Improved Electricity Load Forecasting via Kernel Spectral Clustering of Smart Meters

Carlos Alzate, Mathieu Sinn
IBM Research - Ireland
Mulhuddart, Dublin 15, Ireland
{carlos.alzate, mathsinn}@ie.ibm.com

***Abstract*—This paper explores kernel spectral clustering methods to improve forecasts of aggregated electricity smart meter data. The objective is to cluster the data in such a way that building a forecasting models separately for each cluster and taking the sum of forecasts leads to a better accuracy than building one forecasting model for the total aggregate of all meters. To measure the similarity between time series, we consider wavelet feature extraction and several positive-definite kernels. To forecast the aggregated meter data, we use a periodic autoregressive model with calendar and temperature information as exogenous variable. The data used in the experiments are smart meter recordings from 6,000 residential customers and small-to-medium enterprises collected by the Irish Commission for Energy Regulation (CER). The results show a 20% improvement in forecasting accuracy, where the highest gain is obtained using a kernel with the Spearman's distance. The resulting clusters show distinctive patterns particularly during hours of peak demand.**

***Index Terms*—load forecasting; clustering; disaggregation; smart meter data**

I. Introduction

Accurate load forecasts are essential for control and planning operations in electrical grids and electricity markets. Short-term estimates of the future consumption (typically, 24-hours ahead) are needed as decision support for unit commitment, balancing supply and demand on different perimeters of the network, and buying/selling energy in day-ahead spot markets. There exists a large body of literature on load forecasting methods, including seasonal time series models with exogeneous factors (e.g., PARX, SARIMAX), neural networks, semi-parametric approaches (e.g., additive models), and exponential smoothing (see, e.g., [1] [2] [3] [4]). Typical features of electrical load series are seasonal cycles (e.g., daily/weekly patterns due to social behavior) and strong effects of meteorological variables (e.g., extreme temperatures leading to an increase in demand due to electrical heading or air conditioning).

The ongoing deployment of smart meters in most industrialized countries (notably, US, Europe, China and Australia) offers new insights into how electrical loads emerge. Smart meters are digital devices measuring the electricity consumption of individual households with a high time resolution (typically from 60 to 10 minutes time intervals). The key purpose of smart meters is more accurate billing of electricity consumption which allows for dynamic pricing, e.g., imposing higher prices during hours of peak demand, and offering lower prices during night hours or on weekends. Mining smart meter data has found considerable interest in the past few years. A topic of particular interest has been the identification of typical customer profiles according to individual demand patterns. The present paper investigates clustering of smart meter data to improve forecasts of disaggregated load series. The key question is: can we cluster the smart meters and build a forecasting model separately for each cluster in such a way that the sum of the forecasts per cluster is more accurate than the forecasts using one model fitted to the total aggregate of all meters? In [5], experiments on smart meter data from industrial customers in France indicate that, indeed, clustering load series helps to improve the accuracy of forecasts for the total aggregate. On the other hand, a recent study by [6] suggests that the forecasting accuracy monotonically increases with the number of aggregated series, hence, disaggregation would not help to build more accurate forecasting models. Another direction is [7] where smart meter series are clustered in the temporal dimension, e.g., to build separate models for different seasons or week days.

This paper applies kernel spectral clustering (KSC) and different kernels for time series to find aggregations of smart meters which help to improve the overall forecasting accuracy. In contrast to existing approaches (e.g., [8] [9]), the presented methodology allows for predicting the cluster membership of unseen time series, and model selection in a learning setting. The data used in the experiments of this paper are 6,000 smart meter series collected by the Irish Commission for Energy Regulation (CER) [1]. The CER data set includes residential customers and small-to-medium enterprises. For a subset of the residential meters, panel data with socio-demographic information is available (however, for privacy reasons, the geographic location of the smart meters is anonymized).

The remainder of this paper is organized as follows. Section II describes the CER data set. In Section III, we summarize classical spectral clustering and describe the basics of kernel spectral clustering (KSC). Section IV contains a description of different similarity measures used for comparing time series. In Section V, we outline the forecasting algorithm based on load disaggregation. Experimental results are provided in Section VI and concluding comments are given in Section VII.

[1] http://www.cer.ie/en/information-centre-reports-and-publications.aspx?article=5dd4bce4-ebd8-475e-b78d-da24e4ff7339

1550-4786/13 $31.00 © 2013 IEEE
DOI 10.1109/ICDM.2013.144

IEEE computer society

II. Data Description and Preprocessing

The data used in the experiments of this paper were collected as part of an electricity smart metering pilot study conducted by the Irish Commission for Energy Regulation (CER) in 2009-2010. In total, data from around 6,445 residential and commercial customers were recorded. The study took place from July 14, 2009 to December 31, 2010 (536 days). Electricity consumption (in kW) was measured half-hourly, so the data set consists of $536 \cdot 48 = 25,728$ recordings per series. Originally, the data set was grouped into three different customer types: *residential* (66%), *small-to-medium enterprises* (SME) (7%) and *others* (27%).

