Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at *Energies* **2018**, *11*, 1893; doi:10.3390/en11071893

Article

Scalable Clustering of Individual Electrical Curves for Profiling and Bottom-up Forecasting

Benjamin Auder [1], Jairo Cugliari [2], Yannig Goude [3] and Jean-Michel Poggi [4]

1 LMO, Univ. Paris-Sud, Orsay, France; benjamin.auder@math.u-psud.fr
2 Univ. de Lyon, Lyon 2, ERIC EA 3083; jairo.cugliari@univ-lyon2.fr
3 EDF R&D, LMO, Univ Paris-Sud, Orsay, France; yannig.goude@edf.fr
4 Univ Paris Descartes & LMO, Univ. Paris-Sud, Orsay, France; jean-michel.poggi@math.u-psud.fr

Abstract: Smart grids require flexible data driven forecasting methods. We propose clustering tools for bottom-up short-term load forecasting. We focus on individual consumption data analysis which plays a major role for energy management and electricity load forecasting. The two first sections are dedicated to the industrial context and a review of individual electrical data analysis. We are interested in hierarchical time-series for bottom-up forecasting. The idea is to disaggregate the signal in such a way that the sum of disaggregated forecasts improves the direct prediction. The 3-steps strategy defines numerous super-consumers by curve clustering, builds a hierarchy of partitions and selects the best one minimizing a forecast criterion. Using a nonparametric model to handle forecasting, and wavelets to define various notions of similarity between load curves, this disaggregation strategy applied to French individual consumers leads to a gain of 16% in forecast accuracy. We then explore the upscaling capacity of this strategy facing massive data and implement proposals using R, the free software environment for statistical computing. The proposed solutions to make the algorithm scalable combines data storage, parallel computing and double clustering step to define the super-consumers.

Keywords: Clustering; Forecasting; Hierarchical Time-Series; Individual Electrical Consumers; Scalable; Short Term; Smart Meters; Wavelets

1. Industrial context

Energy systems are facing a revolution and many challenges. On the one hand, electricity production is moving to more intermittency and complexity with the increase of renewable energy and the development of small distributed production units such as photovoltaic panels or wind farms. On the other hand, consumption is also changing with plug-in (hybrid) electric vehicles, heat pumps, the development of new technologies such as smart phones, computers, robots that often come with batteries. To maintain the electricity quality, energy stakeholders are developing smart grids (see [1,2]), the next generation power grid including advance communication networks and associated optimisation and forecasting tools. A key component of the smart grids are smart meters. They allow two-sided communication with the customers, real time measurement of consumption and a large scope of demand side management services. A lot of countries have deployed smart meters, as stated in [3], the UK, the US and China have respectively deployed 2.9, 70 and 96 million of such equipments in 2016. In France, 35 millions will be deployed before 2021 for a global cost of 5 billions (see e.g. [4]). [5] mentions that Sweden and Italy have achieved full deployment and [6] that Italian DSOs are planning the second wave of roll-outs.

This results into new opportunities such as local optimisation of the grid, demand side management and smart control of storage devices. Exploiting the smart grid efficiently requires advanced data analytics and optimisation techniques to improve forecasting, unit commitment, and load planning at different geographical scales. Massive data sets are and will be produced as

© 2018 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at Energies 2018, 11, 1893; doi:10.3390/en11071893

explained in [7]: data from energy consumption measured by smart meters at a high frequency (every half minute instead of every 6 months); data from the grid management (e.g. Phasor Measurement Units); data from energy markets (prices and bidding, TSOs and DSOs data like balancing and capacity); data from production units and equipments for their maintenance and control (sensors, periodic measures...). A lot of efforts are made by utilities to develop datalakes and IT structures to gather and make these data available for their business units in real time. Designing new algorithms to analyse and process these data at scale is a key activity and a real competitive advantage.

We will focus on individual consumption data analysis which plays a major role for energy management and electricity load forecasting, designing marketing offers and commercial strategies, proposing new services as energy diagnostics and recommendations, detect and prevent non-technical losses.

The paper is organized as follows. After this first section introducing the industrial context, Section 2 offers a state-of-the-art review of individual electrical data analysis. Section 3 provides the big picture of our proposal for bottom-up forecasting from smart meter data, without technical details. The next three sections focus on the main tools: wavelets (Section 4) to represent functions and to define similarities between curves, the nonparametric forecasting method KWF (Section 5) and the wavelet-based clustering tools to deal with electrical load curves (Section 6). Section 7 is specifically devoted to the upscaling issue and strategy. Section 8 describes an application for forecasting a French electricity dataset. Finally, Section 9 collects some elements of discussion. It should be noted that we tried to write the paper in such a way that each section could be read independently of each other. Conversely, some sections could be skipped by some readers, without altering the local consistency of the others.

2. Individual electrical consumption data: a state-of-the-art

Individual consumption data analysis is, according to the development on smart meters, a popular and growing field of research. Composing an exhaustive survey of recent realizations is then a difficult challenge not addressed here. As detailed in [3], individual consumption data analytics covers various fields of statistics and machine learning: time series, clustering, outlier detection, deep learning, matrix completion, online learning among others.

Given a data set of individual consumptions, a first natural step is exploratory: clustering, which is the most popular unsupervised learning approach. The purpose of clustering is to partition a dataset into homogeneous subsets called clusters (see [8]). Homogeneity is measured according to various criteria such as intra and inter class variances, or distance/dissimilarity measures. The elements of a given cluster are then more similar to those of the same cluster than the elements of the other clusters. Time series clustering is an active subfield where each individual is not characterised by a set of scalar variables but are described by time series, signals or functions, considered as a whole, opening the way for signal processing techniques or functional data analysis methods (see [9] and [10] for general surveys).

Clustering methods for electricity load data have been widely applied for profiling or demand response management. [11] and [12] give an overview of the clustering techniques for customer grouping, finding patterns into electricity load data or detecting outliers and apply it to 400 non-residential medium voltage customers. Clustering can be seen as longitudinal when the objective is to cluster temporal patterns (e.g. daily load curves) from a single individual or transversal when the goal is to build clusters of customers according to their load consumption profile and/or side information. The main application of clustering is load profiling which is essential for energy management, grid management and demand response (see [13]). For example, in [14] data mining techniques are applied to extract load profiles from individual load data of a set of low voltage Portuguese customers, and then supervised classification methods are used to allocate customers to the different classes. In [15], load profiles are obtained by iterative self-organizing data analysis on metered data and demonstrated on a set of 660 hourly metered customers in Finland. [16] proposes an

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at Energies 2018, 11, 1893; doi:10.3390/en11071893

unsupervised clustering approach based on k-means on features obtained by average seasonal curves using minute metered data from 103 homes in Austin, TX. Correspondence between clusters, their associated profiles and survey data are also studied. Authors of [17] suggest a k-means clustering to derive daily profiles from 220,000 homes and a total of 66 millions daily curves in California. Other approaches based on mixture models are presented in [18] for customers categorization and load profiling on a data set of 2,613 smart metered household from London.

Another interest of clustering is forecasting, more precisely bottom-up forecasting which means forecasting the total consumption of a set of customers using individual metered data. Forecasting is an obvious need for optimisation of the grid. As pointed previously, it becomes more and more challenging but also crucial to forecast electricity consumption at different "spatial" scale (a district, a city, a region but also a segment of customers). Bottom up methods are a natural approach consisting in building clusters, forecasting models in each cluster and then aggregating them. In [19], clustering algorithms are compared according to their forecasting accuracy on a data set consisting of 6000 residential customers and SME in Ireland. Good performances are reached but the proposed clustering methods are defined quite independently of the model used for forecasting. On the same data set, [20] associate a longitudinal clustering and a functional forecasting model similar to KWF (see [21]) for forecasting individual load curves.

A clustering method supervised by forecasting accuracy is proposed in [22] to improve the forecast of the total consumption of a French industrial subset obtaining a substantial gain but suffering from high computational time. In [23], a k-means procedure is applied on features consisting in mean consumption for 5 well chosen periods of day, mean consumption per day of a week and peak position into the year. In each cluster a deep learning algorithm is used for forecasting and then the bottom up forecast is the simple sum of clusters forecasts. Results showing a minimum gain of 11% in forecast accuracy are provided on the Irish data set and smart meter data from New-York. On the Irish data again, [3] propose to build an ensemble of forecasts from a hierarchical clustering on individual average weekly profiles, coupled with a deep learning model for forecasting in each cluster. Different forecasts corresponding to different sizes of the partition are at the end aggregated using linear regression.

We propose here a new approach, following the previous work of [24], to build clusters and forecasting models that are performant for the bottom-up forecasting problem as well as from the computational point of view.

3. Bottom-up forecasting from smart meter data: big picture

We present in this section the whole procedure to obtain a hierarchy of partitions of customers, schematically represented in Figure 1. On the bottom line, there are N individual customers, say $I_1, \ldots, I_N$. Each of them has an individual demand coded into an electrical load curve. At the top of the schema, there is one single global demand G obtained by the simple aggregation of the individual ones at each time step, i.e. $G = \sum_n I_n$. We look for the construction of a set of K medium level aggregates, $A_1, \ldots, A_K$ such that they form a partition of the individuals. Each of the considered entities (individuals, medium level aggregates and global demand) can be considered as time series since they carry important time dependent information.

