \widehat{a}_n - a \otimes a\|_{\mathcal{K}_2} = \mathcal{O}\left(\left(\frac{n}{\ln n}\right)^{\frac{2}{4+d}}\right) \qquad \text{a.s.}$$

With a similar reasoning, the same kind of result is obtained for the third term in (A.5). Putting the result for the three terms together conclude the proof.

Proof of Corollary 19.8.

First item is a direct consequence of the following property on eigenvalues of compact linear operators Bosq (2000, p. 104),

$$\sup_{j\geq 1} |\lambda_j(v) - \hat{\lambda}_{j,n}(v)| \leq \|\Gamma_v - \widehat{\Gamma}_{v,n}\|_{\mathcal{L}},$$

and the asymptotic result obtained for $\|\Gamma_v - \widehat{\Gamma}_{v,n}\|_{\mathcal{K}_2}$.

For the second item, Bosq (2000, Lemma 4.3) shows that, for each $j \geq 1$,

$$\|e_j(v) - e'_{j,n}(v)\|_H \leq \xi_j \|\Gamma_v - \widehat{\Gamma}_{v,n}\|_{\mathcal{L}}.$$

Again, the rates of convergence follows from Proposition 19.7.

Proof of Proposition 19.9.

The proof follows the same guidelines that those of Proposition 19.7, replacing $\hat{R}(v)$ and $R(v)$ by $\hat{R}_1(v) = \sum_{i=1}^{n-1} w_{n,i}(v, h_\gamma) Z_i(s) Z_{i+1}(t)$ and $R_1(v) = \mathbb{E}^v[Z_0(s)Z_1(t)|V]$ respectively. Then, a decomposition like A.5 and the same kind of observations done for that proof entails the result.

Proof of Theorem 19.11.

The proof follows along the same lines of Proposition 4.6 in Bosq (1991) by using Propositions 19.5, 19.7, 19.9 and Corollary 19.8.

Proof of Theorem 19.12.

The proof follows along the same lines of Proposition 3 in (Mas, 2000, Chapter 3) by using Propositions 19.5, 19.7 and 19.9.

Annexes

A The `wavkerfun` package.

The following pages are the documentation files of the package `wavkerfun` that we write to perform the analysis exposed on the first part of this thesis.

The package will continue to be developed to include some of the lines of work mentioned on the conclusion and future work sections (e.g. the computation of a confidence tube for the predictions). The code is inspired from an early version of package `far`.

Package 'kerwavfun'

September 11, 2011

Type Package

Title Non parametric forecasting of functional-valued processes using the Discrete Wavelet Transform.

Version 0.6

Date 2011-06-06

Author Jairo Cugliari

Maintainer Jairo Cugliari <jairocugliari@gmail.com>

Description

Implementation of a predicition model for functional-valued processes using kernel non parametric regression. The functional nature of curves is modeled by wavelet decomposition.

LazyLoad yes

Depends wavethresh

R topics documented:

`kerwavfun-package`	*Non parametric forecasting of functional-valued processes based on the Discrete Wavelet decomposition.*

Description

Implementation of a predicition model for functional-valued processes using kernel non parametric regression. The functional nature of curves is modeled by wavelet decomposition.

Author(s)

Jairo Cugliari <jairocugliari@gmail.com>

References

Antoniadis, A. and Paparoditis, E. and Sapatinas T. (2006) A functional wavelet-kernel approach for time series prediction. *J. R. Statis. Soc. B* **68**, pp.837–857

Antoniadis, A. and Brossat, X. and Cugliari, J. and Poggi, J.M. (2011) Clustering functional data using wavelets. *eprint* **arXiv:1101.4744**

Poggi, J.M. (1994) Prévision nonprametrique de la consommation électrique. *Revue de Statistique Appliquée* **XLII**, pp.83–98

Examples

```
library(kerwavfun)
# 1. Simulate data
delta  <- 2^6  # block size
n      <- 30   # number of blocks

simulated <- function(x, beta, theta, sd){
   beta[1] + beta[2] * (cos(2 * pi * x / 64) + sin(2 * pi * x / 64)) +
   beta[3] * (cos(2 * pi * x /  6) + sin(2 * pi * x /  6)) +
   arima.sim(list(ma=theta), n = length(x), sd = sd)
   }

mysim <- simulated(1:(n * delta), beta = c(0, 0.8, 0.18), theta = 0.8,
                   sd = sqrt(0.005))

# 2. Transform to wkdata (performs DWT)
wkdata_mysym <- wkdata(X  = mysim, gr = NULL, p  = delta,
                       colnames = 1:n, rownames = 1:delta )

# 3. Calibrate the model with the first n-1 blocks
model <- wavkerfun(obj= select.wkdata(wkdata_mysym, 1:(n-1)))

# 4. Perform the prediction of n-th block
predict_n <- predict.wavkerfun(obj= model)
```

`D1inC` *Distance on wavelets coefficients*

Description

Computes a distance between the wavelets coefficients of the wavelet transform of two vectors.

