Knowl Inf Syst (2018) 57:229–244
https://doi.org/10.1007/s10115-018-1169-y

REGULAR PAPER

Load forecasting through functional clustering and ensemble learning

Fátima Rodrigues[1] · **Artur Trindade**[2]

Received: 16 May 2017 / Revised: 13 November 2017 / Accepted: 30 January 2018 /
Published online: 10 February 2018

Abstract In this paper a load forecasting methodology for 2 days-ahead based on functional clustering and on ensemble learning is presented. Due to the longitudinal nature of the load diagrams, these are segmented using a functional clustering procedure to group together similar daily load curves concerning its phase and amplitude. Next, ensemble learning of extreme learning machine models, developed for several load curves groups, is made to fully integrate the advantages of all models and improve the accuracy of the final load forecasting. The quality of this methodology is illustrated with a real case study concerning load consumption patterns of clients with different economic activities from a Portuguese energy trading company. The forecasting results for 2 days-ahead are good for practical use, yielding a $R^2 = 0.967$.

Keywords Load forecasting · Functional clustering · Extreme learning machine · Ensemble models

1 Introduction

Historically, load forecasting is very important for both transmission and distribution electricity companies. With the liberalization of electricity markets, load forecasting is also extremely important for trading companies that have to purchase electrical energy in bulk at variable prices and sell it to consumers at fixed rates. To reduce this risk, the trader must forecast as precisely as possible the demand of its customers to provide them with good services at a low

✉ Fátima Rodrigues
mfc@isep.ipp.pt

Artur Trindade
artur.trindade@elergone.pt

1 Department of Informatics, Institute of Engineering Polytechnic of Porto (ISEP/IPP), Porto, Portugal

2 Elergone Energia, R. Almeiriga 586, 4450-608 Matosinhos, Portugal

cost. However, the load forecasting is becoming increasingly difficult due to the variability of load curves resulting from dynamic bidding strategies, time-varying electricity price, price-dependent loads, economic cycles, weather conditions, among other factors. Therefore, it is imperative to investigate advanced prediction models.

Several approaches for short-term load forecasting (STLF), that is, the prediction of the system load over an interval ranging from 1 h to 1 week, have been reported in the last decades and can be divided into statistical methods, artificial-intelligence-based methods and hybrid approaches [1,2]. The first one includes linear regression, exponential smoothing, stochastic process, state space and time series methods. A review of statistical methods for electric load forecasting has been given in [3] for instance. Approaches based on artificial intelligence, such as pattern recognition, neural networks, fuzzy neural networks and expert systems, have been widely explored for load forecasting [4]. The extensive work in this domain has shown a trend that is influenced both by increasing complexity of factors that affects consumption and by a trend to apply an increasing number of different technologies that have been proposed and tested. This has led to the development of more accurate load forecast methods and a new era of hybrid load forecasting methods have appeared [5]. Using hybrid model or combining several models has become a common practice that often leads to improved forecasting performance. Given that combining several methods outperforms single methods, we focus on load forecasting through functional clustering and ensemble learning.

In this work, an approach based on the divide-and-conquer paradigm is proposed. First, the original load diagrams database is segmented into distinct groups, accordingly to phase and amplitude of load curves using a functional clustering algorithm. Next, the consumption points of each one of the previously obtained groups are subdivided in accordance with seven climatic regions that we have set for our country. It is our purpose with this more detailed division of the groups, to study the effect of temperature on load forecasting. Then, extreme learning machines (ELMs) of varying complexity are individually trained on each one of the subdivisions made and specific ELMs models are generated. In order to obtain a final prediction, these individual models are combined in a single model, through ensemble learning, which is an effective strategy to improve upon the accuracy of a single learner.

Functional clustering is a recent research area [6,7] and is little explored in load forecasting. Just a few works have recently appeared in the literature such as the work developed in [8], which uses functional clustering and linear regression to short-term peak load forecasting applied to past heating demand data in a district heating system, and in [9], which focuses on predicting electricity consumption by means of a functional linear regression model. ELM is a simple learning algorithm for feedforward neural network, which randomly selects the hidden nodes parameters (including input weights and bias) and analytically determines the output weights of single hidden layer feedforward neural network. In doing this, the training time is substantially reduced while reaching minimum training error. ELM has been adopted to establish prediction models in various real-world problems, showing fast learning speed and good generalization performance. For short-term load forecasting where high volumes must be processed, ELM is evidently suitable to undertake the training and forecasting task. For instance, in [10] an online power load forecasting method based on regularized fixed-memory ELM is proposed to improve the accuracy and speed of load forecasting. Ensemble methods have also been applied and tested for load forecasting. For instance, in [11] the authors proposed a meta-learning system for multivariate time series forecasting as a general framework for using selective ensemble techniques. And in [12] the re-forecast ensembles consist of various time series models combined using least-squares optimization.

The work that will be here described distinguishes from all the previous solutions reported in the load forecasting literature, because it combines functional clustering with ensemble

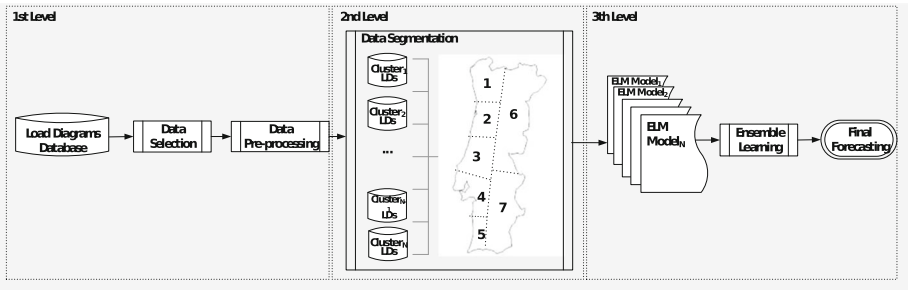

Fig. 1 Block diagram of the proposed methodology

learning of extreme learning machines which is an innovative solution with good results, as we will show.