In the context of economic seasonal univariate continuous time series, it is often natural to segment it in time, into consecutive curves, for example days, which are then treated as a discrete time series of functions. In particular, in the electrical context, the shape of the curves exhibits rich information about the calendar day type, the meteorological conditions or the existence of special electricity tariffs. Using the information contained in the shape of the load curves leads to very elegant formulation of functional forecasting.

The shape of the curves exhibits rich information about the calendar day type, the meteorological conditions or the existence of special electricity tariffs. Using the information contained in the shape of the load curves, [21] proposed a flexible nonparametric function-valued forecast model called KWF

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at Energies 2018, 11, 1893; doi:10.3390/en11071893

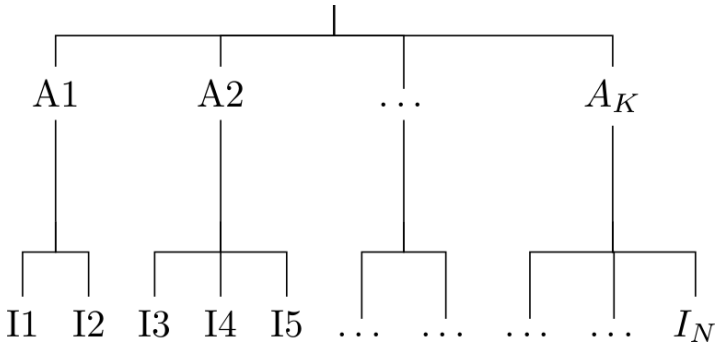

Figure 1. Schematic representation of a hierarchy of customers.

(*Kernel + Wavelet + Functional*) well suited to handle nonstationary series. The predictor can be seen as a weighted average of futures of past situations, where the weights increase with the similarity between the past situations and the actual one.

Pattern research-based methods repose on a fully non parametric and thus more general frame than prediction approaches more adapted to electricity load demand. This point can be seen as both a weakness and a strength. Specific models can better express the known dependences of electricity demand to long-term trend, seasonal components (due to the interaction of economic and social activities) and climate. However, they usually need more human time to be calibrated. The arrival of new measurement technologies structure of intelligent networks, with more local and high resolution information, unveils the need for local electricity load forecasting at different levels of the grid.

Bottom-up approaches, based on a two stage process combining clustering and forecasting methods, are a promising perspective. First, it consists in building classes in a population such that each class could be sufficiently well forecast but corresponds to different load shapes or reacts differently to exogenous variables like temperature or prices (see e.g. [25] in the context of demand response). The second stage consists in aggregating predictions to forecast the total or any subtotal of the population consumption. For example, identify and forecast the consumption of a sub-population reactive to an incentive is an important need to optimize a demand response program.

Recently, [24] proposed to build clustering tools useful for the two tasks simultaneously: clustering individual customers and forecasting the load consumption. The idea is to disaggregate the global signal in such a way that the sum of disaggregated forecasts significantly improves the prediction of the whole global signal. The general strategy is in three steps: first we cluster individual curves defining super-consumers, then we build a hierarchy of partitions within which a best one is finally selected with respect to a disaggregated forecast criterion. The predictions are made with the KWF model which allows one to use it as a off-the-shelve tool.

In concrete, data for each customer is a set of P time dependent (potentially noisy) records evenly sampled at a relatively high frequency (e.g. $1/4$, $1/2$ or hourly records). Then, we consider the data for each individual as a time series that we treat as a function of time. Wavelets are used to code the information about the shape of the curves. Thanks to nice mathematical properties of wavelets, we compress the information of each curve into a handy number of coefficients (in total $J = [\log_2(P)]$) that are called relative energetic contributions. The compression is such that discriminative power if kept even if information is lost. This allows us to fall fall into the classical framework of data analysis where data is tabulated into a matrix with lines containing observations and columns containing variables (see Figure 2).

The following sections are devoted to the technical details of the different modules that compose our framework. Wavelets are rapidly reviewed in Section 4, the prediction model KWF is presented in Section 5 and the clustering strategy is detailed in Section 6. In order to upscale the procedure to cope with tens of millions of load curves two strategies are discussed in Section 7, along with some

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at Energies 2018, 11, 1893; doi:10.3390/en11071893

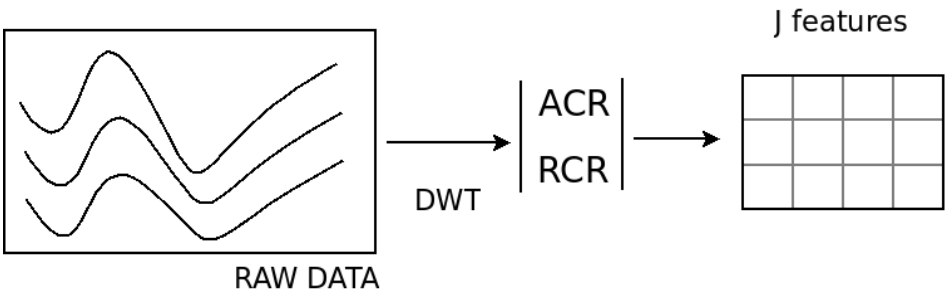

Figure 2. From curves to matrices.

computational considerations. Finally, 8 presents numerical experiments and results using a French electricity dataset.

4. Wavelets

A wavelet ψ is a sufficiently regular and well localized function verifying a simple admissibility condition. During a certain time a wavelet oscillates like a wave and is then localized in time due to a damping. Figure 3 represents the Daubechies least-asymmetric wavelet of order 6. From this single function ψ, using translation and dilation a family of functions that form the basic atoms of the Continuous Wavelet Transform (CWT) is derived

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right), a \in \mathbf{R}_*^+, b \in \mathbf{R}.$$

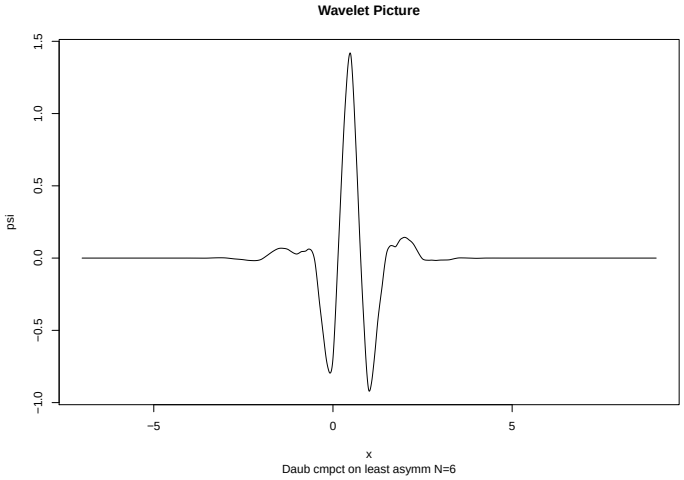

Figure 3. Daubechies least-asymmetric wavelet with filter size 6.

For a function $z(t)$ of finite energy we define its CWT by the function C_z of two real-valued variables:

$$C_z(a,b) = \int_{-\infty}^{\infty} z(t)\psi_{a,b}(t)\,\mathrm{d}t$$

Each single coefficient measures the fluctuations of function f at scale a, around the position b. Figure 4 gives a visual representation of $|C_z(a,b)|^2$, also known as wavelet spectrum, for a 10 days period of load demand sampled at 30 minutes. The waves that one can visually find on the image indicate the highest zone of fluctuations which corresponds to the days. CWT is then extremely redundant but it is useful for example, to characterize the Holderian regularity of functions

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at Energies 2018, 11, 1893; doi:10.3390/en11071893

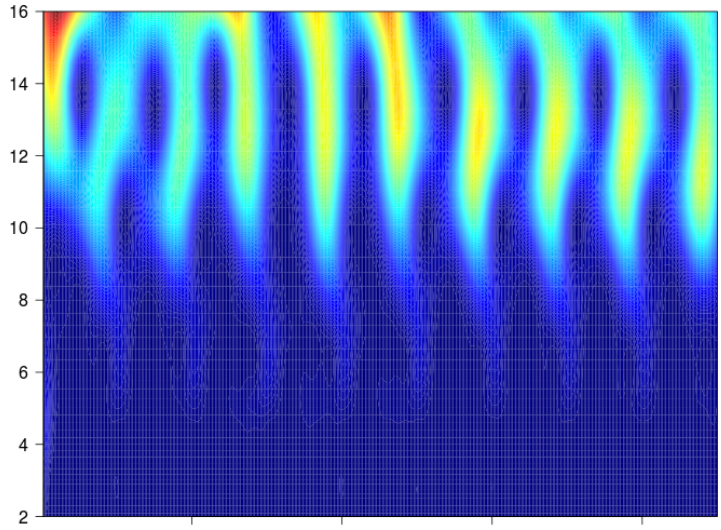

Figure 4. Wavelet spectrum of a week of electrical load demand

or to detect transient phenomena or change-points. A more compact wavelet transform can also be defined.