In our smart meter data analysis, we include only smart meters with less than 100 missing values. Any missing values in these time series are linearly interpolated. Some abnormal time series are removed from the data set. Outliers are detected using the kurtosis on the reconstructed signals using the discrete wavelet transform and the Haar wavelet at level 5. Reconstructed signals with kurtosis larger than 10 are considered outlying. The wavelet decomposition level and the kurtosis threshold are determined by analyzing the distribution of the kurtosis values of the whole set. This procedure reduces the number of residential, SME and other smart meter series to $3,994$, 444 and 897, respectively.

III. Clustering

In this section, we summarize the basics of the clustering techniques used in this work.

A. Spectral Clustering

Consider a dataset of N d-dimensional data points $\mathcal{D} = \{x_i\}_{i=1}^N, x_i \in \mathbb{R}^d$ and a non-negative symmetric similarity $s_{ij} \geq 0, s_{ij} = s_{ji}, i, j = 1, \ldots, N$ between every pair of data points. An intuitive way of representing the data points and their similarities is through an undirected graph, where each data point is a node and each edge has an associated weight given by s_{ij}. The similarity matrix (or affinity matrix) $S \in \mathbb{R}^{N \times N}$ is a symmetric matrix with ij-entry $S_{ij} = s_{ij}$ and the graph degree matrix is defined as $D = \text{diag}(d_1, \ldots, d_N)$, $d_i = \sum_{j=1}^N s_{ij}$ is the degree of the i-th node. Several properties of the graph can be explained through spectral graph theory which is the study of the eigenspectrum of graph Laplacian matrices [10]. Typical graph Laplacians are: the unnormalized Laplacian defined as $L = D - S$, the symmetric normalized Laplacian $L_{\text{SYM}} = D^{-1/2} L D^{-1/2} = I_N - D^{-1/2} S D^{-1/2}$ and the non-symmetric normalized Laplacian $L_{\text{RW}} = D^{-1} L = I_N - D^{-1} S$ denoted L_{RW} because it is related to a random walk on the graph. These Laplacians are positive semi-definite matrices where the smallest eigenvalue equals to zero and the corresponding eigenvector is the vector of all ones 1_N, except in the case of L_{sym} where the eigenvector is $D^{1/2} 1_N$.

For spectral clustering, we are particularly interested in the spectrum of the Laplacians when the graph has k connected components denoted by $\mathcal{A}_1, \ldots, \mathcal{A}_k$. This means that the graph is composed of k subgraphs with no similarities between them. The eigenvalue 0 of L and L_{RW} equals the number of connected components k and the eigenspace of 0 is spanned by the indicator vectors $q^{(1)}, \ldots, q^{(k)}$ where $q_i^{(p)} = \mathcal{I}[x_i \in \mathcal{A}_p]$, $i = 1, \ldots, N, p = 1, \ldots, k$. For L_{SYM}, the eigenspace of 0 is spanned by the vectors $D^{1/2} q^{(p)}$, $p = 1, \ldots, k$ [10], [11]. In practice, these k eigenvectors are not necessarily cluster membership indicators since the numerical eigensolvers converge to some orthonormal basis of the eigenspace. However, they are *piecewise constant*[2]. The partial solution provided by the eigenspace of 0 is then considered as a new representation of the input data points. Simple clustering methods such as k-means can be applied on this new representation in order to obtain the final grouping.

B. Kernel Spectral Clustering

A different view of spectral clustering in the context of kernel machines has been described in [12]. This method sets spectral clustering as a model-based weighted kernel PCA problem in a primal and dual setting typical of least squares support vector machines (LS-SVM). The methodology allows out-of-sample extensions and model selection in a learning setting with training, validation and test stages. The kernel spectral clustering (KSC) problem is summarized as follows. Given training data $\mathcal{D} = \{x_i\}_{i=1}^N, x_i \in \mathbb{R}^d$ and the number of clusters k, assume the following clustering model:

$$e_i^{(l)} = w^{(l)^T} \varphi(x_i) + b_l, \tag{1}$$

where $\varphi : \mathbb{R}^d \to \mathbb{R}^{d_h}$ is the mapping to a high-dimensional feature space, b_l are bias terms, $i = 1, \ldots, N$ and $l = 1, \ldots, n_e$. The projections $e_i^{(l)}$ represent the latent variables of a set of n_e binary clustering indicators given by $\text{sign}(e_i^{(l)})$ which can be combined to form the final k groups in a similar style as in multiclass kernel machines [13], [14]. The KSC primal problem is defined as:

$$\begin{aligned} \min_{w^{(l)}, e^{(l)}, b_l} \frac{1}{2} \sum_{l=1}^{k-1} w^{(l)^T} w^{(l)} - \frac{1}{2N} \sum_{l=1}^{k-1} \gamma_l e^{(l)^T} D^{-1} e^{(l)} \\ \text{such that } e^{(l)} = \Phi w^{(l)} + b_l 1_N \end{aligned} \tag{2}$$

where $e^{(l)} = [e_1^{(l)}, \ldots, e_N^{(l)}]$, $l = 1, \ldots, n_e$, $D = \text{diag}([\text{deg}_1, \ldots, \text{deg}_N])$ is the degree matrix, $\text{deg}_i = \sum_{j=1}^N s(x_i, x_j)$ is the degree of the i-th data point, $s : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is a similarity function, Φ is the $N \times d_h$ feature matrix $\Phi = [\varphi(x_1)^T; \ldots; \varphi(x_N)^T]$ and $\gamma_l \in \mathbb{R}^+$ are regularization constants. After constructing the Lagrangian of (2) and satisfying the Karush-Kuhn-Tucker (KKT) conditions of the Lagrangian, the following dual problem is obtained:

$$D^{-1} M_D \Omega \alpha^{(l)} = \lambda_l \alpha^{(l)}, l = 1, \ldots, k-1, \tag{3}$$

where $M_D = I_N - 1_N 1_N^T D^{-1} / (1_N^T D^{-1} 1_N)$ is a $N \times N$ centering matrix, Ω is the $N \times N$ kernel matrix with ij-th entry $\Omega_{ij} = K(x_i, x_j)$, $K : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is a

[2] A vector $\alpha = [\alpha_1, \ldots, \alpha_N]$ is piecewise constant relative to a partitioning $\mathcal{A}_1, \ldots, \mathcal{A}_k$, if $\alpha_i = \alpha_j$ for data points x_i and x_j in the same cluster $\mathcal{A}_p$

Mercer kernel satisfying $K(x_i, x_j) = \varphi(x_i)^T \varphi(x_j)$, $\lambda_l = 1/\gamma_l$ is the l-th ordered eigenvalue $\lambda_1 \geq \ldots \geq \lambda_N$ and $\alpha^{(l)} \in \mathbb{R}^N$ is the corresponding eigenvector. The clustering model (1) can now be expressed in terms of the dual variables $\alpha^{(l)}$: $e_i^{(l)} = \sum_{j=1}^N \alpha_j^{(l)} K(x_i, x_j) + b_l$, with bias terms $b_l = 1_N^T D^{-1} \Omega \alpha^{(l)} / 1_N^T D^{-1} 1_N$. At the training stage, each point x_i is represented as a binary codeword in $\mathbb{R}^{k-1}$ by $x_i \mapsto \text{sign}([e_i^{(1)}, \ldots, e_i^{(k-1)}]^T)$, with the k most occurring codewords becoming the representation of the k clusters and thus, forming the cluster codebook $\mathcal{C}$. If there exists more than k different codewords, the cluster membership assignment of these points is performed by choosing the codeword from the codebook that minimizes the Hamming distance.

Consider the ideal clustering case of k clusters $\mathcal{A}_1, \ldots, \mathcal{A}_k$ with zero inter-cluster similarity. Then, the following holds:

1) $D^{-1} M_D \Omega$ has $k-1$ eigenvectors with eigenvalue 1.
2) These eigenvectors are piecewise constant on the partition.
3) All eigenvectors have zero mean due to M_D.

Properties 1 and 2 show that the eigenvectors with large eigenvalue have clustering information. Property 3 combined with the orthogonality of the eigenvectors leads to data points in the same cluster represented as a single point in $\mathbb{R}^{k-1}$ and each cluster mapped to a different orthant. These properties allows the use of the aforementioned binary encoding and decoding schemes. The main advantage of this model-based spectral clustering approach is the possibility to compute cluster membership of unseen (also called out-of-sample) data points. For an arbitrary data point x, the projected variables can be obtained by: $\hat{e}^{(l)}(x) = \sum_{j=1}^N \alpha_j^{(l)} K(x_j, x) + b_l$ with encoding vector: $\text{sign}([\hat{e}^{(1)}(x), \ldots, \hat{e}^{(k-1)}(x)])^T$. The cluster membership of the new point x can then be obtained in the same fashion as in the training stage (i.e., via comparing the encoding vector with the codebook).