The remainder of this paper is organized as follows: In Sect. 2, the proposed methodology is presented, as well as a brief explanation of the techniques used to implement it, such as functional clustering, ELM and the theory of ensemble models. In Sect. 3, the methodology is demonstrated with the case study, by starting with a presentation of the clients load diagram database, followed by the segmentation obtained with the functional clustering algorithm and an analysis of the groups reached. Next, the ELMs model generation is explained, followed by a description of the ensemble learning made. In Sect. 4, the results from the experiments made are presented and discussed. In the last section, conclusions and future work are disclosed.

2 Methodology

The load forecast to be supplied by a trader, with clients from different economic activities, must be predicted from the bottom up, i.e., by aggregating the clients load profiles. This aggregation must result into clients groups with load profiles as similar as possible. If this is achieved, better predictions will be also achieved with each one of the groups models. As we are interested in a single forecast, by combining together all single models, trained with different samples, they can compensate for each other, and the final model can reduce the aggregated variance and bias, thus tending to increase the accuracy over the individual models (see Sect. 2.3). The block diagram of the proposed methodology is shown in Fig. 1.

Generically, the methodology is divided into three main levels and is implemented in a modular way, so as to be easy to experiment with different configurations and study several effects on the final load forecasting, such as the influence of temperature conditions on the clusters electricity consumption, the number of different models on ensemble learning and also different combinations schemes.

In order to minimize the temperature forecast error, the country was segmented into seven distinct regions, each of which with homogeneous climate conditions. Portugal is Europe's southwestern extremity. The country is bathed by the Atlantic Ocean along its extensive eastern coast, from north to south, and it is in the coast where most of its population is concentrated. The climate is quite different inland and on the coast. On the coast, the temperatures are milder and the rainfall is higher. In inland, there is a greater difference between the minimum and maximum temperatures, but less rain. There is also a gradual increase in temperature from north into south. Because of these differences and also because

of the population distribution, the country was segmented in two inland regions (6, 7) and five regions along the coast (see Portugal country map in Fig. 1).

The methodology is oriented toward handling short-term forecasting in a unified framework composed by three levels. In the first level, the daily load diagrams for all clients since 2011 are selected from the database, followed by data preprocessing operations such as data cleaning, data reduction and data normalization. In the second level, the load diagrams are segmented concerning its phase and amplitude. Next, the consumption points of each one of the achieved clusters are distributed according to the seven weather regions defined to the country. In the third level, specific ELMs models are developed based on the clustered weather region subdivisions and its ambient temperature. In the last level, the final model prediction is enhanced by an ensemble of the individual models predictions, in order to achieve a higher forecasting accuracy.

2.1 Functional clustering

Cluster analysis groups data objects based only on information in data that describes the objects and their relationships. The goal is that the objects within a group are similar to one another and different from the objects in the other groups. The grouping of objects is based on a distance or similarity function, so that clusters can be formed from objects with a high similarity to each other.

The majority of clustering algorithms have been developed on static data, that is, on data whose feature values do not change with time, or change very little. Due to this, the traditional clustering methods, such as partition, hierarchical or model-based methods are not adequate for group data that arise as curves, designated as functional data or longitudinal data. When applied to functional data, traditional clustering methods are able to detect several extremes and local amplitude variation, but do not take phase variation into account and hence assume that its presence is limited. In [13], it is illustrated that ignoring phase variation of functional data may result in a possible loss of information. To handle this kind of data, some functional clustering algorithms have been developed. Some algorithms are modifications of traditional clustering algorithms in order to handle time series data; others convert time series data so that the traditional clustering algorithms can be directly used. The former approach usually works directly with raw time series data, thus called raw-data-based approach, and the major modification lies in replacing the distance/similarity measure for static data with an appropriate one for time series. The latter approach first converts a raw time series data either into a feature vector of lower dimension or a number of model parameters and then applies a conventional clustering algorithm to the extracted feature vectors or model parameters, thus called feature-based approach [6].

In the system here described the KmL algorithm [14], a new implementation of k-means specifically designed to analyze longitudinal data is used. The algorithm provides several different techniques for dealing with missing values in trajectories, and it runs with distances specifically designed for longitudinal data, like Frechet distance or dynamic time warping, or any user-defined distance. As K-means is a hill-climbing algorithm, in order to avoid the convergence to a local solution, KmL runs several times, varying the starting conditions and/or the number of clusters looked for, and uses the Calinsky and Harabasz quality criterion [15] to select the adequate number of clusters. In practice, it is better to have several criteria so that their concordance will strengthen the reliability of the result. In addition to Calinsky and Harabasz, the KmL also provides two other criteria, Ray and Turi [16] and Davies and Bouldin [17]. One of the advantages of KmL over the existing algorithms is exactly its graphical interface that helps the user to choose the appropriate number of clusters.

2.2 Extreme learning machines

Initially, a NN was applied with satisfactory results, but low performance. Given that performance was critical (the prediction for the next day has to be carried out overnight), more efficient techniques were investigated. In [18], the authors argue that the learning speed of feedforward neural networks is in general far slower than required for two key reasons: (1) the slow gradient-based learning algorithms are extensively used to train neural networks and (2) all the parameters of the networks are tuned iteratively by using such learning algorithms. So they propose a new learning algorithm called extreme learning machine (ELM) for single hidden layer feedforward neural networks. This algorithm tends to provide good generalization performance at *extremely* fast learning speed, hence its name.