Usage

```
D1inC( x )
```

Arguments

`x`	vector of length $2^J - 1$ for some integer J.

Details

Wrapper function to code C implementation of the wavelet distance proposed on Antoniadis et al. (2006). The distance is computed from the wavelet coefficients of the vectors x, y representing the samplings of functions.

Value

Computed distance (positive real).

References

Antoniadis, A. Paparoditis, E., Sapatinas T. 2006 *J. R. Statis. Soc. B* **68**, pp.837–857

Examples

```
D1inC (x = rnorm(2^5 - 1) )
```

`dKL` *Kullback-Leibler divergence*

Description

Computes the Kullback-Leibler divergence between two probability vectors.

Usage

```
dKL(x, y, sym = TRUE)
```

Arguments

`x, y`	Nonnegative vectors.
`sym`	Logical: should the divergence be symmetrized?

Details

The function computes the Kullback-Leibler divergence. If `sym=TRUE`, the is dissimilarity symmetrized.

Value

Computed distance (positive real value).

Warning

No check is performed.

Examples

```
dKL(x = rep(0.2, 5), y = c(0.1, 0.1, 0.1, 0.1, 0.6), sym = FALSE)
```

`predict.wavkerfun` *Forecasting of wavkerfun model.*

Description

Forecasting using the wavkerfun model.

Usage

```
## S3 method for class 'wavkerfun'
predict(obj, cent = "SIMPLE")
```

Arguments

`obj`	A 'wavkerfun' object result of the `wavkerfun` function.
`cent`	String indicating the way the approximation part of the wavelet transform is treated. Options are use the same treatement as for detail part (`SIMPLE`), predict the first difference (`DIFF`) or use a persistence model (`PRST`).

Details

This function computes one step forward prediction for a `wavkerfun` model.

No `newdata` option is available as usual for predict `methods`. The prediction is made for the segment following to the last value on the object `obj`.

Value

A `wkdata` object containing the predicted values.

`select.wkdata` *Select Observations from a wkdata object.*

Description

Selects functional observation(s) from a 'wkdata' object.

Usage

```
select.wkdata(data, colnames = NULL)
```

Arguments

`data`	A 'wkdata' object.
`colnames`	A vector giving the chosen name.

Details

This function select one or several variables from 'data' and can also subset the dates.

Value

A 'wkdata' object.

`wavkerfun` *Functional wavelet-kernel model.*

Description

Estimates the parameters of a functional wavelet-kernel model.

Usage

```
wavkerfun(obj, dist = 1, r, H, EPS = 1e-06, kerneltype, plot)
```

Arguments

`obj`	A 'wkdata' object.
`dist`	Integer indicating the distance to be used. Available options are `dist=1` for `D1inC` and `dist=2` for `dKL`.
`r`	Number of observations to be used in the estimation of the bandwidth.
`H`	Vector of size 2, indicating minimum and maximum of the grid for bandwidth estimation. If only one number is given, it is used as bandwith.
`EPS`	Epsilon of tolerance for precision accurancy.
`kerneltype`	String indicating the type of kernel to be used. The available options are "uniform", "triangular", "biweight", "triweight", "epanechnikov", "gaussian", and "cauchy".

Details

The prediction is done by a kernel based estimator of the associated regression parameter. This function estimates the bandwidths value used on the kernel regression. For this, it calculates the prediction error for the last `r` observations over a grid of possible values ranging between `H[1]` and `H[2]`.

Value

An `wavkerfun` object .

See Also

See Also as `wavkerfun`.

`wkdata` *Wavelet functional data class*

Description

Object of class `wkdata` and its methods.