It has been shown in [12] that if i) the validation set $\mathcal{D}^{\text{val}}$ has been sampled i.i.d. from the same probability distribution underlying the training set $\mathcal{D}$; ii) the k clusters are well-represented in $\mathcal{D}$ and $\mathcal{D}^{\text{val}}$ and iii) $K(x, z) = 0$ if x and z are in different clusters; then the clusters are represented as lines in the $\mathbb{R}^{k-1}$ projection space and each line is in a different orthant. This structural property of the projections on out-of-sample data allows the design of model selection criteria in order to determine the number of clusters and the kernel parameters. Given $\mathcal{D}^{\text{val}}$ and the estimated cluster memberships $\hat{q}^{\text{val}}$, the balanced linefit (BLF) criterion introduced in [12] is a function of the cluster collinearity and cluster balance: $\text{BLF}(\mathcal{D}^{\text{val}}, \hat{q}^{\text{val}}) = \eta f(\text{collinearity}(\mathcal{D}^{\text{val}}, \hat{q}^{\text{val}})) + (1 - \eta)\text{balance}(\mathcal{D}^{\text{val}}, \hat{q}^{\text{val}}))$ where $f : \mathbb{R}^k \to \mathbb{R}$ (e.g., mean, min), the collinearity is measured by computing the percentage of total variance explained by the largest eigenvalue of the p-th cluster sample covariance matrix in the projection space. Cluster balance is defined as $\min\{\log|\mathcal{A}_p|\}/\max\{\log|\mathcal{A}_p|\}, p = 1, \ldots, k$, $|\mathcal{A}_p|$ is the size of the p-th cluster and $0 \leq \eta \leq 1$ is a user-defined constant defining the influence given to the cluster collinearity with respect to the cluster balance[3]. The BLF values are bounded between 0 and 1, attaining its maximal value when the clusters are perfectly collinear and have the same sizes.

IV. Comparing Time Series

Several distances and similarity measures for time series have been proposed in the literature [8], [9]. However, the applicability of these similarities to the framework of kernel-based methodologies is restricted. This issue is due to the fact that most similarities are not positive definite which is a required condition in kernel methods. In this Section, we summarize the distances and similarities used to compare smart meter time series.

A. Wavelet Feature Extraction

Extracting features using wavelets in power load time series in the context of clustering has been discussed in [5], [15], [16]. These schemes propose to perform hierarchical clustering on the wavelet approximation coefficients on each time series at a given level [5]. The relative energies per decomposition level of the wavelet details are used as input to k-means in [15] with DB6 (Daubechies 6) as the chosen wavelet function. For a given orthogonal wavelet (e.g., DB family) and maximum decomposition level L, the total energy E of a time series x is given by: $E(x) = \|x\|_2^2 = \|a^{(L)}\|_2^2 + \sum_{j=1}^L \|d^{(j)}\|_2^2$. Thus, a time series x is represented as an L-dimensional feature vector containing the relative energy of the details at each decomposition level: $x \mapsto x^{\text{wav}} = [\text{rel}_1, \ldots, \text{rel}_L]^T$ where: $\text{rel}_j = \|d^{(j)}\|_2^2 / \sum_{j'=1}^L \|d^{(j')}\|_2^2$. A local kernel such as the RBF kernel can then be applied on this new representation of the load time series: $K_{\text{wav}}(x, z) = \exp(-\|x^{\text{wav}} - z^{\text{wav}}\|_2^2 / \sigma_{\text{wav}}^2)$ with parameter σ_{wav}^2.

B. RBF Kernel with Spearman's distance

The Spearman's rank correlation r_s is a measure of statistical dependence between two variables. It is defined as the standard Pearson correlation after ranking the variables. The key point of this measure is the capability to detect non-linear relationships between variables as long as they are related through a monotonic function. The RBF kernel with Spearman's distance between x and z is defined as: $K_{\text{SP}}(x, z) = \exp(-\phi_{\text{SP}}(x, z)/\sigma_{\text{SP}}^2)$, where $\phi_{\text{SP}}(x, z) = \sqrt{1 - r_s(x, z)^2}$ is a metric distance leading to a positive definite kernel and σ_{SP}^2 is a free parameter.

C. VAR Kernels

A vector autoregressive (VAR) kernel for time series in the context of probabilistic modelling has been proposed in [17]. The likelihood function $p_\theta(x)$ of different parameters θ in the VAR model is considered as features describing a multivariate time series x. In order to compare two multivariate time series $x \in \mathbb{R}^{n_1 \times d}$ and $z \in \mathbb{R}^{n_2 \times d}$, the product of their features $p_\theta(x)$ and $p_\theta(z)$ is formed and θ is integrated out. Given a VAR

[3] The original BLF is defined with the mean cluster collinearity and cluster balance is defined without taking the logarithm on the cluster sizes.