The ELM algorithm makes use of single-layer feedforward neural networks (SLFN) having only an input layer, a hidden layer and an output layer. The main concept behind ELM lies in the random initialization of the SLFN weights and biases. Under the condition that the transfer functions in the hidden layer are infinitely differentiable, the optimal output weights for a given training set can be determined analytically. The obtained output weights minimize the square training error. The trained network is thus obtained in very few steps and is very fast to train, which is the main reason we use them to make the load forecasting of each cluster. Moreover, ELM algorithm, unlike SLFN, can be used as an adaptive algorithm. Given a training dataset with M arbitrary distinct samples (x_i, y_i), with $x_i \in R^d$ and $y_i \in R$, the output function of the SLFN with N hidden nodes is modeled as the following sum

$$\sum_{i=1}^{N} \beta_i g(\omega_i \cdot x_j + b_i) = y_j \qquad j = 1, 2, \ldots, M \tag{1}$$

with $g(x)$ being the activation function, w_i the input weights to the ith neuron in the hidden layer, b_i the biases, and β_i the output weights. ELM is completely different from traditional iterative learning algorithms as it randomly selects the input weights and biases for hidden nodes, w_i and b_i, and analytically calculates the output weights β_i by finding the least-square solution. In doing so, it is proven that the training error can still be minimized with even better generalization performance.

In the case where the SLFN would perfectly approximate the data, meaning the error between the output $\hat{y}_i$ and the actual value y_i is zero, according to ELM theory (1) can be expressed in the following matrix form

$$H_n \beta_n = Y_n, \tag{2}$$

where H is the hidden layer output matrix defined as:

$$H = \begin{pmatrix} g(w_1 x_1 + b_1) & \cdots & g(w_N x_1 + b_N) \\ \cdots & \cdots & \cdots \\ g(w_1 x_M + b_1) & \cdots & g(w_N x_M + b_N) \end{pmatrix}$$

and $\beta = (\beta_1 \ldots \beta_N)^{\mathrm{T}}$ and $Y = (y_1 \ldots y_M)^{\mathrm{T}}$.

Given the randomly initialized first layer of the ELM and the training inputs $x_i \in R^d$, the hidden layer output matrix H can be computed. Given H and the target outputs $y_i \in R$ (i.e., Y), the output weights β can be solved from the linear system defined by (2). This solution is given by $\beta = H\dagger Y$, where $H\dagger$ is the Moore–Penrose generalized inverse of the matrix H [19]. This solution for *beta* is the unique least-square solution to Eq. 2. The ELM algorithm then is:

Given a training set $(x_i, y_i), x_i \in R^d$, an activation function $g : R \mapsto R$ and N the number of hidden nodes,

1. Randomly assign input weights w_i and biases $b_i, i \in [1, N]$;
2. Calculate the hidden layer output matrix H;
3. Calculate output weights matrix $\beta = H\dagger Y$.

A more detailed presentation of the algorithm with theoretical proofs is presented in the original paper [18].

2.3 Ensemble models

The ensemble approach is based on multiple uncorrelated models with low error rates. The individual models are combined in some way (typically by voting) into a single ensemble model as follows:

$$\hat{p}_{ens}(t) = \frac{1}{m} \sum_{i=1}^{m} \hat{p}_i(t), \tag{3}$$

where $\hat{p}_{ens}(t)$ is the output of the ensemble models, $\hat{p}_i(t)$ are the outputs of the individual models, and m is the number of models. In [20], it was shown that the variance of the ensemble model is lower than the average variance of all the individual models. Let $p(t)$ denote the true output that we are trying to predict and $\hat{p}_i(t)$ the estimation for this value of model i. Then, we can write the output $\hat{p}_i(t)$ of model i as the true value $p(t)$ plus some error term $e_i(t)$:

$$\hat{p}_i(t) = p(t) + e_i(t). \tag{4}$$

Then, the expected square error of a model becomes:

$$E[(\hat{p}_i(t) - p(t))^2] = E[e_i(t)^2]. \tag{5}$$

The average error for m models is given by:

$$E_{avg} = \frac{1}{m} \sum_{i=1}^{m} E[e_i(t)^2]. \tag{6}$$

Similarly, the expected error of the ensemble as defined in 5 is given by:

$$E_{ens} = E\left[\left(\tfrac{1}{m} \textstyle\sum_{i=1}^{m} \hat{p}_i(t) - p(t)\right)^2\right] = E\left[\left(\tfrac{1}{m} \textstyle\sum_{i=1}^{m} e_i(t)\right)^2\right] \tag{7}$$

Assuming the errors $e_i(t)$ are uncorrelated, i.e., $([e_i(t)e_j(t)] = 0)$ and have zero mean $(E[e_i(t)] = 0)$, we get

$$E_{ens} = \frac{1}{m} E_{avg}. \tag{8}$$

In practice, errors tend to be highly correlated, so they may not be reduced as much as suggested by these equations. The use of ensemble models can, however, lead to a further reduction in the errors. Indeed, the test error of the ensemble is smaller than the average test error of the individual models ($E_{ens} < E_{avg}$). The effectiveness of the ensemble model depends on the accuracy and the diversity of the base models. By segmenting the data into several clusters and developing a distinct forecasting model for each cluster, such effectiveness can be achieved.