The Discrete Wavelet Transform is a technique of hierarchical decomposition of the finite energy signals. It allows to represent a signal in the time-scale domain, where the scale plays a role analogous to that of the frequency in the Fourier analysis ([26]). It allows to describe a real-valued function through two objects: an approximation of this function and a set of details. The approximation part summarizes the global trend of the function, while the localized changes (in time and frequency) are captured in the detail components at different resolutions. The analysis of signals is carried out by analysing functions called wavelets obtained from simple transformations of a single function called mother wavelet. 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 an orthonormal basis of waveforms derived from scaling (i.e. dilating or compressing) 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

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

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

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

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

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

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at Energies 2018, 11, 1893; doi:10.3390/en11071893

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

$$c_{j,k} =< z, \phi_{j,k} >_{\mathcal{H}} \qquad d_{j,k} =< z, \psi_{j,k} >_{\mathcal{H}} .$$

To efficiently calculate the WT, Mallat introduced the notion of mutiresolution analysis of $\mathcal{H}$ (MRA) and designed a family of fast algorithms (see [26]). With MRA, the first term at the right hand side of (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 $j \geq j_0$. We will focus our attention on the finer details, i.e. on the information at the scales $\{j : j \geq j_0\}$.

Figure 5 is the multiresolution analysis of a daily load curve. The original curve is represented on the top leftmost panel. The bottom rightmost panel contains the approximation part at the coarsest scale $j_0 = 0$, that is, a constant level function. The set of details are plotted by scale which can be connected to frequencies. With this, the detail functions clearly show the different patterns ranging between low and high frequencies. The structure of the signal is centred on the highest scales (lowest frequencies), while the lowest scale (highest frequencies) keep the noise of the signal.

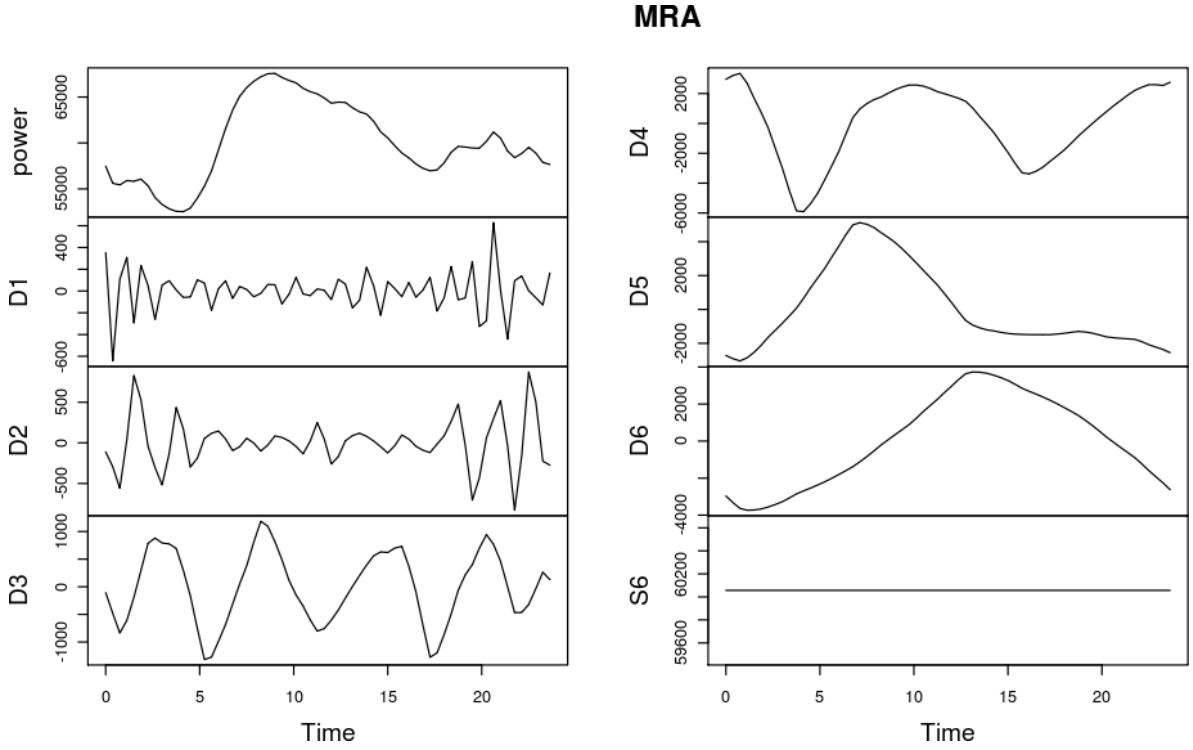

Figure 5. Multiresolution analysis of a daily load curve.

From a practical point of view, let us suppose for simplicity that each function is observed on a fine time sampling grid of size $N = 2^J$ (if not, one may interpolate data to the next power of two). In this context we use a highly efficient pyramidal algorithm ([27]) to obtain the coefficients of the Discrete Wavelet Transform (DWT). Denote by $\mathbf{z} = \{z(t_l) : l = 0, \ldots, N_i - 1\}$ 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 (1) using the truncation imposed by the 2^J points and the coarser approximation level $j_0 = 0$, as:

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

Hence, for a given wavelet ψ and a coarse resolution $j_0 = 0$, one may define the DWT operator:

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

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at Energies 2018, 11, 1893; doi:10.3390/en11071893

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, the energy of a square integrable signal is preserved:

$$\|\mathbf{z}\|_2^2 = c_0^2 + \sum_{j=0}^{J-1} \sum_{k=0}^{2^j-1} d_{j,k}^2 = c_0^2 + \sum_{j=0}^{J-1} \|\mathbf{d}_j\|_2^2. \tag{3}$$

Hence, the global energy $\|\mathbf{z}\|_2^2$ of $\mathbf{z}$ is distributed over some energetic components. The way these energies 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.

5. KWF

5.1. From Discrete to Functional Time Series

Theoretical developments and practical applications associated with functional data analysis were mainly guided by the case of independent observations. However, there is a wide range of applications in which this hypothesis is not reasonable. In particular, when we consider records on a finer grid of time assuming that the measures come from a sampling of an underlying unknown continuous-time signal.

Formally, the problem can be written by considering a continuous stochastic process $X = (X(t), t \in \mathbb{R})$. So the information contained in a trajectory of X observed on the interval $[0, T], T > 0$ is also represented by a discrete-time process $Z = (Z_k(t), k = 0, \ldots, n; t \in [0, \delta])$ where $Z_k(t) = X((\delta - 1)k + t)$ comes from the segmentation of the trajectory X in n blocks of size $\delta = T/n$ ([28]). Then, the process Z is a time series of functions. For example, we can forecast $Z_{n+1}(t)$ from the data $Z_1, \ldots, Z_n$. This is equivalent to predicting the future behaviour of the X process over the entire interval $[T, T + \delta]$ by having observed X on $[0, T]$. Note that by construction, the $Z_1, \ldots, Z_n$ are usually dependent functional random variables.

This framework is of particular interest in the study of electricity consumption. Indeed, the discrete consumption measurements can naturally be considered as a sampling of the load curve of an electrical system. The usual segment size, $\delta = 1$ day, takes into account the daily cycle of consumption.

In [21], the authors proposed a prediction model for functional time series in the presence of non stationary patterns. This model has been applied to the electricity demand of *Electricité de France* (EDF). The general principle of the forecasting model is to find in the past, situations similar to the present and linearly combine their futures to build the forecast. The concept of similarity is based on wavelets and several strategies are implemented to take into account the various non stationary sources. [29] proposes for the same problem to use a predictor of a similar nature but applied to a multivariate process. Next, [30] provide an appropriate framework for stationary functional processes using the wavelet transform. The latter model is adapted and extended to the case of non-stationary functional processes ([31]).

Thus, a forecast quality of the same order of magnitude as other models used by EDF is obtained for the national curve (highly aggregated) even though our model can represent the series in a simple and parsimonious way. This avoids explicitly modeling the link between consumption and weather covariates, which are known to be important in modeling and often considered essential to take into account. Another advantage of the functional model is its ability to provide multi-horizon forecasts simultaneously by relying on a whole portion of the trajectory of the recent past, rather than on certain points as univariate models do.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at Energies 2018, 11, 1893; doi:10.3390/en11071893

5.2. Functional model KWF

5.2.1. Stationary case

We consider a stochastic process $Z = (Z_i)_{i\in\mathbb{Z}}$ assumed for the moment, to be stationary, with values in a functional space H (for example $H = L_2([0,1])$). We have a sample of n curves $Z_1, \ldots, Z_n$ and the goal is to forecast Z_{n+1}. The forecasting method is divided in two steps. First, find among the blocks of the past those that are most similar to the last observed block. Then build a weight vector $w_{n,i}, i = 1, \ldots, n-1$ and obtain the desired forecast by averaging the future blocks corresponding to the indices $2, \ldots, n$ respectively.

First step.