Usage

```
wkdata(X, p, gr, J, colnames = NULL, rownames = NULL,
               fn = 6, fmly = "DaubLeAsymm", bc = "periodic")
```

Arguments

`X`	A vector.
`p`	An integer giving the number of discretization point chosen.
`gr`	An optional vector giving groups.
`J`	An integer choosing the number of scales in the `dwt`.
`colnames`	A vector of character containing the dates of the observations.
`rownames`	A vector of character containing the discretization points.
`fn`	The filter number ranges from 1 to 8. This selects the smoothness of wavelet, see `wd`.
`fmly`	String indicating the wavelet family. Available options are "DaubExPhase" and "DaubLeAsymm". See `filter.select` for details.
`bc`	Specifies the boundary handling. Available options are 'periodic' and 'symmetric'. See `filter.select` for details.

Details

The vector `X` is arranged in a matrix of `p` rows. Each observations, i.e. the columns of the matrix, is transformed using the Discrete Wavelet Transform (DTW). The number of scales, wavelet family, filter number and boundary control are parsed to `wavethresh::wd`.

Value

An object of class 'wkdata' containing:

`X`	Original vector `X` in a matrix form.
`S0`	Wavelet transform's approximation coefficients of `X`.
`D0`	Wavelet transform's detail coefficients of `X`.
`p`	Discretization grid size.
`J`	Number of scales of the DWT.
`gr`	Optional vector indicating a grouping of observations.
`wav`	List containing `fn`, `fmly`, `bc`.

See Also

`wavethresh::wd`

References

C. Abraham, P.A. Cornillon, E. Matzner-Løber, and N. Molinari. Unsupervised curve clustering using B-splines. *Scandinavian Journal of Statistics*, 30(3):581–595, 2003.

L.F. Aguiar and M.J. Soares. Business cycle synchronization across the euro-area: a wavelet analysis. NIPE Working Papers 8/2009, NIPE - Universidade do Minho, 2009.

U. Amato, A. Antoniadis, and M. Pensky. Wavelet kernel penalized estimation for non-equispaced design regression. *Statistics and Computing*, 16(1):37–55, 2006.

J. Andersson and J. Lillestol. Modeling and forecasting electricity consuption by functional data analysis. *The Journal of Energy Markets*, 3:3–15, 2010.

G. Aneiros, R. Cao, J.M Vilar-Fernandez, and A. Muñoz San-Roque. Functional prediction for the residual demand in electricity spot markets. In F. Ferraty, editor, *Recent advances in functional data analysis and related topics*, Contributions to Statistics. Physica-Verlag Heidelberg, 2011.

J. Antoch, L. Prchal, M.R. De Rosa, and P. Sarda. Functional linear regression with functional response: application to prediction of electricity consumption. In S. Dabo-Niang and F. Ferraty, editors, *Functional and Operatorial Statistics*. Physica-Verlag Heidelberg, 2008.

A. Antoniadis. Smoothing noisy data with coiflets. *Statistica Sinica*, 4:651–678, 1994.

A. Antoniadis and T. Sapatinas. Wavelet methods for continuous-time prediction using hilbert-valued autoregressive processes. *Journal of Multivariate Analysis*, 87(1): 133–158, 2003.

A. Antoniadis, E. Paparoditis, and T. Sapatinas. A functional wavelet-kernel approach for time series prediction. *Journal-Royal Statistical Society Series B Statistical Methodoloty*, 68(5):837, 2006.

A. Antoniadis, E. Paparoditis, and T. Sapatinas. Bandwidth selection for functional time series prediction. *Statistics & Probability Letters*, 79(6):733 – 740, 2009.

M.A. Ariño, P.A. Morettin, and B. Vidakovic. On wavelet scalograms and their applications in economic time series. *Brazilian Journal of Probability and Statistics*, 18:37–51, 2004.

P. Besse and H. Cardot. Approximation spline de la prévision d'un processus fonctionnel autorégressif d'ordre 1. *Canadian Journal of Statistics*, 24(4):467–487, 1996.

P. Besse, H. Cardot, and D. Stephenson. Autoregressive forecasting of some functional climatic variations. *Scandinavian Journal of Statistics*, 27(4):673–687, 2000.

D. Bosq. Modelization, nonparametric estimation and prediction for continuous time processes. In George Roussas, editor, *Nonparametric functional estimation and related topics*, pages 509–529. NATO ASI Series, 1991.

D. Bosq. *Nonparametric statistics for stochastic processes. Estimation and prediction.* Springer-Verlag, New York, 1996.