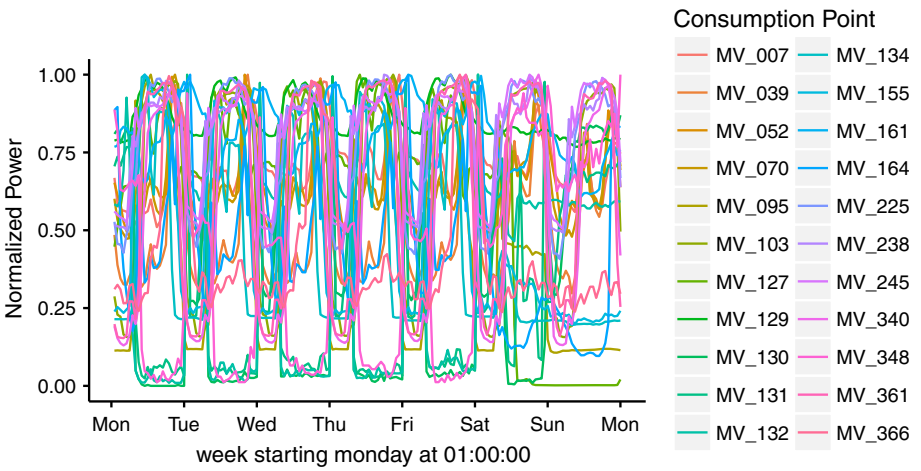

Fig. 2 Weekly load curves

3 Short-term load forecasting

3.1 Client's database description

Elergone is an energy trading company that buys energy in bulk at variable prices and sells it at fixed rates to its clients. If Elergone does not buy the energy necessary to supply their clients, Elergone incurs in losses, so a correct prediction of their needs is fundamental to their business. Their database contains load diagrams that belong to clients with different economic activities such as offices, factories with continuous or weekly laboring, hotels, restaurants, health clubs, schools, shopping malls and supermarkets, among others. For each client, the respective daily load diagram (or load curve) is represented by a vector $l = \{l_1, \ldots, l_h\}$, where l_h is the energy consumed during the period h, $h = 1, 2,\ldots,96$. Such measurements were made every 15 min. These load diagrams have been collected since 2011. Nowadays is composed by a universe of 370 clients. This dataset is publicly available at UCI machine learning repository [21].

An initial analysis of the load curves for all clients shows that the difficulty in defining a satisfactory forecasting model is mainly due to the high variability of the load curves of the different clients. In Fig. 2, a random selection of load curves is plotted. Clearly, the curves are not aligned and show variation both in phase (horizontal) and in amplitude (vertical). This turns the figure very difficult or impossible to interpret, but in the same way it shows the necessity to separate the load curves into homogeneous groups, otherwise it would be very challenging to create a model with an acceptable accuracy.

3.2 Clients database segmentation

The first goal of this project was to stratify the set of load curves into a few homogeneous groups exhibiting similar demand patterns, that is, curves with phase and amplitude similarity. The aim of grouping load diagrams with similar characteristic patterns is to detect few groups which may determine the changes in the load demand.

Several functional clustering algorithms were applied. The feature-based functional algorithms did not lead to significant groups, and from all raw-data-based algorithms tested, the KmL clustering algorithm, available in the KmL R package on CRAN [22], has provided the best results. To correctly separate the load diagrams according to their shape and not by their magnitude, it is important to consider the seasonality of the load diagrams, and for that it is necessary to use at least information gathered over 1 year. As raw data are collected in a 15 min frequency and because the market previsions are made every hour, data were converted from 15 min frequency samples into hourly samples. Even so, due to the high dimensionality of the dataset, the KmL algorithm presented convergence difficulties. So, concerning the number of load diagrams to consider by each client, there were two possibilities: choose some representative weeks of each season or produce an average week. The latter option, called average weekly load diagram (AWLD), has been chosen. The data used to produce the AWLD ranged from July 2013 until June 2014 for all days of the week and for each consumption point, which gives a 370×168 matrix.

As already stated, the main goal of this segmentation is to group data by shape instead of magnitude. Thus, the AWLD for each client is normalized using the min-max normalization [23], resulting for all consumption points a normalized AWLD with values between [0, 1]. To segment the load diagrams into several clusters, the KmL algorithm was applied to the normalized AWLD. This algorithm starts to transform the load diagrams into a ClusterizLongData object where all partitions found are stored. Once an object of class ClusterizLongData has been created, the KmL runs k-means several times, varying the starting conditions and the number of clusters. The range of variation of the number of clusters was preset between 2 and 15, and the default distance function used was the Euclidean distance with Gower adjustment.

The optimal number of clusters is the one that maximizes the Calinski and Harabasz criterion $C(k)$. However, a given criterion may work better on some datasets than others, because of that two distinct criteria were computed. Figure 3 displays two criteria estimated by the algorithm (Calinsky and Harabasz and Davies and Bouldin). Small values of Davies and Bouldin correspond to clusters that are compact, whose centers are far away from each other. Therefore, the cluster configuration that minimizes Davies and Bouldin is the optimal number of clusters.

As can be see in Fig. 3, both criteria are concordant. There is a distinct peak in the Calinski and Harabasz and the minimum value of the Davies and Bouldin is achieved when the number of clusters is also seven. So, for both criteria the best value of k is seven, which indicates that for this database seven clusters is the best partition.

Table 1 describes the clusters obtained in quantitative terms, that is, the number of consumption points included in each one of the clusters and the corresponding aggregate consumption.

Figure 4 displays for each cluster the shape of its average week load diagrams. The results of the segmentation are quite satisfactory, since the load curves of each one of the clusters have distinct curve shapes.

Table 1 Quantitative cluster characterization

Cluster	1	2	3	4	5	6	7
Consumption points	194	77	34	32	19	10	4
Aggregate consumption (%)	52.4	20.8	9.2	8.6	5.1	2.7	1.1