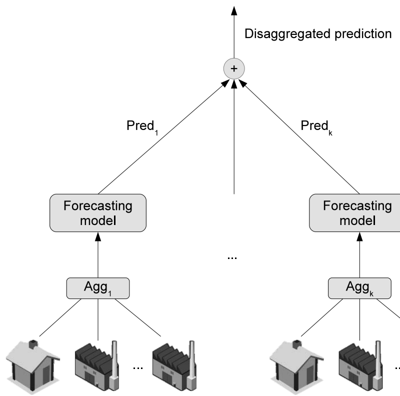

Fig. 1: Schematic showing the proposed disaggregated forecasting scheme. A forecasting model is fit on the aggregated load of each cluster (aggregator). The k predicted signals are then summed up to form the disaggregated prediction. The number of aggregators k and the time series being fed to each aggregator are determined via kernel spectral clustering.

model of order p, the negative definite autoregressive kernel between x and z is defined as: $\phi_{\text{VAR}}(x,z) = C_N + (1-\zeta)\log(\det(X^TX+\Delta^{-1})) + \zeta\log(\det(X^TX+Y^Y+\Delta^{-1}))$, where X and Y are matrices formed with the explanatory and response variables respectively, $0 \le \zeta \le 1$ is a free parameter, C_N, Δ are a constant and a diagonal matrix depending on the sizes of X and the order p. A positive definite kernel can then be formed by: $K_{\text{VAR}}(x,z) = \exp(-\phi_{\text{VAR}}(x,z)/\sigma^2_{\text{VAR}})$ where σ^2_{VAR} is a free parameter. For more information, the Reader is referred to [17].

D. Global Alignment Kernels

Dynamic Time Warping (DTW) distances have become a widely-used standard to compare time series. The main idea is to find the best alignment between two time series of different length. Fast global alignment kernels [18] generalize the DTW concept towards positive-definite similarities that can be used in kernel machines. Given two multivariate time series $x, z \in \mathbb{R}^{n\times d}$ of equal length n, the triangular global alignment (TGA) becomes: $K_{\text{TGA}}(x,z) = \prod_{m=1}^{n}\exp(-\phi_{\text{TGA}}(x^{(m)}, z^{(m)}))$ where $x^{(m)} \in \mathbb{R}^d$ is the m-th row of x, $\phi_{\text{TGA}}(x^{(m)}, z^{(m)}) = \|x^{(m)} - z^{(m)}\|^2/2\sigma^2_{\text{TGA}} + \log\left(2 - \exp(-\|x^{(m)} - z^{(m)}\|^2/2\sigma^2_{\text{TGA}})\right)$, and σ^2_{TGA} is a free parameter. In this particular case (i.e., the time series having equal length n), the triangular global alignment kernel is a modified RBF kernel ignoring the temporal structure.

V. Improving Forecasting via Disaggregation

Accurate models for load forecasting are essential to the planning and operation of a utility company. The approach followed in this work is to improve the forecasting of a global load time series (e.g., total regional or national demand) by first performing clustering on the smart meter time series, fitting a forecasting model on each cluster and summing the prediction per cluster to obtain the total disaggregated forecast. Figure 1 shows a schematic of the approach.

A. Forecasting Models

We use a periodic autoregressive model with exogenous variables (PARX) to model the aggregated load [1]. Consider a univariate time series $y_t, t = 1,\ldots,N$ of half-hourly measurements for a sample of $N_d = N/48$ days. A PAR model of order p can be written as: $y_t = \beta_{s,1}y_{t-1} + \beta_{s,2}y_{t-2} + \ldots + \beta_{s,p}y_{t-p} + C_s + \varepsilon_{s,t}$ where $\beta_{i,s}$ are the autoregressive parameters varying across the total number of seasons N_s, C_s is a seasonally varying bias term and $\varepsilon_{s,t}$ is a standard zero-mean white noise with variance σ^2_s (seasonal heteroskedasticity). In this case, we introduced dummy variables to model the monthly and weekly seasonals. Intraday seasonal patterns are assumed to be captured by the PAR parameters thus, the number of seasons is $N_s = 48$. The order of the PAR model p can be set heuristically to the lowest multiple of 48 that gives satisfactory in-sample performance. Temperature readings from a reference location (Dublin airport) are also incorporated into the model as exogenous variables to identify their influence for each half-hour of the day. Note that the CER dataset does not provide geographical information about individual smart meters hence, no local weather can be included in the forecasting model. Consider $y_{h,d}$ the value of the load measured at half-hour h of day d, $h = 1,\ldots,48, d = 1,\ldots,N_d$. The final PARX($p$) system can be written in vector autoregression (VAR) form as: $\Psi_0 y_d = \Psi_1 y_{d-1} + \Psi_2 y_{d-2} + \Psi_3 X^{\text{exo}}_d + C + \varepsilon_d$, where $y_d = [y_{1,d}, y_{2,d}, \ldots, y_{48,d}]^T$, Ψ_0, Ψ_1, Ψ_2 are matrices containing the PAR coefficients $\beta_{i,s}$ which can be estimated using ordinary least squares. The matrix X^{exo}_d contains all exogenous variables for calendar and temperature information. One day-ahead forecasts can be estimated as: $\hat{y}_{d+1} = \Psi_0^{-1}(\Psi_1 y_d + \Psi_2 y_{d-1} + \Psi_3 X^{\text{exo}}_{d+1})$, where Ψ_0 is always nonsingular. For more details on PAR models, the Reader is referred to [1], [2].