To take into account in the dissimilarity the infinite dimension of the objects to be compared, the KWF model represents each segment $Z_i, i = 1, \ldots, n$, by its development on a wavelet basis truncated to a scale $J > j_0$. Thus, each observation Z_i is described by a truncated version of its development obtained by the discrete wavelet transform (DWT):

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

The first term of the equation is a smooth approximation to the resolution j_0 of the global behaviour of the trajectory. It contains non-stationary components associated with low frequencies or a trend. The second term contains the information of the local structure of the function. For two observed segments $Z_i(t)$ and $Z_{i'}(t)$, we use the dissimilarity D defined as follows:

$$D(Z_i, Z_{i'}) = \sum_{j=j_0+1}^{J} 2^{-j} \sum_{k=0}^{2^j-1} (d_{j,k}^{(i)} - d_{j,k}^{(i')})^2. \tag{4}$$

Since the Z process is assumed to be stationary here, the approximation coefficients do not contain useful information for the forecast since they provide local averages. As a result, they are not taken into account in the proposed distance. In other words, the dissimilarity D makes it possible to find similar patterns between curves even if they have different approximations.

Second step.

Denote $\Xi_i = \{c_{J,k}^{(i)} : k = 0, 1, \ldots, 2^J - 1\}$ the set of scaling coefficients of the i-th segment Z_i at the finer resolution J. The prediction of scaling coefficients (at the scale J) $\widehat{\Xi_{n+1}}$ of Z_{n+1} is given by:

$$\widehat{\Xi_{n+1}} = \frac{\sum_{m=1}^{n-1} K_{h_n}(D(Z_{n,J}, Z_{m,J}))\Xi_{m+1}}{1/n + \sum_{m=1}^{n-1} K_{h_n}(D(Z_{n,J}, Z_{m,J}))},$$

where K is a probability kernel. Finally, we can apply the inverse transform of the DWT to $\widehat{\Xi_{n+1}}$ to obtain the forecast of the Z_{n+1} curve in the time domain. If we note

$$w_{n,m} = \frac{K_{h_n}(D(Z_{n,J}, Z_{m,J}))}{\sum_{m=1}^{n-1} K_{h_n}(D(Z_{n,J}, Z_{m,J}))}, \tag{5}$$

these weights allow to rewrite the predictor as a barycentre of future segments of the past:

$$\widehat{Z_{n+1}}(t) = \sum_{m=1}^{n-1} w_{n,m} Z_{m+1}(t). \tag{6}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at Energies 2018, 11, 1893; doi:10.3390/en11071893

5.2.2. Beyond the stationary case

In the case where Z is not a stationary functional process, some adaptations in the predictor (6) must be made to account for nonstationarity. In Antoniadis *et al.* (2012) corrections are proposed and their efficiency is studied for two types of non-stationarities: the presence of an evolution of the mean level of the approximations of the series and the existence of classes segments. Let us now be more precise.

It is convenient to express each curve Z_i according to two terms $\mathcal{S}_i(t)$ and $\mathcal{D}_i(t)$ describing respectively the approximation and the sum of the details,

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

When the curves Z_{m+1} have very different average levels, the first problem appears. In this case, it is useful to centre the curves before calculating the (centred) prediction, and then update the forecast in the second phase. Then, the forecast for the segment $n+1$ is $\widehat{Z_{n+1}(t)} = \widehat{\mathcal{S}_{n+1}(t)} + \widehat{\mathcal{D}_{n+1}(t)}$. Since the functional process $\mathcal{D}_{n+1}(t)$ is centred, we can use the basic method to obtain its prediction

$$\widehat{\mathcal{D}_{n+1}(t)} = \sum_{m=1}^{n-1} w_{m,n} \mathcal{D}_{n+1}(t), \tag{7}$$

where the weights $w_{m,n}$ are given by (5). Then, to forecast $\mathcal{S}_{n+1}(t)$ we use

$$\widehat{\mathcal{S}_{n+1}(t)} = \mathcal{S}_n(t) + \sum_{m=1}^{n-1} w_{m,n} \Delta(\mathcal{S}_n)(t). \tag{8}$$

To solve the second problem, we incorporate the information of the groups in the prediction stage by redefining the weights $w_{m,n}$ according to the belonging of the functions m and n to the same group:

$$\widetilde{w}_{m,n} = \frac{w_{m,n} \mathbf{1}_{\{gr(m)=gr(n)\}}}{\sum_{m=1}^{n} w_{m,n} \mathbf{1}_{\{gr(m)=gr(n)\}}}, \tag{9}$$

where $\mathbf{1}_{\{gr(m)=gr(n)\}}$ is equal to 1 if the groups $gr(n)$ of the n-th segment is equal to the group of the m-th segment and zero elsewhere. If the groups are unknown, they can be determined from an unsupervised classification method.

The weight vector can give an interesting insight on the prediction power carried on by the shape of the curves. Figure 6 represents the computed weights obtained for the prediction of a day during Spring 2007. When plotted against time, it is clear that the only days found similar to the current one are located in a remarkably narrow position of each year in the past. Moreover, the weights seem to decrease with time giving more relevance to those days closer to the prediction past. A closer look at the weight vector (not shown here) reveals that only days in Spring are used. Note that no information about the position of the year was used to compute the weights. Only the information coded in the shape of the curve is necessary to locate the load curve at its effective position inside the year.

Figure 7 is also of interest to understand how the prediction works. There, the plot on the left contains all the days of the dataset against which the similarity was computed with respect to the curve in blue. A transparency scale which makes visible only those curves with a relatively high similarity index. The plot on the right contains the futures of the past days on the left. These are also plot on the transparent scale with the curve in orange which is the prediction given by the weighted average.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at Energies 2018, 11, 1893; doi:10.3390/en11071893

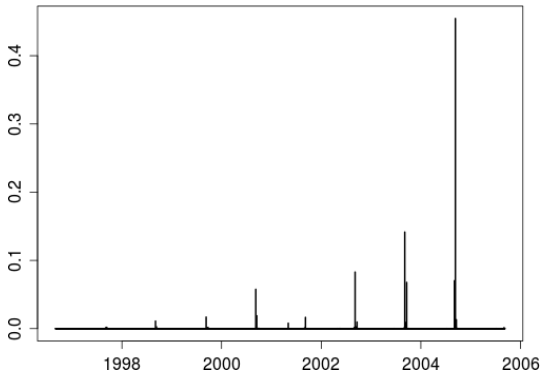

Figure 6. Vector of weights (sorted chronologically) obtained for the prediction of a day during Spring. On each panel, all the days are represented with a transparent colour making visible only the most relevant days for the construction of the predictor.

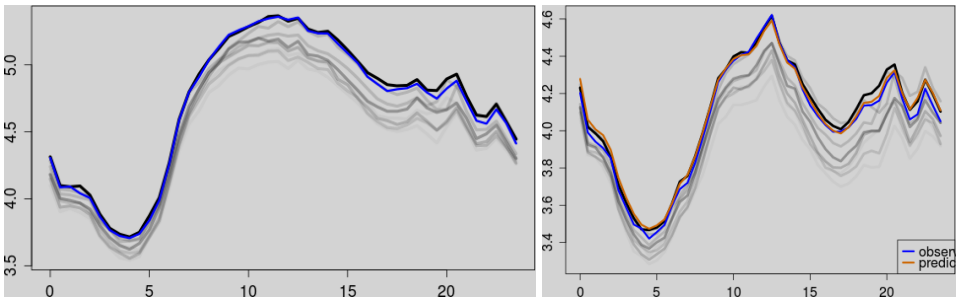

Figure 7. Past and future segments involved in the construction of the prediction by KWF.

6. Clustering electrical load curves

The lack of regularity of the individual signals make the prediction of each customer a very difficult task. The variability of each individual demand is such that the ratio signal to noise decreases dramatically when passing from aggregate to individual data. With almost no hope of predicting individual data, an alternative strategy is to use these data to improve the prediction of the aggregate signal. For this, one may rely on clustering strategies where customers of similar consumption structure will be put into classes in order to form groups of heterogeneous clients. If the clients are similar enough, the signal of the aggregate will gain in regularity and thus in predictability.

Many clustering methods exist in the specialized literature. We adopt the point of view of [32] were two strategies for clustering functional data using wavelets are presented. While the first one allows to rapidly create groups using a dimension reduction approach, the second one permits to better exploit the time-frequency information at the price of some computational burden.

Clustering by feature extraction

From Equation (3), we can see that the global energy of the curve is approximately decomposed into energy components associated with the smooth approximation of the curve (c_0^2) plus a set of components related to each detail level. In [32] these detail levels were called the absolute contributions $\mathrm{AC}_j = \|d_j\|_2^2, j = 0, \ldots, J-1$ of each scale to the global energy of the curve. Notice that the approximation part is not of primary interest since the underlying process of electrical demand may be highly non stationary. With this, we focus only on the shape of the curves and on its frequency content in order to unveil the structure of similar individual consumers to construct clusters. A normalized version of absolute contributions can be considered, which is called relative contributions and is defined as $\mathrm{RC}_j = \mathrm{AC}_j / \sum_j \mathrm{AC}_j$. After this representation step, the result is depicted by the

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at Energies 2018, 11, 1893; doi:10.3390/en11071893

schema in Figure 2, where the original curves are now embedded into a multi dimensional space of dimension J. Moreover if relative contributions are used, the points are in the simplex of $\mathbb{R}^{\mathbb{J}}$.

Now we begin with the proper clustering step. For this any clustering algorithm on multivariate data can be used. Since the time complexity of this step depends on the number of observation N and of variables P we decide to screen out irrelevant features using a feature selection algorithm for unsupervised learning described in [33]. Besides the reduction of the computation time, feature selection allows also to gain in interpretability of the clustering since it highly depends on the data.