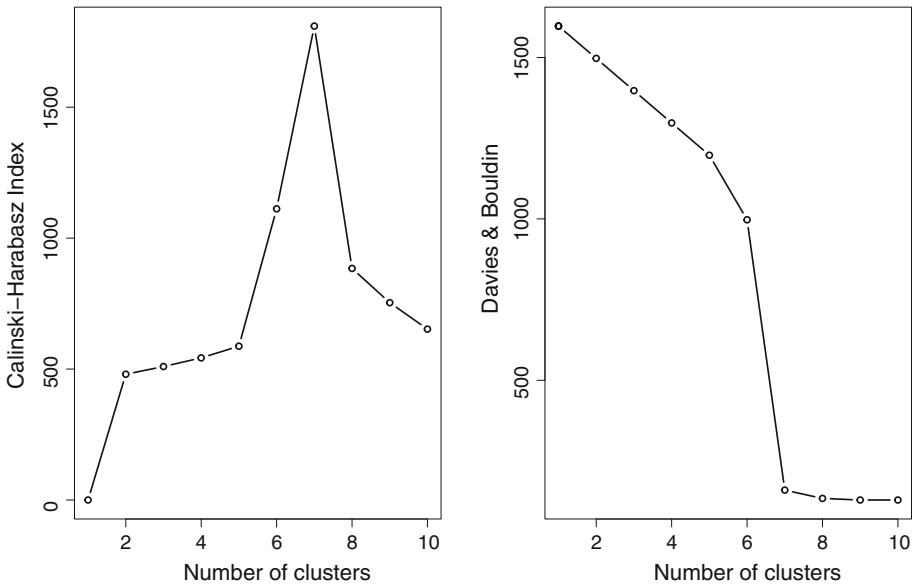

Fig. 3 Calinski and Harabasz and Davies and Bouldin criteria

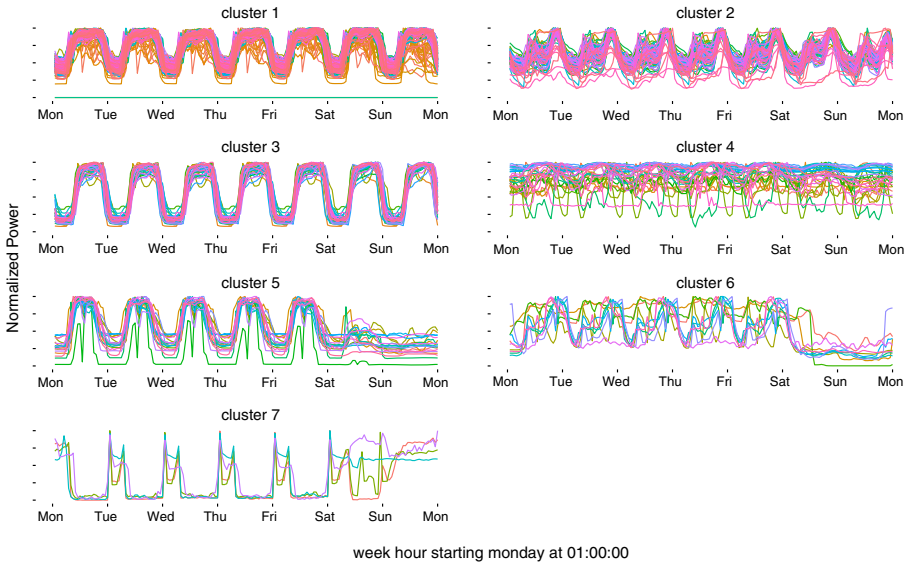

Fig. 4 Average week load curves

By inspecting the clients activity of the clusters, cluster 1 was found to be mainly composed by retailers, cluster 2 essentially includes public medium/lower voltage (MV/LV) substations, cluster 3 basically contains shopping malls, cluster 4 is a mix of factories with weekly and continuous laboring and logistics, cluster 5 includes colleges, schools and offices, cluster 6 includes logistic installations, and finally cluster 7 contains four cogeneration plants.

Table 2 Clusters points distribution over the country regions

Cluster	Regions							Cluster points
	1	2	3	4	5	6	7	
1	17	42	27	52	7	32	17	194
2	60	11	0	3	0	2	1	77
3	4	8	3	11	2	5	1	34
4	4	9	2	11	0	4	2	32
5	4	6	0	9	0	0	0	19
6	1	1	2	5	0	1	0	10
7	0	2	0	1	0	1	0	4

Table 3 Summary of configurations

Config.	Description	No. of ELMs	Includes Temp.
First	1 ELM for each consumption point	370	No
Second	1 ELM for each cluster	7	No
Third	1 ELM for each cluster/region	36	Yes
Fourth	1 ELM for each cluster/region	36	No

3.3 ELMs configuration models

Many factors affect load demand including economic cycles, client activity, among other variables. In addition, load profile of a client can change due to the introduction of efficiency energy measures. Thus, the current year data are important to provide economic cycles and recent behavior of the forecasting point/cluster.

Another factor on load demand on which there are disparate positions in scientific community is temperature. Several previous research works [10,24–26] confirm that outside temperature has a strong influence on electricity consumption because of the use of air conditioning/refrigeration in summer and heating in winter, among other reasons. On the other hand, there are also works [27,28] that argue that temperature hasn't almost any value at all in load forecasting. So, in this work we will study the effect of both aggregate consumption and temperature in the electricity load forecasting with four distinct configurations.

In the first configuration, one ELM is trained for each one of the consumption points. In the second experiment, one ELM is trained for each one of the clusters obtained with the functional segmentation. In these first two configurations, no temperature will be considered. In the following two configurations, each one of the clusters will be divided spreading their consumption points through the seven climate country regions, accordingly to the geographical location of its consumption points. For example, cluster 7 has four cogeneration plants that are spread over three regions, because two are located in region 2, one in region 4 and another one in region 6. Making the same procedure to all consumption points of all clusters resulted in 36 distinct partitions (Table 2). In this way, in the third configuration more accurate temperature is included as input variable in each one of the ELMs subdivision models. The last experiment is equal to the previous, but load forecasting is made without considering temperature.The four experiments are summarized in Table 3.

In the first, second and fourth configurations, because no temperature is used, and in order to best capture the influence of all previous mentioned factors on load forecast-

Table 4 Two training sets of different sizes for forecast two different days

n days	Day to forecast	Current year dataset	Previous year dataset	Two years before dataset
30	10-03-2014	07-01-2014:	08-02-2013:	09-02-2012:
		07-03-2014	09-04-2013	09-04-2012
60	10-05-2014	07-01-2014:	09-02-2013:	08-02-2012:
		07-05-2014	09-07-2013	09-07-2012

ing, several ELMs were trained using data selected from the current year and from the 2 years before the date to forecast. Therefore, each training dataset for each one of the points/clusters/subdivisions is composed of two past time periods (*n*_days) from the day before the day to forecast, and one past period time (*n*_days) before and after the same day to forecast, in the two previous years. Table 4 shows two training sets of size 30 and 60 days to load forecasting two different days. The number of records of each dataset depends on the configuration, for example, in the first configuration the dataset with 30 days for all 370 clients contains 66,600 records ($60 \times 3 \times 370$). When the day to forecast is changed, the datasets are updated accordingly. The ELMs were trained with several datasets with different period time sizes (*n*_days: 20, 30, 45, 60, 90 and 120).