B. Clustering for Forecasting

We propose the use of KSC models as the aggregating methodology and PARX models as forecasting models. The number of aggregators (or clusters) k is determined by the BLF model selection criterion. Consider $x_{i,t}$, the i-th smart meter time series at instant t with $i = 1,\ldots,N$. The global load is defined as $y_{\text{global},t} = \sum_{i=1}^{N} x_{i,t}$. After fitting a forecasting model on the global load, we have the forecasted global load $\tilde{y}_{\text{global},t}$. Given an associated clustering of k groups $\mathcal{A}_1,\ldots,\mathcal{A}_k$, the consumption of the p-th cluster is given by $y^{(p)}_t = \sum_{j\in\mathcal{A}_p} x_{j,t}, p = 1,\ldots,k$ with forecast given by $\tilde{y}^{(p)}_t$. The disaggregated forecast is then obtained by $\tilde{y}_{\text{disagg},t} = \sum_{p=1}^{k}\tilde{y}^{(p)}_t$. The main idea is to find a grouping such that the error on the disaggregated forecast $\tilde{y}_{\text{global},t}$ is minimized. Note that, this scheme was initially introduced in [5] using wavelets as feature extractors, hierarchical clustering with Ward's method as initial aggregators and seasonal ARIMA models for forecasting. This methodology also starts with a

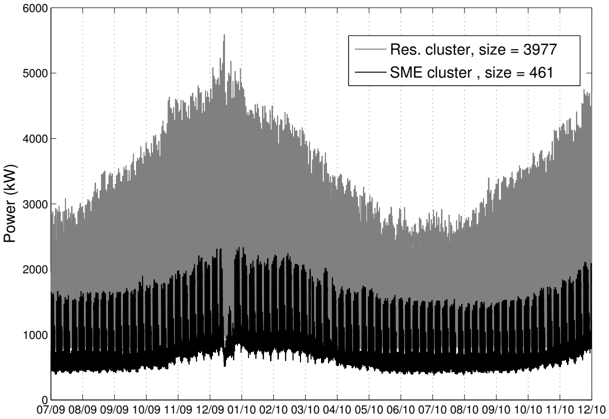

Fig. 2: Aggregated load obtained by KSC using the VAR kernel. The median agreement with the ground truth (residential and SME class labels) in terms of the ARI is 0.84. The percentages of correctly clustered series is 98.55% and 91% on the residential and SME groups respectively. Weekly cycles are very visible on the SME group also displaying less temperature influence compared to the residential group.

arbitrarily large number of clusters and iteratively reduces them by optimizing a cross-prediction distance index greatly increasing the computational burden. Our proposed methodology is similar in essence but differs from [5] on several aspects such as the use of kernel versions of spectral clustering, the automatic determination of the number of clusters, the use of different similarities for time series and the lack of ad-hoc supervised distances for decreasing the forecasting error.

VI. Experimental Results

All experiments reported were performed in MATLAB 8.0 on an Intel Core i7, 2.2 GHz, 16 GB RAM. The data have been partitioned into training, validation and test sets with several randomizations unless specified otherwise. When comparing the clustering results with an external partitioning, the adjusted Rand index (ARI) is used. This external clustering performance measure is bounded between -1 and 1 attaining its maximal value when the two partitionings completely agree. Each time series has been standardized (zero-mean and standard deviation 1) for the clustering analysis.