The aim of this first clustering step is to produce first a coarse clustering with a sufficiently large number K' of prototypical customers that we call super-customers (SC). For each SC we can compute the synchrone demand, that is the direct sum of the individual demand of customers that belongs to the group at each time step of the sampling grid. Notice the parallelism with the initial situation, we have now K' coarsely aggregated demands over P records that can be seen as a discretized noisy sampling of a curve. Figure 8 shows this first clustering round on the first row of the schema.

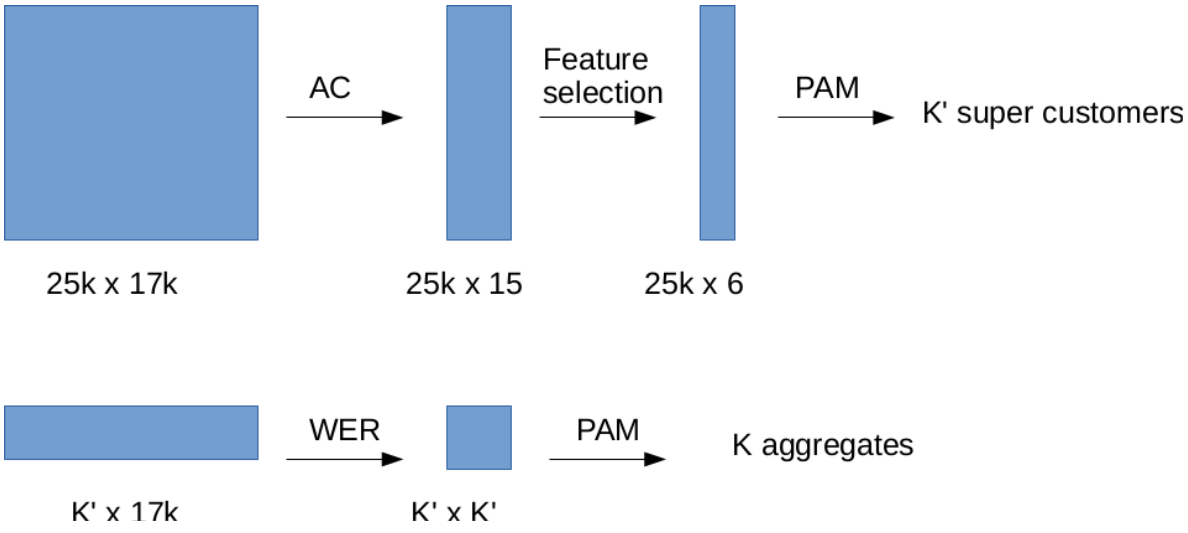

Figure 8. Two step clustering.

Clustering using a dissimilarity measure

The aim of the second stage of the clustering is to aggregate the SC into a small number K of aggregates and to construct a hierarchy. Instead of projecting curves and work with coefficients, we choose to use a notion of dissimilarity between objects of functional nature to construct a dissimilarity matrix between the SC.

Following [32] we use the wavelet extended R^2 based distance (WER) which is constructed on top of the wavelet coherence. If $x(t)$ and $z(t)$ are two signals, the the wavelet coherence between them is defined as

$$R_{x,z}(a,b) = \frac{|S(C_{x,z}(a,b))|}{|S(C_{x,x}(a,b))|^{1/2}|S(C_{z,z}(a,b))|^{1/2}},$$

where $C_{x,z}(a,b)) = C_x(a,b))C_z^*(a,b))$ is the cross-wavelet transform, and S is a smooth operator. Then, the wavelet coherence can be seen as a linear correlation coefficient computed in the wavelet domain and so localized both in time and scale. Notice that smoothing is a mandatory step in order to avoid a trivial constant $R_{x,z}(a,b) = 1$ for all a, b.

The wavelet coherence is then a two dimensional map that quantifies for each time-scale location the strength of the association between the two signals. In order to produce a single measure of this map, some kind of aggregation must be done. Following the construction of the extended

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at Energies 2018, 11, 1893; doi:10.3390/en11071893

determination coefficient R^2, [32] propose to use the wavelet extended R^2 which can be computed using

$$WER^2_{x,z} = \frac{\sum_{j=1}^{J}\left(\sum_{k=1}^{N}|S(C_{x,z}(j,k))|\right)^2}{\sum_{j=1}^{J}\left(\sum_{k=1}^{N}|S(C_{x,x}(j,k))|\sum_{k=1}^{N}|S(C_{z,z}(j,k))|\right)}.$$

Notice that $\mathrm{WER}^2_{x,z}$ is a similarity measure and it can easily be transformed into a dissimilarity one by

$$D(x,z) = \sqrt{JN(1 - WER^2_{x,z})},$$

where the computations are done over the grids $\{1,\dots,N\}$ for the locations b and $\{a_j, j = 1,\dots J\}$ for the scales a. 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 usually to a linear interpolation on a base 2 logarithmic scale.

While the measure is justified by the power of the wavelet analysis, in practice this distance implies heavy computations involving complex numbers and so requires of a larger memory space. This is one of the two reason that renders its use on the original dataset intractable. The second reason is related to the size of the dissimilarity matrix that results from its applications and that grows with the square of the number of time series. Indeed, such a matrix obtained from the SC is largely tractable for a moderate number of super customers of about some hundreds, but it is not if applied on the whole dataset of some tens of millions of individual customers. The trade off between computation time and precision is resolved by a first clustering step that dramatically reduces the number of time series using the RC features; and a second step that introduces the finer but computationally heavier dissimilarity measure on the SC aggregates.

Since the number of SC, K', is small enough, we can construct a dissimilarity matrix between the SC which is the input of the classical Agglomerative Hierarchical Clustering (AHC) used here with the Ward link. The output of the AHC is the desired hierarchy of (super-)customers. Otherwise, one may use other clustering algorithms that use a dissimilarity matrix as input (for instance Partitioning Around Mediods, PAM) to get an optimal partitioning for a fixed number of clusters. The second row of the scheme in Figure 8 represents this second step clustering.

7. Upscaling

We discuss in this section the ideas we develop to upscale the problem. Our final target is to work over twenty million time-series. For this, we run many independent clustering tasks in parallel, before merging the results to obtain an approximation of the direct clustering. We give proposed solutions that were tested in order to improve the code performance. Some of our ideas proved to be useful for moderate sample sizes (say tens of thousands) but turned to be counter-productive for larger sizes (tens of millions). Of course all these considerations depend heavily on the specific material and technology. We recall that our interest is on relatively standard scientific workstations. The algorithm we use on the first step of the clustering is described below. We then show the results of the profiling of our whole strategy to highlight where are the bottlenecks when one wishes to upscale the method. We end this section discussing the solutions we proposed.

7.1. Algorithm description

The massive dataset clustering algorithm is as follows:

0. *Data serialization.* Time series are given in a verbose by-column format. We re-code all of them in a binary file (if suitable), or a database.
1. *Dimensionality reduction.* Each series of length N is replaced by the $\log_2(N)$ energetic coefficients defined using a wavelet basis. Eventually a feature selection step can be performed to further reduction on the number of features.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at Energies 2018, 11, 1893; doi:10.3390/en11071893

2. *Chunking.* Data is chunked into groups of size at most n_c, where n_c is a user parameter (we use $n_c = 5000$ in the next section experiments).
3. *Clustering.* Within each group, the PAM clustering algorithm is run to obtain K_0 clusters.
4. *Gathering.* A final run of PAM is performed to obtain K' mediods, $K' \ll n$ out of the $n_c \times K_0$ mediods obtained on the chunks..

From these K' medoids the synchrone curves are computed (sum of all curves within each group), and given on output for the prediction step.

7.2. Code profiling

Figure 9 gives some timings obtained by profiling the runs of our initial (C) code. In order to give a clearer insight, we also report the size of the objects we deal with. The starting point is the ensemble of individual records of electricity demand for a whole year. Here, we treat over 25000 clients sampled half-hourly during a year. The tabulation of these data to obtain a matrix representation suitable to fit in memory take about 7 minutes and requires over 30 Gb of memory.

Task	Time	Memory	Disk
Raw (15Gb) to matrix	7 min	30 Gb	2.7 Gb
Compute contributions	7 min	<1Gb	7 Mb
1st stage clustering	3 min	<1Gb	–
Aggregation	1 min	6Gb	30 Mb
Wer distance matrix	40 min	64Gb	150 Kb
Forecasts	10 min	<1Gb	–

Figure 9. Code profiling by tasks.

7.3. Proposed solutions

Two main solutions are to be discussed, concerning the internal data storage strategy and the use of a simple parallelization scheme. The former looks for reducing the communication time of internal operations using serialization. The latter attacks the major bottleneck of our clustering approach, that is the construction of the WER dissimilarity matrix.

The initial format (verbose, by-column) is clearly inappropriate for efficient data processing. There are several options starting from this data format, they imply having all series stored as

- an ASCII file, one sample per line; very fast, but data retrieval will depend on line number;
- a binary format (3 or 4 octets per value); compression is unadvised since it would increase both preprocessing time and (by a large amount) reading times;
- a database (this is the slowest option), so that retrieval can be very quick.