In the third configuration, in order to consider the seasonal variations in temperature by region described in Sect. 2, both electrical data and local temperature data were used and the ELMs models were trained using data from the previous year and from 2 years before (2012, 2013) and evaluated using data from the year 2014.

Like in SLFN, also in ELM the data should be normalized aiming to improve the forecasting results. Therefore, the load diagrams were normalized to the [0, 1] range with the min-max normalization. This kind of normalization allows maintaining the shape of the load curve and thus permits to make a better comparison of the consumption patterns [23].

In the Iberian Electricity Market (MIBEL), trading companies should provide to the market operator their daily load forecasting needs. Buy orders for the following day must be uploaded until 11:00 am (Portuguese local time). At the deadline, the latest information known is the previous day consumption. Thus, the closest period to the day to forecast starts always 2 days before it, which is why this is a 2-day-ahead load forecasting. As a result, the most recent real consumption known is 2 days before the day to forecast, which is used as input to the ELMs. Additionally as input to ELMs it is also used the real consumption 72 h (3 days) and 168 h (7 days) before the day to forecast and a Boolean indicating whether it is a holiday or not. These give a total of four input neurons to the ELM. The output layer has one neuron: the consumption forecast by hour. Besides the training dataset and the input variables, it is also important to optimize the number of neurons in the ELM hidden layer. Smaller number of neurons in the hidden layer speed up the training step, but higher number usually lead to best forecast accuracy. Several experiments with 5, 10, 20 and 75 neurons in the hidden layer were performed.

It should be noted that the methodology can be used to create models to predict longer periods, because the methods that read from database to create the datasets to train the algorithms are parameterized, which guarantees this flexibility.

3.4 Ensemble of ELMs configuration models

Load forecasting is a non-stationary process where data are continuously generated. Therefore, since the information that has been gathered from past samples can become inaccurate,

it is needed to keep learning once new samples become available. One possible way of doing this is using a combination of different models, each of which is specialized on part of the state space. In this work, diverse models are developed each of which is specialized on each one of the consumption points, or on each one of the clusters, or yet on each one of the cluster/region and all contribute to the ensemble.

Many ensemble schemes exist from the most simple ones such as the product rule, the sum rule, the min, max or median rule, the simple weighted, the majority voting, to more elaborate ensemble schemes that use regression, evolutionary programming, neural networks, just to name a few. The key of these ensemble methods is to determine the weight coefficients of each model. Due to the nature of the load forecast to be supplied by an energy trader, we will evaluate two ensemble schemes: the simple sum of the ELMs predictions and the linear regression to accurately deduce the weights of each individual ELM model in the final prediction. The ensemble model consists of a number of randomly initialized ELMs, which each have their own parameters. The model ELM_i has an associated weight h_i which determines its contribution to the prediction of the ensemble. Each ELM is individually trained on the training data, and the outputs of the ELMs contribute to the output of the ensemble $\hat{y}_{ens}$ as follows: $\hat{y}_{ens}(t+1) = \sum_{i=1} h_i \hat{y}_i(t+1)$.

4 Forecasting results and discussion

Several ELM typologies for all the configurations were tested using training data from the current year and from the two previous years, as explained before. In the second configuration, the differences among the experiments rely on the number of neurons in the ELM hidden layer and on the size of the training dataset (n_days). The best results were obtained with an ELM topology with 5 internal neurons, and the best size of the training dataset was: 30 days for cluster 1, 120 days for cluster 5 and 60 days for the other clusters.

In the first, third and fourth configurations, the changes among the experiences rely on the size of the training dataset, 1 or 2 years, and the best results were obtained with a 2-year training data set.

To evaluate and compare the configurations predictive capabilities, the R-squared (R^2) measure will be used. The reason for choosing this metric is because it is an easily interpretable statistic that is more used in industry. As this work was developed for a trade company, where managers are more familiarized with R^2, we used this metric to communicate the results with them. Also, in this article we are interested in evaluating the fit of the forecasting models to past data, rather than acting as a measure of forecast accuracy, so the R^2 measure is more adequate. In the context of predictive models where y is the true outcome, $\bar{y}$ is the average of the true outcomes and f is the model's prediction, R^2 is defined by (9).

$$R^2 = 1 - \frac{\sum_{i=1}^{M}(y_i - f_i)^2}{\sum_{i=1}^{M}(y_i - \bar{y})^2}. \tag{9}$$

In words, R^2 is a measure of how much of the variance in y is explained by the model f and the best possible R^2 is 1.0.

To better understand the influence of temperature and evaluate the effect of aggregation on load forecasting, the predictions obtained with the divisions by regions, in the third and fourth configurations, and also the individual point consumption predictions, first configuration, were added in order to obtain the respective cluster load forecasting and be possible to compare them with the cluster prediction of the second configuration. Table 5 presents the

Table 5 R^2 1st, 2nd, 3rd and 4th configurations

Cluster	% Aggreg. consump.	1st config.	2nd config.	3rd config.	4th config.
1	52.4	0.954	0.709	**0.963**	0.919
2	20.8	0.812	**0.948**	0.835	0.842
3	9.2	0.008	0.959	**0.977**	0.956
4	8.6	**0.894**	0.8	0.719	0.741
5	5.1	0.907	0.9	0.901	0.878
6	2.7	0.590	0.802	**0.828**	0.794
7	1.1	**0.893**	0.174	0.004	0.124

Table 6 R^2 for the two ensemble schemes

Config.	Description	No. of models	Ensemble schemes	
			Sum	Linear regress.
1st	Indiv. consump. points	370	0.748	–
2nd	KmL clusters	7	0.819	0.877
3rd	Clusters versus regions	36	0.965	**0.967**

R^2 of the ELMs models developed for all the configurations. The numbers set in bold give the best R^2 value achieved for each cluster/configuration.