A. Kernels and Model Selection

The parameters of the kernel used in this work are detailed as follows. Unless stated otherwise, the σ^2 parameters of each kernel were obtained by searching in $\{0.25, 0.5, 1, 2, 4\}$ multiplied by $\text{median}(\phi(x, z))$ for all x, z in the training set and selecting the factor that maximizes the BLF on 10 randomized training and validation sets.

1) VAR kernel: The order of the VAR process p is set to 1 (1-day lag). This value was found empirically using grid search and maximizing the average BLF on 500 randomly selected time series (10 randomizations). The constant $\zeta = 0.5$ was fixed to the value suggested in [17].

2) Triangular Global Alignment kernel: For TGA kernels, searching in $\{0.5, 1, 2, 5, 10\}$ multiplied by $\text{median}(\phi_{\text{TGA}}(x, z))\sqrt{n}$ is suggested in [18]. As before, the factor that maximizes the BLF on 10 randomized training and validation sets is selected.

TABLE I: Summary of model selection and disaggregated forecast results. The Spearman kernel outperforms with an improvement of 20.55%. *kmeans-DB6-11* refers to applying k-means on the wavelet relative energy of the details. *Random* clusters were obtained with 50 randomizations.

Kernel	Model Selection		Disaggregated Forecast Baseline MAPE = 3.26%		
	$k^\star$	BLF($k^\star$)	$k^\star$	MAPE($k^\star$)	Gain
VAR	7	0.49	13	2.85%	12.68%
TGA	5	0.63	8	2.61%	20.04%
Spearman	**6**	**0.59**	**6**	**2.59%**	**20.55%**
RBF-DB6-11	4	0.53	5	3.02%	7.36%
$k$means-DB6-11			16	2.93%	
Random			3	2.93% ± 0.03	

B. Discriminating Between Residential and SME Series

This first experiment aims at assessing the discriminatory power of the different similarities between times series. In this setting, we have a ground truth corresponding to class labels of the time series (residential and SME). The training set consists of 600 randomly sampled time series (300 from the residential set and 300 from the SME set). The test set consists of the whole data (3,994 time series). The VAR kernel gives the maximal agreement between the clustering results and the class labels with a median ARI of 0.84 composed of 58 out of 3,994 (1.45%) residential time series incorrectly clustered and 40 out of 444 (9%) SME time series assigned to the wrong cluster. As a point of comparison, kernel k-means on the whole VAR kernel matrix (i.e., without out-of-sample extension) with 50 randomizations of the starting points gives an ARI of 0.75 ± 0.05. Figure 2 shows the aggregated load obtained by the VAR kernel for the whole time period. The SME cluster displays the typical strong weekly periodicity associated to the enterprise profiles.

C. Disaggregation Results

The PARX models where trained on the first 400 days of the data leaving 136 days for testing. All Mean Absolute Percentage Errors (MAPE) are reported on the test set. The model order $p = 48$ were determined by taking the smallest multiple of 48 that still gives satisfactory in-sample performance. Figure 3 shows the MAPE on the test set with respect to k. The forecasting error increases as k increases indicating an optimal value at a small number of clusters. The unsupervised BLF criterion agrees with the supervised forecasting objective with respect to preference to a rather small k. Table I summarizes the best result for each kernel.

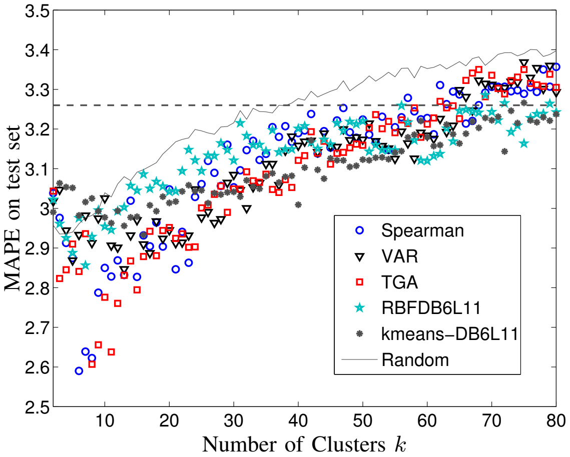

Fig. 3: MAPE on test set (136 days) using the disaggregation scheme. The gray dash-dot line corresponds to MAPE on the whole signal. The lowest MAPE is achieved with the kernel based on the Spearman correlation and $k = 6$ clusters.