Since we plan to deal with millions of series of thousands time steps, binary files seemed like a good compromise because they can easily fit on disk − and often also in memory. Our R package uses this format internally, although it allows to input data in any of these three shapes. If we were speaking of billions of series of a million time steps or more, then distributed databases would be required. In this case one would only has to fill the database and tell the R package how to access time-series.

The current version is mostly written in `R` using the `parallel` package for efficiency, [34]. A partial version written fully in `C` was slightly faster, but not enough compared to the loss of code clarity. The current `R` version can theoretically process the 20 million samples on a standard desktop computer in 24 hours or less – assuming the curves can be stored and accessed quickly.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at Energies 2018, 11, 1893; doi:10.3390/en11071893

Table 1. Mean average running times (over 5 replicates) for different sample sizes (in log).

Sample size	Time (in seconds)
$25x10^3$	67
$25x10^4$	513
$25x10^5$	4420
$25x10^6$	43893

8. Forecasting French electricity dataset

8.1. Data presentation

We work on the data provided by EDF also used in [24] which is composed of big customers equipped with smart meters. The dataset consists in approximately 25000 half-hourly load consumption series over two years (2009-2010). The first year is used for partitioning and the calibration of our forecasting algorithm, then the second year is used as a test set to simulate a real forecasting use-case.

The initial dataset contains over 25,000 individual load curves. In order to test the upscaling ability of our implementation, we create three datasets of sizes 250,000; 2,500,000 and 25,000,000. In other words, we progressively increase the sample sizes by a factor of 10, 100 and 1000 respectively. The creation follows a simple scheme where each individual curve is multiplied by the realization of independent variables uniformly distributed on $[0.95, 1.05]$ at each time step. Each curve is then replicated using this scheme by a number of times equal to the upscaling factor.

8.2. Numerical experiments

The first task clustering is crucial for reducing the dimension of the dataset. We give some timings in order to illustrate how our approach can deal with tens of thousands of time series. Of course, the total computation time depends on the technical specification of the structure used to perform the computation. In our case, we restrict ourselves to a standard scientific workstation with 8 physical cores and 70 Gigabits of live memory. We use all the available cores to cluster chunks of 5000 observations following the algorithm described in Section 7 for both the first and second clustering task.

A very simple pretreatment is done in order to eliminate load curves with eventual errors. For this, we measure the standard deviation of the contributions of each curve to keep only the 99% central observations eliminating the extremest ones. With this, too flat curves (maybe constant) consumptions or very wiggle ones are considered to be abnormal.

Table 1 gives mean average running times over 5 replicates for each of the different sample sizes. These figures show that our strategy yields on a linear increment on the computation time with respect to the number of time series. The maximum number of series we treat, that is 25 millions of individual curves, needs about 12 hours to achieve the first task clustering.

The result of this first clustering task is the load curves of 200 super consumers (SC). We now explore how much time series contains the super consumers. For this, we plot (in Figure 10) the relative frequency of each SC cluster (i.e. proportion of observation in the cluster) against its size rank (in logarithmic scale). With this, the leftmost point of the curve represents the largest cluster, while the following ones are sorted in decreasing order of size. To compare between sample sizes four curves are plotted, one for each sample size. A common decreasing trend of the curves appears producing several relatively small clusters. This is not a desired behaviour for a final clustering task. However, we are in an intermediate step which aims at reducing the number of curves n to a certain number K' of super customers, here $K' = 200$. The isolated super customers may merge together in the following step, producing meaningful aggregates.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at Energies 2018, 11, 1893; doi:10.3390/en11071893

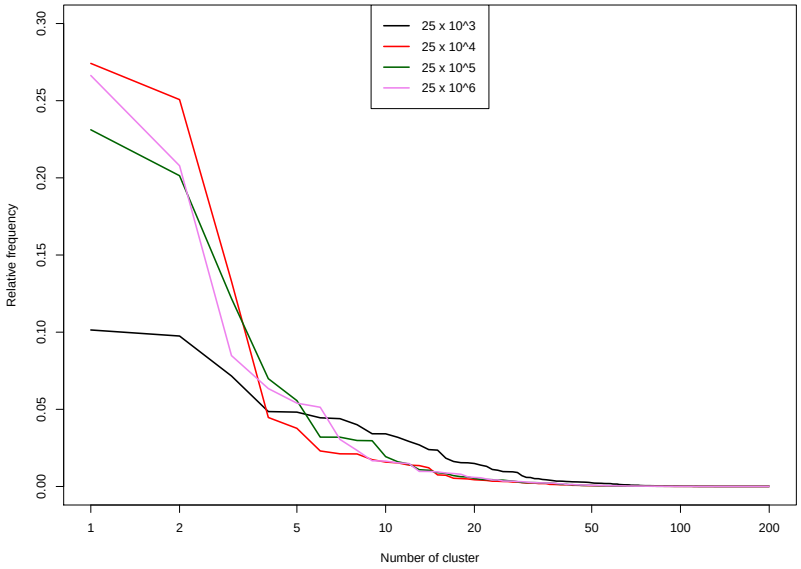

Figure 10. Relative frequency of observation by clusters, in decreasing order, for different sample sizes.

In what follows we focus on the results for the largest dataset, that is the one with over 25 millions of load curves. The resulting 200 super consumers are used to construct the WER dissimilarity matrix, which contains rich information about the clustering structure. One may use for instance a hierarchical clustering algorithm to obtain a hierarchy of SC. A graphical result of this structure in the object of Figure 11, which corresponds to the dendrogram obtained by agglomerative hierarchical clustering using the Ward link function. Then, one may get a partitioning of the ensemble of SC by setting some threshold (a value of height in the figure). However, we will not follow this idea to concentrate on the bottom-up prediction task.

The WER dissimilarity matrix encodes rich information about the pairwise closeness between the 200 super consumers. A way to visualize this matrix is to obtain a multidimensional scaling, that is to construct a setting of low dimension coordinates that best represent the dissimilarities between the curves. Figure 12 contains the matrix scatter plot of the first 4 dimensions of such a setting. For each bivariate scatter plot, the points are drawn with a discrete scale of 15 colours, each one representing a different cluster. This low dimensional representation succeed to represent the clustering structure since points with the same colour are closer forming compact groups.

Bottom-up forecast is the leading argument of using the individual load curve clustering. We test the appropriateness of our proposition by getting for a final number of clusters ranging from 2 to 20, 50, 100 and 200 the respective aggregates in terms of load demand. Then, we use KWF as an automatic prediction model for both strategies: prediction of the global demand using the global demand, and the one based on the bottom-up approach.

We use the second year on the dataset to measure the quality of the daily prediction using a rolling basis. Figure 13 reports the prediction error using the MAPE for both the two forecasting strategies. The full horizontal line indicates the annual mean MAPE using the direct method and so it is independent of the number of clusters. For different choices of the number of clusters, the dashed line represent the associated MAPE. All possible clusterings produce then bottom-up forecasts that are better than the one obtained from direct global forecasting.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at Energies 2018, 11, 1893; doi:10.3390/en11071893

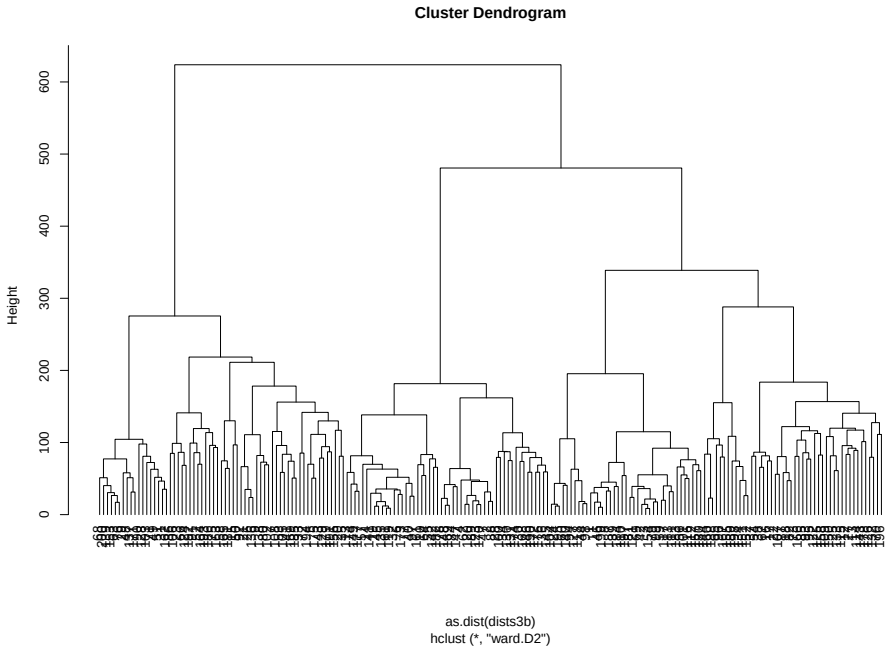

Figure 11. Dendrogram obtained from the WER dissimilarity matrix.