As can be seen from the values presented in Table 5, the best results were achieved with the first three configurations, which lead to the conclusion that smaller subdivisions of the clusters without considering temperature do not lead to better predictions. Analyzing the results of the first three configurations in detail, cluster 2 has obtained the best results without ambient temperature, because cluster 2 points are mainly public MV/LV substations with a mix of consumption profiles. For clusters 4 and 7, the best prediction occurs with individual point consumption prediction, because cluster 4 is a mix of factories and cluster 7 contains four cogeneration plants, whose consumption doesn't depend on temperature. It should be noted that the predictions of these three clusters (2, 4, 7) in the fourth configuration are also better than in the third configuration, which confirms that temperature has no influence on their load forecasting. Cluster 5 has identical values with all the first three configurations which is inconclusive. For the remaining clusters (1, 3, 6), the best prediction occurs considering temperature, given that in some way their activities are influenced by temperature and this reflects in their consumption electricity. Also, concerning the aggregation effect on load forecasting is somewhat beneficial, because the majority of the best predictions were achieved with the second and third configurations, and both grouped the consumption points.

The final step of this methodology consists in making the ensemble of all the individual ELMs models predictions achieved with each one of the configurations. The fourth configuration was not considered to ensemble because it was only developed to test the effect of aggregation and temperature on load forecasting. The R^2 of the two ensemble schemes using the individual consumption points (370 models), the KmL clusters (7 models) and the cluster subdivisions (36 models) with temperature is presented in Table 6.

The linear regression scheme is slightly better than the simple sum scheme for all configurations. This is due to the fact that the regression scheme assigns different weights to the models, depending on their forecast ability, which makes the final forecast more accurate.

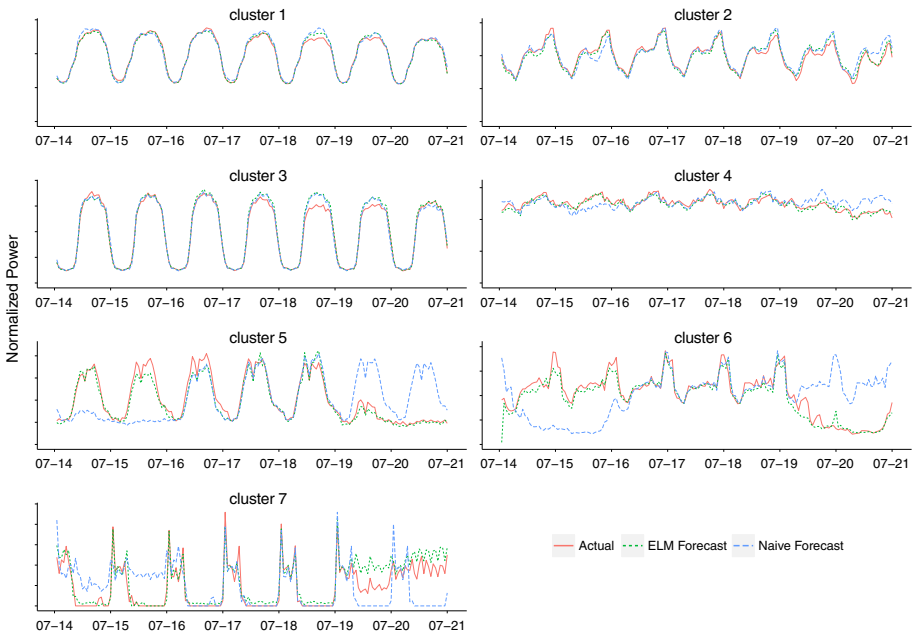

Fig. 5 Best week load forecasting versus real load curve consumption

Concerning the three configurations, the better results were achieved with the third configuration that considers 36 models with temperature, because 69.5% of the aggregate consumption of clients' database (all clusters except 2, 4 and 7, see Table 5) is in some way influenced by the temperature conditions. Another reason is related with the large number of models that this configuration involves which is beneficial to the ensemble scheme.

Next, simple visualization of the model, such as plotting the observed and predicted values, to discover areas of the data where the model does particularly good or bad, will be presented. This type of qualitative information is critical and is lost when the model is gauged only on summary statistics, like R^2. For that, the week average R^2 for all weeks of the 2014 year (test set) was calculated. Figure 5 compares for 7 days, the curve of real load consumption registered, with the curve of our model prediction and the curve whose prediction is equal to the consumption of the 2 days ago, that is, a prediction that is equal to the latest known information, in this case 48 h before the forecast point. The latter curve corresponds to the prediction made by the company prior to the adoption of our methodology. As the figure shows, the curve obtained with the ELMs linear regression ensemble is very similar with the real load consumption.

5 Conclusions and future work

In this paper, a methodology based on functional clustering and on ensemble learning was presented. The experiments showed that the KmL functional clustering algorithm performs well, having correctly separated the load diagrams accordingly to their shape, phase and amplitude. It was also showed that the linear regression ensemble scheme outperforms the sum scheme and can effectively improve the final prediction.

The several configurations tested suggest that the influence of ambient temperature in the electricity consumption is related with the economic activity and the aggregation of load consumption curves with similar characteristics is beneficial to load forecasting.

The final prediction achieved with the methodology here described presented a $R^2 = 0.967$ for a 2-day-ahead prediction, which is good for practical use. An added advantage of the methodology is its low computational cost due to the segmentation of search space and the very fast training speed of the ELMs, which allows the daily load forecasting to be done quickly. As future work, we intend to investigate techniques for real-time load forecasting, such as deep learning.