D. Bosq. *Linear processes in function spaces: Theory and applications*. Springer-Verlag, New York, 2000.

D. Bosq and D. Blanke. *Inference and Prediction in Large Dimensions*. Wiley series in probability and statistics. John Wiley & Sons, Ltd., 2007.

A. Bruhns, G. Deurveilher, and J.S. Roy. A non linear regression model for mid-term load forecasting and improvements in seasonality. In *Proceedings of the 15th Power Systems Computation Conference*, pages 22–26, 2005.

J. Cancelo, A. Espasa, and R. Grafe. Forecasting the electricity load from one day to one week ahead for the spanish system operator. *International Journal of Forecasting*, 24:588–602, 2008.

H. Cardot. Conditional functional principal components analysis. *Scandinavian journal of statistics*, 34(2):317, 2007.

C. Chatfield. *The Analysis of Time Series*. Ed. Chapman and Hall, 1989.

A. Cohen, I. Daubechies, and P. Vial. Wavelets on the interval and fast wavelet transforms. *Applied and Computational Harmonic Analysis*, 1(1):54–81, 1993.

S. Collineau. Some remarks about the scalograms of wavelet transform coefficients. In J.S. Byrnes, J.L. Byrnes, K.A. Hargreaves, and K Berry, editors, *Wavelets and their applications*, pages 325–329. Kluwer Academic Publications, 1996.

J.A. Cuesta-Albertos and R. Fraiman. Impartial trimmed k-means for functional data. *Computational Statistics & Data Analysis*, 51(10):4864–4877, 2007.

A. Cuevas, M. Febrero, and R. Fraiman. On the use of the bootstrap for estimating functions with functional data. *Computational statistics & data analysis*, 51(2):1063–1074, 2006.

S. Dabo-Niang and N. Rhomari. Kernel regression estimation in a banach space. *Journal of Statistical Planning and Inference*, 139(4):1421–1434, 2009.

J. Damon and S. Guillas. The inclusion of exogenous variables in functional autoregressive ozone forecasting. *Environmetrics*, 13(7):759–774, 2002.

J. Damon and S. Guillas. Estimation and simulation of autoregressive hilbertian processes with exogenous variables. *Statistical Inference for Stochastic Processes*, 8(2):185–204, 2005.

J. Damon and S. Guillas. *far: Modelization for Functional AutoRegressive processes*, 2007. URL `http://cran.r-project.org`. R package version 0.6-2.

I. Daubechies. Orthonormal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics*, 41(7):909–996, 1988.

I. Daubechies. *Ten lectures on wavelets*. Society of Industrial Mathematics, 1992.

V. Dordonnat. *State-space modelling for high frequency data*. PhD thesis, Vrije Universiteit, 2009.

M. Febrero, P. Galeano, and W. González-Manteiga. Outlier detection in functional data by depth measures, with application to identify abnormal nox levels. *Environmetrics*, 19(4):331–345, 2008.

F. Ferraty and P. Vieu. *Nonparametric functional data analysis: theory and practice*. Springer-Verlag, New York, 2006.

A. Goia, C. May, and G. Fusai. Functional clustering and linear regression for peak load forecasting. *International Journal of Forecasting*, 26(4):700–711, 2010.

Y. Goude. *Mélange de prédicteurs et application à la prévision de consommation électrique*. PhD thesis, Université Paris Sud XI, 2008.

A. Grinsted, J.C. Moore, and S. Jevrejeva. Application of the cross wavelet transform and wavelet coherence to geophysical time series. *Nonlinear Processes in Geophysics*, 11(5/6):561–566, 2004.

S. Guillas. Doubly stochastic hilbertian processes. *Journal of Applied Probability*, 39(3): 566–580, 2002.

K. Gurley, T. Kijewski, and A. Kareem. First-and higher-order correlation detection using wavelet transforms. *Journal of engineering mechanics*, 129(2):188–201, 2003.

P. Hall and M. Hosseini-Nasab. On properties of functional principal components analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1): 109–126, 2006. ISSN 1369-7412.

W. Härdle. *Applied nonparametric regression*. Cambridge University Press, 1990.

T. Hastie and R. Tibshirani. Varying-coefficient models (with discussion). *Journal of the Royal Statistician Society Series B*, 55(4):757–796, 1993.