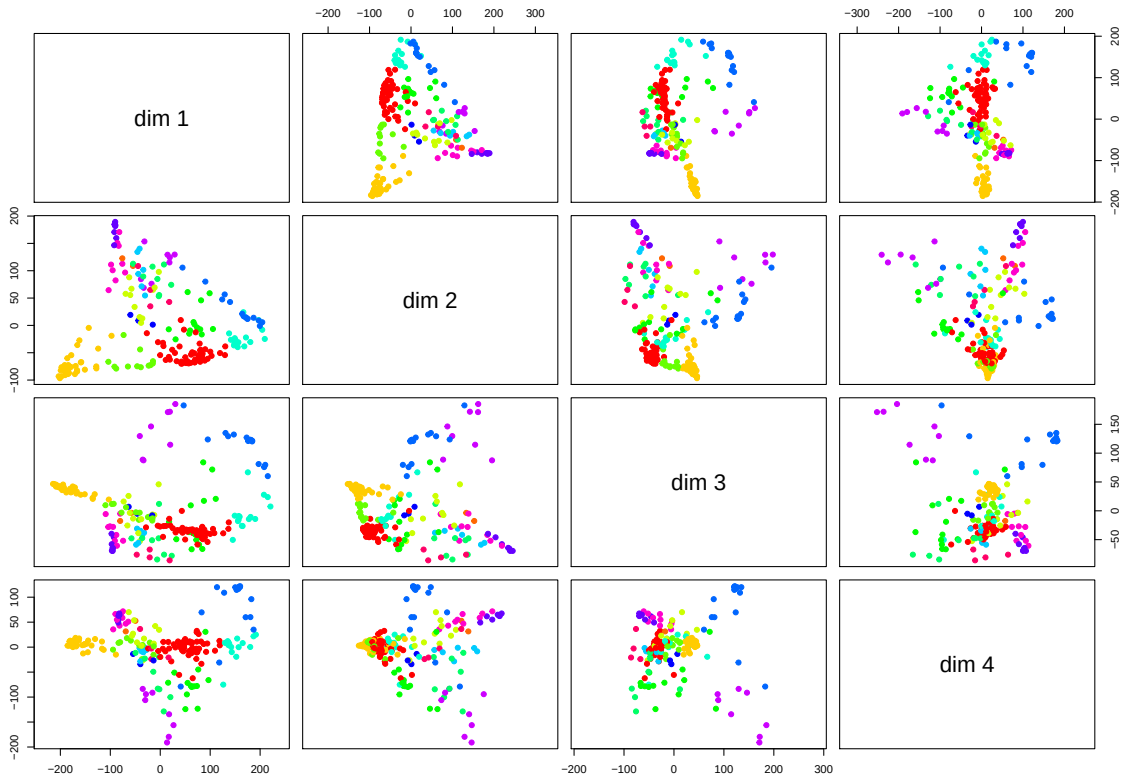

Figure 12. Multidimensional scaling of the WER dissimilarity matrix for the 200 super consumers.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at Energies 2018, 11, 1893; doi:10.3390/en11071893

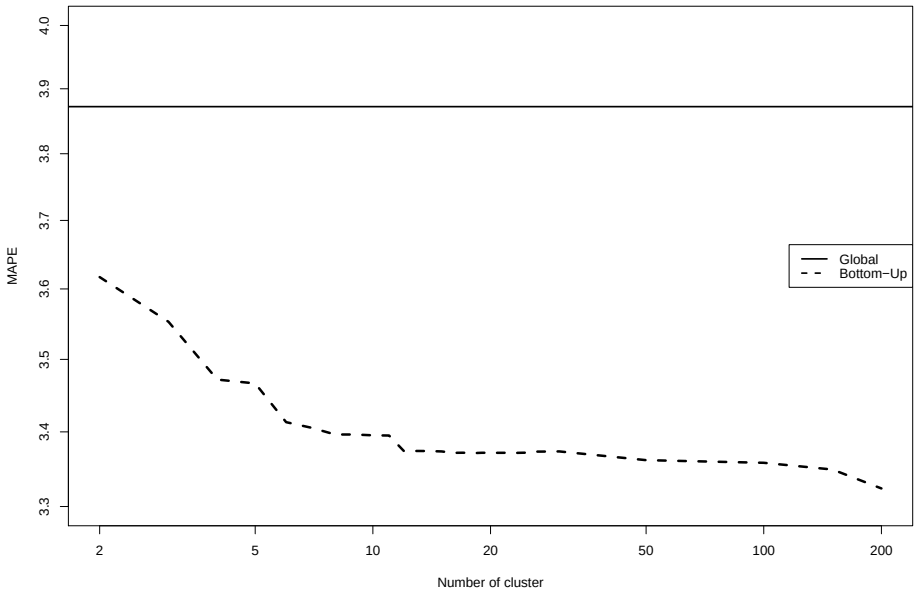

Figure 13. MAPE for the aggregate demand by number of classes for the two strategies: direct global demand forecast (full) and bottom-up forecast (dashed).

9. Discussion

In this final section, we discuss the various choices made as well as some possible extension to cope with multiscale model point of view and how to handle non stationarity.

9.1. Choice of methods

The three main tools are:

- the wavelet decomposition to represent functions and compute dissimilarities. Of course, several other choices could be interesting like splines for bases of functions which are independent of the data or even some data-driven bases like those coming from functional principal component analysis. With respect to these two classical alternatives, (more or less related to a monoscale strategy) the choice of wavelets allows simultaneously a parsimonious representation capturing local features of the data as well as redundant one delivering a more accurate multiscale representation. In addition, from a computational viewpoint, DWT is a very fast: of linear complexity. So to design the super-customers the discrete transform is good enough, for the final clusters, the continuous transform leads to better results. Let us remark that combining wavelets and clustering has recently been considered in [35] from a different viewpoint: details and approximations of the daily load curves are clustered separately leading to two different partitions which are then fused.
- the PAM algorithm and the hierarchical clustering to build the clusters are of very common use and well adapted to their specifc role in the whole strategy. It should be noted that the use of PAM to construct the super customers must necessarily be biased towards a large number of clusters (defining the super customers) so it is useless to include sophisticated model-selection rules to choose an optimal number of clusters since the strategy is used only to define a sufficiently large number of clusters.
- the Kernel-Wavelet-Functional (KWF) method to forecast time-series. The global forecasting scheme is clearly fully modular and then, KWF could be replaced by any other time-series model forecasting. The model must be flexible and easy to automatically be tuned because

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at Energies 2018, 11, 1893; doi:10.3390/en11071893

the modeling and forecasting must be performed in each cluster in a rather blind way. The main difficulty with KWF is to introduce exogenous variables. We could imagine to include a single one quite easily but not a richer family in full generality. Nevertheless, it is precisely when dealing with models corresponding to some specific clusters that it could be of interest to use exogenous variables especially informative, for example describing meteo at a local level or some specific market segment. So some alternative could be considered like generalized additive models (see [36] for a strategy which could be plugged in our scheme).

9.2. Multiscale modeling and forecasting

In fact, such a forecasting strategy combining clustering in individuals and forecasting of the total consumption of each cluster can be also viewed as a multiscale modeling. Indeed a byproduct is a forecasting at different levels of aggregation from the super customers to the total population. So, instead of restricting our attention on the forecasting of the global signal for a given partition we could imagine to combine in time the different predictions given by each piece of the different partitions in such a way that all the levels could contribute to the final forecasting. The way to weight the different predictions could be fixed for all the instants (see [37] for a large choice of proposals) or, on the contrary, time-dependent according to a convenient choice of the updating policy (see the sequential learning strategies already used in the electrical context in [38]). An attempt in this direction can be found in [39].

Another related topic is individual forecasting or prediction. It must be mentioned since it is interesting to have some ideas about the kind of statistical models or strategies used in this especially hard context, due to extreme volatility and wild nonstationarity. [40] examine the short-term (one hour) forecasting of individual consumptions using a sparse autoregressive model which is compared against well-known alternatives (support vector machine, principal component regression, and random forests). In general, exogenous variables are used to forecast electricity consumptions, but some authors focus on the reverse. [41] and [42] are interested in determining household characteristics or customers information based on temporal load profiles of household electricity demand. They use sophisticated deep learning algorithm for the first one and more classical tools for the second one. In the context of customers surveys, [43] use smart meter data analytics for optimal customer selection in demand response programs.

9.3. How to handle non stationarity?

Even if the model KWF is well suited to handle non stationarities in the time-domain, it remains that the clusters of customers are also subjected to some dynamics which could be of interest to model in order to control these changes. A first naive possibility is to periodically recompute the entire process including a new calculation of the super-customers and decide, at some stage if the change is significant to be taken into account. A second possibility could be to directly model the evolution of the clusters. For example, in [44] a time-varying extension of the K-means algorithm is proposed. A multivariate vector autoregressive model is used to model the evolution of clusters' centroids over time. This could help to model the changes of clusters along time but we have to think about a penalty mechanism allowing to make changes in the cluster only when it is useful.

Acknowledgments: This research benefited from the support of the FMJH 'Program Gaspard Monge for optimization and operations research and their interactions with data science', and from the support from EDF and Thales.

Author Contributions: All the authors equally contributed to this work.

Conflicts of Interest: The authors declare no conflict of interest.