Acknowledgements The authors would like to acknowledge the support by FEDER Funds through the program "Operacional Regional do Norte - Concurso 07/SI/2012" under the project Ferramenta de Gestão para a Aquisição de Electricidade nos Mercados Grossistas OMIP e OMIE (WATTUP-2013-04/2014).

References

1. Feinberg EA, Genethliou D (2005) Load forecasting. In: Applied mathematics for restructured electric power systems. Springer, pp 269–285
2. Suganthi L, Samuel AA (2012) Energy models for demand forecasting a review. Renew Sustain Energy Rev 16(2):1223–1240
3. Alfares HK, Nazeeruddin M (2002) Electric load forecasting: literature survey and classification of methods. Int J Syst Sci 33(1):23–34
4. Metaxiotis K, Kagiannas A, Askounis D, Psarras J (2003) Artificial intelligence in short term electric load forecasting: a state-of-the-art survey for the researcher. Energy Convers Manag 44(9):1525–1534
5. Badar-Ul-Islam E, Qureshi SA (2011) Comparison of conventional and modern load forecasting techniques based on artificial intelligence and expert systems. Int J Comput Sci 8(5):28–37
6. Liao TW (2005) Clustering of time series data a survey. Pattern Recognit 38(11):1857–1874
7. Jacques J, Preda C (2014) Functional data clustering: a survey. Adv Data Anal Classif 8(3):231–255
8. Goia A, May C, Fusai G (2010) Functional clustering and linear regression for peak load forecasting. Int J Forecast 26(4):700–711
9. Antoch J, Prchal L, de Rosa MR, Sarda P (2008) Functional linear regression with functional response: application to prediction of electricity consumption. In: Functional and operatorial statistics. Springer, pp 23–29
10. Cheng Q, Yao J, Wu H, Chen S, Liu C, Yao P (2013) Short-term load forecasting with weather component based on improved extreme learning machine. In: Chinese Automation Congress (CAC). IEEE, pp 316–321
11. Matijaš M, Suykens JA, Krajcar S (2013) Load forecasting using a multivariate meta-learning system. Expert Syst Appl 40(11):4427–4437
12. Kaur A, Pedro HT, Coimbra CF (2014) Ensemble re-forecasting methods for enhanced power load prediction. Energy Convers Manag 80:582–590
13. Slaets L, Claeskens G, Hubert M (2012) Phase and amplitude-based clustering for functional data. Comput Stat Data Anal 56(7):2360–2374
14. Genolini C, Falissard B (2010) KmL: k-means for longitudinal data. Comput Stat 25(2):317–328
15. Caliński T, Harabasz J (1974) A dendrite method for cluster analysis. Commun Stat Theory Methods 3(1):1–27
16. Ray S, Turi RH (1999) Determination of number of clusters in k-means clustering and application in colour image segmentation. In: Proceedings of the 4th international conference on advances in pattern recognition and digital techniques, Calcutta, India, pp 137–143
17. Davies DL, Bouldin DW (1979) A cluster separation measure. IEEE Trans Pattern Anal Mach Intell 2:224–227
18. Huang G-B, Zhu Q-Y, Siew C-K (2006) Extreme learning machine: theory and applications. Neurocomputing 70(1):489–501
19. Rao CR, Mitra SK (1971) Generalized inverse of matrices and its applications, vol 7. Wiley, New York
20. Bishop CM et al (2006) Pattern recognition and machine learning, vol 4. Springer, New York
21. Trindade A (2015) UCI machine learning repository. http://archive.ics.uci.edu/ml/datasets/ElectricityLoadDiagrams20112014

22. Genolini C, Alacoque X, Sentenac M, Arnaud C (2015) kml and kml3d: R packages to cluster longitudinal data. J Stat Software 65(4):1–34. http://www.jstatsoft.org/v65/i04/
23. Han J, Kamber M (2006) Data mining: concepts and techniques. Morgan Kaufmann, Burlington
24. Taylor JW, Buizza R (2003) Using weather ensemble predictions in electricity demand forecasting. Int J Forecast 19(1):57–70
25. Fan S, Chen L, Lee W-J (2008) Short-term load forecasting using comprehensive combination based on multi-meteorological information. In: IEEE/IAS industrial and commercial power systems technical conference. IEEE, pp 1–7
26. Taylor JW, Buizza R (2002) Neural network load forecasting with weather ensemble predictions. IEEE Trans Power Syst 17(3):626–632
27. López M, Valero S, Senabre C, Aparicio J, Gabaldón A (2011) Development of a model for short-term load forecasting with neural networks and its application to the electrical Spanish market. In: 8th international conference on the European Energy Market (EEM). IEEE, pp 321–326
28. Llanos J, Saez D, Palma-Behnke R, Nunez A, Jimenez-Estevez G (2012) Load profile generator and load forecasting for a renewable based microgrid using self organizing maps and neural networks. In: The 2012 international joint conference on neural networks (IJCNN). IEEE, pp 1–8

Fátima Rodrigues received the B.Sc. degree from the University of Minho, Portugal, in 1989, the M.Sc. degree from the University of Porto, Portugal, in 1997, and the Ph.D. degree in Computer Science from the University of Minho in 2000. She is currently a Coordinator Professor of Computer Engineering with the Polytechnic Institute of Porto, Portugal. Her research areas include Data Mining, Machine Learning, Recommender Systems, Automatic Assessment.

Artur Trindade received the B.Sc. degree from the University of Porto, Portugal, in 2005. In 2014 after teaching during 8 years in Portuguese high school he received the M.Sc. degree from the University of Porto, Portugal. From April 2014 to August 2015 he was Forecasting Models researcher in Elergone Energia. He is currently a Tableau Expert in Siemens SA in Lisbon, Portugal. His areas of interest include Data Mining, Machine Learning, Process Automation and Optimization, aiming to Decision Systems as well as Visualization.