L. Horváth, M. Hušková, and P. Kokoszka. Testing the stability of the functional autoregressive process. *Journal of Multivariate Analysis*, 101(2):352–367, 2010. ISSN 0047259X.

L. Hubert and P. Arabie. Comparing partitions. *Journal of Classification*, 2(2):193–218, 1985.

S. Hörmann and P. Kokoszka. Weakly dependent functional data. *Annals of Statistics*, 38(3):1875–1884, 2010.

G.M. James and C.A. Sugar. Clustering for sparsely sampled functional data. *Journal of the American Statistical Association*, 98(462):397–409, 2003a.

G.M. James and C.A. Sugar. Finding the number of clusters in a dataset: An information-theoretic approach. *Journal of the American Statistical Association*, 98(463):750–764, 2003b.

V. Kargin and A. Onatski. Curve forecasting by functional autoregression. *Journal of Multivariate Analysis*, 99(10):2508–2526, 2008.

T. Kato. *Perturbation Theory for Linear Operators*. Springer-Verlag, Berlin, 1976.

A. Keogh and M. Pazzani. An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback. In *Proceedings of the 4th International Conference of Knowledge Discovery and Data Mining*. AAAI Press, 1998.

A. Laukatis. Functional data analysis for cash flow and transactions intensity continuous-time prediction using hilbert-valued autoregressive processes. *European Journal of Operational Research*, 185:1607–1614, 2008.

F. Leisch. A toolbox for k-centroids cluster analysis. *Computational Statistics and Data Analysis*, 51(2):526–544, 2006.

F. Leisch. Neighborhood graphs, stripes and shadow plots for cluster visualization. *Statistics and Computing*, 20(4):457–469, 2010.

Y. Luan and H. Li. Clustering of time-course gene expression data using a mixed-effects model with b-splines. *Bioinformatics*, 19(4):474—482, 2003.

S.G. Mallat. Multiresolution approximations and wavelet orthonormal bases of l2(r). *Transactions of the American Mathematical Society*, 315(1):69–87, 1989a.

S.G. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE transaction on pattern analysis and machine intelligence*, 11(7):674–693, 1989b.

S.G. Mallat. *A wavelet tour of signal processing*. Academic Press, 1999.

D. Maraun and J. Kurths. Cross wavelet analysis. significance testing and pitfalls. *Nonlinear Process on Geophysics*, 11(4):505–514, 2004.

D. Maraun, J. Kurths, and M. Holschneider. Nonstationary gaussian processes in wavelet domain: Synthesis, estimation, and significance testing. *Physical Revue E*, 75(1): 016707, 2007.

A. Mas. *Estimation d'opérateurs de corrélation de processus fonctionnels: lois limites, tests, déviations modérées*. PhD thesis, Université Paris 6, 2000.

A. Mas. Weak convergence in the functional autoregressive model. *Journal of Multivariate Analysis*, 98(6):1231–1261, 2007.

A. Mas and L. Menneteau. Perturbation approach applied to the asymptotic study of random operators. *Progress in Probability*, 55(1):127–133, 2003.

A. Mas and B. Pumo. The ARHD process. *J. of Statistical Planning and Inference*, 137 (25):538–553, 2007.

A. Mas and B. Pumo. Linear processes for functional data. In Frédéric Ferraty and Yves Romain, editors, *The Oxford Handbook of Functional Data Analysis*, Oxford Handbooks in Mathematics, chapter 3, pages 47–71. Oxford University Press, 2011. ISBN 978-0-19-956844-4.

E.A. Nadaraya. On estimating regression. *Theory of Probab. and Applic.*, 9:141–142, 1964.

G. Nason. *Wavelet methods in statistics with R*. Springer, 2008.

G. Nason. *wavethresh: Wavelets statistics and transforms.*, 2010. URL `http://CRAN.R-project.org/package=wavethresh`. R package version 4.5.

J. Nowicka-Zagrajek and R. Weron. Modeling electricity loads in california: Arma models with hyperbolic noise. *Signal Processing*, 82(12):11, 2001. URL `http://arxiv.org/abs/cond-mat/0107226`.

P.A. Nze, P. Bühlmann, and P. Doukhan. Weak dependence beyond mixing and asymptotics for nonparametric regression. *The Annals of Statistics*, 30(2):397–430, 2002.

D.B. Percival and A.T. Walden. *Wavelet methods for time series analysis*. Cambridge Univ Press, 2006.