Bibliography

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at Energies 2018, 11, 1893; doi:10.3390/en11071893

1. Yan, Y.; Qian, Y.; Sharif, H.; Tipper, D. A Survey on Smart Grid Communication Infrastructures: Motivations, Requirements and Challenges. *IEEE Communications Surveys Tutorials* **2013**, *15*, 5–20.
2. Mallet, P.; Granstrom, P.O.; Hallberg, P.; Lorenz, G.; Mandatova, P. Power to the People!: European Perspectives on the Future of Electric Distribution. *IEEE Power and Energy Magazine* **2014**, *12*, 51–64.
3. Wang, Y.; Chen, Q.; Hong, T.; Kang, C. Review of Smart Meter Data Analytics: Applications, Methodologies, and Challenges. *CoRR* **2018**, *abs/1802.04117*, [1802.04117].
4. Jamme, D. Le compteur Linky : brique essentielle des réseaux intelligents français. conférence, Office franco-allemand pour la transition énergétique, 2017.
5. Alahakoon, D.; Yu, X. Smart Electricity Meter Data Intelligence for Future Energy Systems: A Survey. *IEEE Transactions on Industrial Informatics* **2016**, *12*, 425–436.
6. Ryberg, T. The second wave of smart meter rollouts begin in Italy and Sweden. *https://www.metering.com/regional-news/europe-uk/second-wave-smart-meter-rollouts-begins-italy-sweden/* **2017**.
7. Jiang, H.; Wang, K.; Wang, Y.; Gao, M.; Zhang, Y. Energy big data: A survey. *IEEE Access* **2016**, *4*, 3844–3861.
8. Kaufman, L.; Rousseeuw, P. *Finding Groups in Data: an introduction to cluster analysis*; Wiley, 1990.
9. Liao, T.W. Clustering of time series data—a survey. *Pattern Recognition* **2005**, *38*, 1857–1874.
10. Jacques, J.; Preda, C. Functional Data Clustering: A Survey. *Adv. Data Anal. Classif.* **2014**, *8*, 231–255.
11. Chicco, G. Overview and performance assessment of the clustering methods for electrical load pattern grouping. *Energy* **2012**, *42*, 68 – 80. 8th World Energy System Conference, WESC 2010.
12. le Zhou, K.; lin Yang, S.; Shen, C. A review of electric load classification in smart grid environment. *Renewable and Sustainable Energy Reviews* **2013**, *24*, 103 – 110.
13. Wang, Y.; Chen, Q.; Kang, C.; Zhang, M.; Wang, K.; Zhao, Y. Load profiling and its application to demand response: A review. *Tsinghua Science and Technology* **2015**, *20*, 117–129.
14. Figueiredo, V.; Rodrigues, F.; Vale, Z.; Gouveia, J.B. An electric energy consumer characterization framework based on data mining techniques. *IEEE Transactions on Power Systems* **2005**, *20*, 596–602.
15. Mutanen, A.; Ruska, M.; Repo, S.; Jarventausta, P. Customer Classification and Load Profiling Method for Distribution Systems. *IEEE Transactions on Power Delivery* **2011**, *26*, 1755–1763.
16. Rhodes, J.D.; Cole, W.J.; Upshaw, C.R.; Edgar, T.F.; Webber, M.E. Clustering analysis of residential electricity demand profiles. *Applied Energy* **2014**, *135*, 461 – 471.
17. Kwac, J.; Flora, J.; Rajagopal, R. Household Energy Consumption Segmentation Using Hourly Data. *IEEE Transactions on Smart Grid* **2014**, *5*, 420–430.
18. Sun, M.; Konstantelos, I.; Strbac, G. C-Vine copula mixture model for clustering of residential electrical load pattern data. 2017 IEEE Power Energy Society General Meeting, 2017, pp. 1–1.
19. Alzate, C.; Sinn, M. Improved Electricity Load Forecasting via Kernel Spectral Clustering of Smart Meters. *2013 IEEE 13th International Conference on Data Mining* **2013**, pp. 943–948.
20. Chaouch, M. Clustering-Based Improvement of Nonparametric Functional Time Series Forecasting: Application to Intra-Day Household-Level Load Curves. *IEEE Transactions on Smart Grid* **2014**, *5*, 411–419.
21. Antoniadis, A.; Brossat, X.; Cugliari, J.; Poggi, J.M. Prévision d'un processus à valeurs fonctionnelles en présence de non stationnarités. Application à la consommation d'électricité. *Journal de la Société Française de Statistique* **2012**, *153*, 52 – 78.
22. Misiti, M.; Misiti, Y.; Oppenheim, G.; Poggi, J.M. Optimized Clusters for Disaggregated Electricity Load Forecasting. *REVSTAT – Statistical Journal* **2010**, *8*, 105–124.
23. Quilumba, F.L.; Lee, W.J.; Huang, H.; Wang, D.Y.; Szabados, R.L. Using Smart Meter Data to Improve the Accuracy of Intraday Load Forecasting Considering Customer Behavior Similarities. *IEEE Transactions on Smart Grid* **2015**, *6*, 911–918.
24. Cugliari, J.; Goude, Y.; Poggi, J.M. Disaggregated Electricity Forecasting using Wavelet-Based Clustering of Individual Consumers. Energy Conference (ENERGYCON), 2016 IEEE International, 2016.
25. Labeeuw, W.; Stragier, J.; Deconinck, G. Potential of active demand reduction with residential wet appliances: A case study for Belgium. *Smart Grid, IEEE Transactions on* **2015**, *6*, 315–323.
26. Mallat, S. *A wavelet tour of signal processing*; Academic Pr, 1999.
27. Mallat, S. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE transaction on pattern analysis and machine intelligence* **1989**, *11*, 674–693.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 July 2018 doi:10.20944/preprints201807.0019.v1

Peer-reviewed version available at Energies 2018, 11, 1893; doi:10.3390/en11071893

28. Bosq, D. Modelization, nonparametric estimation and prediction for continuous time processes. In *Nonparametric functional estimation and related topics*; Roussas, G., Ed.; NATO ASI Series, Springer, 1991; pp. 509–529.
29. Poggi, J.M. Prévision non-paramétrique de la consommation électrique. *Revue de Statistique Appliquée* **1994**, *xlii*, 93 – 98.
30. Antoniadis, A.; Paparoditis, E.; Sapatinas, T. A functional wavelet-kernel approach for time series prediction. *Journal-Royal Statistical Society Series B Statistical Methodoloty* **2006**, *68*, 837.
31. Cugliari, J. Prévision non paramétrique de processus à valeurs fonctionnelles. Application à la consommation d'électricité. PhD thesis, Université Paris Sud, 2011.
32. Antoniadis, A.; Brossat, X.; Cugliari, J.; m. Poggi, J. Clustering functional data using wavelets. *International Journal of Wavelets, Multiresolution and Information Processing* **2013**, *11*, 1.
33. Steinley, D.; M. Brusco, A. new variable weighting and selection procedure for k-means cluster analysis. *Multivariate Behavioral Research* **2008**, *43*, 32.
34. R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2018.
35. Jiang, Z.; Lin, R.; Yang, F.; Budan, W. A Fused Load Curve Clustering Algorithm based on Wavelet Transform. *IEEE Transactions on Industrial Informatics* **2017**.
36. Thouvenot, V.; Pichavant, A.; Goude, Y.; Antoniadis, A.; Poggi, J.M. Electricity forecasting using multi-stage estimators of nonlinear additive models. *IEEE Transactions on Power Systems* **2016**, *31*, 3665–3673.
37. Polikar, R. Ensemble learning. In *Ensemble machine learning*; Springer, 2012; pp. 1–34.
38. Gaillard, P.; Goude, Y. Forecasting electricity consumption by aggregating experts; how to design a good set of experts. In *Modeling and Stochastic Learning for Forecasting in High Dimensions*; Springer, 2015; pp. 95–115.
39. Goehry, B.; Goude, Y.; Massart, P.; Poggi, J.M. Forêts aléatoires pour la prévision à plusieurs échelles de consommations électriques. *Proceedings of the 50 èmes Journées de Statistique, Paris Saclay, talk 112* **2018**.
40. Li, P.; Zhang, B.; Weng, Y.; Rajagopal, R. A sparse linear model and significance test for individual consumption prediction. *IEEE Transactions on Power Systems* **2017**, *32*, 4489–4500.
41. Wang, Y.; Chen, Q.; Gan, D.; Yang, J.; Kirschen, D.S.; Kang, C. Deep Learning-Based Socio-demographic Information Identification from Smart Meter Data. *IEEE Transactions on Smart Grid* **2018**.
42. Anderson, B.; Lin, S.; Newing, A.; Bahaj, A.; James, P. Electricity consumption and household characteristics: Implications for census-taking in a smart metered future. *Computers, Environment and Urban Systems* **2017**, *63*, 58–67.
43. Martinez-Pabon, M.; Eveleigh, T.; Tanju, B. Smart meter data analytics for optimal customer selection in demand response programs. *Energy Procedia* **2017**, *107*, 49–59.
44. Maruotti, A.; Vichi, M. Time-varying clustering of multivariate longitudinal observations. *Communications in Statistics-Theory and Methods* **2016**, *45*, 430–443.