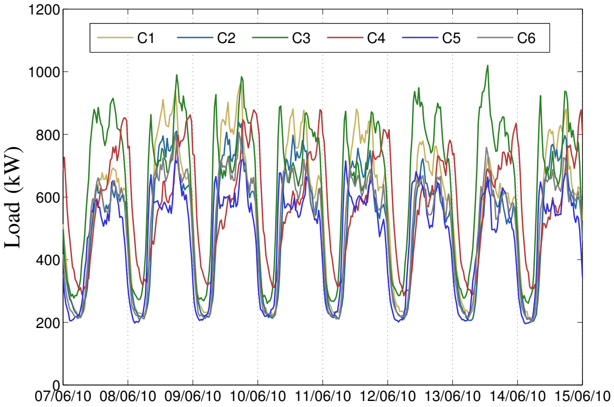

Fig. 4: Visualization of the aggregated loads for Spearman, $k = 6$ on the disaggregated forecast. The clusters differ in shape and magnitude and cluster 4 captures a late evening peak not present on the other clusters.

The kernel based on the Spearman distance outperforms in this setting. A maximum BLF value of 0.59 is achieved at $k = 6$ which also corresponds to the number of clusters needed for the maximal improvement (20.55%) on the disaggregation scheme. Figure 4 shows the aggregated meter series for each of the $k = 6$ clusters obtained based on the Spearman distance (8 days in June). A first observation is that all clusters are of the same order of magnitude. An interesting qualitative finding is the difference in the peak behavior between clusters. In particular, cluster 4 shows pronounced late-evening peaks and overall, has a larger consumption overnight. While all clusters exhibit peak demand in the late afternoon (a typical feature of residential load series), cluster 4 and 5 also show peaks in the early morning, particularly on weekdays.

VII. Conclusion

We have proposed a methodology to improve the prediction accuracy of a global signal composed of highly aggregated smart meter time series of electricity consumption. The proposed methodology uses KSC for finding groups of time series that are similar according to some metric. Periodic autoregressive models are then trained on each cluster and predictions per cluster are computed. The sum of these predictions corresponds to the disaggregated forecast. Improvements of more than 20% in terms of the MAPE are achieved by the approach using a similarity based on the Spearman's rank correlation.

References

[1] M. Espinoza, C. Joye, R. Belmans, and B. De Moor, "Short-term load forecasting, profile identification and customer segmentation: A methodology based on periodic time series," *IEEE Transactions on Power System*, vol. 20, no. 3, pp. 1622–1630, August 2005.

[2] P. Franses and R. Paap, *Periodic Time Series Models*. Oxford University Press, 2003.

[3] J. W. Taylor, "Short-term load forecasting with exponentially weighted methods," *IEEE Transactions on Power Systems*, vol. 27, no. 1, pp. 673–679, 2012.

[4] D. W. Bunn and E. D. Farmer, *Comparative models for electrical load forecasting*. Eds. Wiley, 1985.

[5] M. Misiti, Y. Misiti, G. Oppenheim, and J. Poggi, "Optimized Clusters for Disaggregated Electricity Load Forecasting," *REVSTAT - Statistical Journal*, vol. 8, no. 2, pp. 105–124, 2010.

[6] D. Ilic, P. Goncalves da Silva, S. Karnouskos, and M. Jacobi, "Impact assessment of smart meter grouping on the accuracy of forecasting algorithms," in *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, 2013, pp. 673–679.

[7] C. Flath, D. Nicolay, T. Conte, C. Dinther, and L. Filipova-Neumann, "Cluster analysis of smart metering data," *Business and Information Systems Engineering*, vol. 4, pp. 31–39, 2012.

[8] T. Schreiber and A. Schmitz, "Classification of time series data with nonlinear similarity measures," *Physical Review Letters*, vol. 79, no. 8, pp. 1475–1478, 1997.

[9] T. Warren Liao, "Clustering of time series data - a survey," *Pattern Recognition*, vol. 38, no. 11, pp. 1857–1874, 2005.

[10] F. R. K. Chung, *Spectral Graph Theory*. American Mathematical Society, 1997.

[11] U. von Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.

[12] C. Alzate and J. A. K. Suykens, "Multiway spectral clustering with out-of-sample extensions through weighted kernel PCA," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 2, pp. 335–347, February 2010.

[13] B. Schölkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA, USA: MIT Press, 2001.

[14] J. A. K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle, *Least Squares Support Vector Machines*. World Scientific, Singapore, 2002.

[15] A. Antoniadis, X. Brossat, J. Cugliari, and J. Poggi, "Clustering functional data using wavelets," in *COMPSTATS*, 2010, pp. 697–704.

[16] M. Misiti, Y. Misiti, G. Oppenheim, and J. Poggi, "Clustering signals using wavelets," in *Computational and Ambient Intelligence*, vol. 4507, 2007, pp. 514–521.

[17] M. Cuturi, "Autoregressive kernels for time series," arXiv:1101.0673, Tech. Rep., Jan 2011.

[18] ——, "Fast global alignment kernels," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2011.