M. Piao, H.G. Lee, J.H. Park, and K.H. Ryu. Application of classification methods for forecasting mid-term power load patterns. In *Advanced Intelligent Computing Theories and Applications. With Aspects of Contemporary Intelligent Computing Techniques*, pages 47–54. Springer, 2008.

S. Pittner and S.V. Kamarthi. Feature extraction from wavelet coefficients for pattern recognition tasks. *Pattern Analysis and Machine Learning, IEEE Transactions on*, 21 (1):83–55, 1999.

J.-M. Poggi. Prévision nonprametrique de la consommation électrique. *Rev. Statistiqué Appliquée*, XLII(4):93–98, 1994.

A.D. Poularikas. *Transforms and Applications Hanbook*. CRC Press, 2009.

B. Pumo. *Estimation et prévision de processus autorégressifs fonctionnels*. PhD thesis, University of Paris 6, 1992.

B. Pumo. Prediction of continuous time processes by c [0, 1]-valued autoregressive process. *Statistical Inference for Stochastic Processes*, 1(3):297–309, 1998.

R.Q. Quiroga, Z. Nadasdy, and Y. Ben-Shaul. Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering. *Neural computation*, 16(8):1661–1687, 2004.

J.O. Ramsay and C.J. Dalzell. Some tools for functional data analysis (with discussion). *Journal of the Royal Statistical Society. Series B*, 53(3):539–572, 1991.

J.O. Ramsay and B.W. Silverman. *Functional data analysis*. Springer-Verlag, New Yok, 1997.

J.O. Ramsay and BW Silverman. *Applied functional data analysis: methods and case studies*. Springer Verlag, 2002.

W.M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850, 1971.

T. Rouyer, J.M. Fromentin, N.C. Stenseth, and B. Cazelles. Analysing multiple time series and extending significance testing in wavelet analysis. *Marine Ecology Progress Series*, 359:11–23, 2008.

N. Serban and L. Wasserman. CATS: clustering after transformation and smoothing. *Journal of the American Statistical Association*, 100:990–999, 2004.

R. Shubhankar and Mallick B. Functional clustering by Bayesian wavelet methods. *Journal of the Royal Statistical Society. Series B*, 68(2):305–332, 2006.

L.J. Soares and M.C. Medeiros. Modeling and forecasting short-term electricity load: A comparison of methods with an application to brazilian data. *International Journal of Forecasting*, 24(4):630 – 644, 2008.

D. Steinley and M.J. Brusco. A new variable weighting and selection procedure for k-means cluster analysis. *Multivariate Behavioral Research*, 43(1):32, 2008a.

D. Steinley and M.J. Brusco. Selection of variables in cluster analysis: An empirical comparison of eight procedures. *Psychometrika*, 73(1):125–144, 2008b.

T. Tarpey. Linear transformations and the k-means clustering algorithm: Applications to clustering curves. *The American Statistician*, 61(1):34, 2007.

T. Tarpey and K.K.J. Kinateder. Clustering functional data. *Journal of Classification*, 20(1):93–114, 2003.

J.W. Taylor. Triple seasonal methods for short-term electricity demand forecasting. *European Journal of Operational Research*, 204:139–152, 2010.

R. Tibshirani, G. Walther, and T. Hastie. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 63(2):411–423, 2001.

C. Torrence and G.P. Compo. A practical guide to wavelet analysis. *Bulletin of the American Meteorological Society*, 79(1):61–78, 1998.

B. Vidakovic. *Statistical modeling by wavelets*. Wiley series in probability and mathematical statistics: Applied probability and statistics. Wiley, 1999.

H. Wang, J. Neill, and F. Miller. Nonparametric clustering of functional data. *Statistics and its interface*, 1:47–62, 2008.

G.S. Watson. Smooth regression analysis. *Sankhya Ser. A*, 25:359–372, 1964.

R. Weron. *Modeling and forecasting electricity loads and prices: a statistical approach*, volume 396 of *Wiley finance series*. John Wiley and Sons, 2006.

Y. Wu, J. Fan, and H.-G. Müller. Varying-coefficient functional linear regression. *Bernoulli*, 16(3):730–758, 2010. ISSN 1350–7265.

M. Yan and K. Ye. Determining the number of clusters using the weighted gap statistic. *Biometrics*, 63(4):1031–1037, 2007.