García Hinde, S., Gómez Verdejo, V. & Martínez-Ramón, M. (2020). Forecast-informed power load profiling: A novel approach. *Engineering Applications of Artificial Intelligence, 96*, 103948.

DOI: 10.1016/j.engappai.2020.103948

Forecast-Informed Power Load Profiling: a Novel Approach

Óscar García Hinde[a], Vanessa Gómez Verdejo[a], Manel Martínez-Ramón[b]

[a]*Department of Signal Theory and Communications, Universidad Carlos III*
[b]*Department of Electrical and Computer Engineering, University of New Mexico*

Abstract

Power load forecasting plays a critical role in the context of electric supply optimization. The concept of load characterisation and profiling has been used in the past as a valuable approach to improve forecasting performance as well as problem interpretability.

This paper proposes a novel, fully fledged theoretical framework for a joint probabilistic clustering and regression model, which is different from existing models that treat both processes independently. The clustering process is enhanced by simultaneously using the input data and the prediction targets during training. The model is thus capable of obtaining better clusters than other methods, leading to more informative data profiles, while maintaining or improving predictive performance.

Experiments have been conducted using aggregated load data from two U.S.A. regional transmission organizations, collected over 8 years. These experiments confirm that the proposed model achieves the goals set for interpretability and forecasting performance.

Keywords: Power Load, Forecasting, Profiling, Clustering, Machine Learning, Probabilistic Model

1. Introduction

During the past decade, the use of renewable energy has continued to grow steadily, representing 18.9% of the energy consumed in the European Union

Email addresses: oghinde@tsc.uc3m.es (Óscar García Hinde), vanessa@tsc.uc3m.es (Vanessa Gómez Verdejo), manel@unm.edu (Manel Martínez-Ramón)

[8] and the 11% in the United States [2] in 2018. This trend is expected to continue in the future due to the need to reduce carbon emissions worldwide. The global tendency towards green energy has the effect of turning energy production into a non deterministic process, as it depends mainly on the local availability of solar radiation and wind. Not only do both these sources add uncertainty to the energy production problem, but they also introduce hard constraints due to their limited availability and seasonal nature. This makes the balance between energy availability and demand a complex problem that can only be tackled by demand side-management and energy storage strategies. Accurate power load profiling and forecasting models have thus become vital aids in this context.

In particular, the use of load profiling can mitigate the uncertainties introduced by the diversity of stakeholders in generation, transmission, and distribution agents [14]. In general, the characterization and profiling of electricity consumption or modelling of common behaviours at user level and its applications have become an important research topic in the literature related to renewable energies.

In [27] the profiling methodology is justified by the need for an accurate customer billing assessment. In [19], load profiling is presented with the purpose of designing efficient low voltage distribution networks in residential areas in South Africa. A similar approach is found in [22], where load profiles were characterized in São Paulo, Brazil. Examples of the use of these profiles are "transformer rating selection and management, load diversity evaluation and to determine the expected load profile in any preset point of the distribution network", among others. A different application can be found in [7], where authors group customers based on their different profiles in order to design customized tariffs based on their energy usage preferences. The topic was discussed also in [1], where machine learning is used to detect habits in energy consumption. A similar application, but restricted to modelling energy consumption in buildings is presented in [10, 9]. Load models based on load profiles that mimic observed load are presented in [28].

Regarding power load forecasting, many efforts have been made to design reliable models. Usually, load prediction models are categorized into short-term (from minutes to less than one week) and long-term (from more than one week to several years) [42, 45]. For the purpose of short-term load forecast, traditional methods include linear prediction models that use historical time-series of the observed load. The main structures used in these models are moving-average (MA), auto-regressive moving-average (ARMA) or auto-

regressive integrated moving-average (ARIMA) [4]. The most widely used parameter optimization algorithm consists of the minimization of the mean square error (MMSE) (see, e.g. [17], [34]). Further attempts at using AR or ARMA regression include the use of support vector machines (SVM) [5] due to their robustness [38] and ability to construct nonlinear versions [29]. Some approaches include an SVM-MA [32] or an SVM-ARIMA structure [26]. Both of these works include the use of kernels [40] in order to attain nonlinear properties. Other approaches, such as [30], include the use of Gaussian processes (GP) [36]. While the MMSE criterion can easily be implemented online if the structure is linear [18], SVMs and GPs require block training, which limits the number of data that can be used.

The use of neural networks has become a widespread practice in power-load forecast. Early works using the standard multilayer perceptron neural network include [47, 41]. More recent works use recurrent networks adapted for time series analysis called long short-term memory (LSTM) networks [20, 23], convolutional neural networks [11] and others. In general, due to their complexity, these methods show excellent performance, but they need to be trained with very large data-sets to obtain satisfactory results. These techniques also allow online and batch training.

Most of these models can and have been enhanced with the inclusion of multi-source data (which is also allowed by the use of kernels [6]). In particular, besides historical load time-series, the most used data source in power-load forecast consists of weather parameters such as outdoor temperature, humidity, solar radiation intensity, dew point temperature, wind speed, rainfall and others.

Tying profiling and forecasting together, it has been shown that good profiling can also improve forecasting accuracy when it is used as a form of data selection for model training. The main efforts in this topic take two different approaches. The first one is to directly produce an interpretation of the behaviour of the load time-series in time, space or across users. The second one focuses on the usage of clustering simply for the improvement of load forecasting accuracy. The main idea behind the second approach consists of constructing prediction models specialized in each one of the clusters. This way, it is expected that the possible nonlinear relationships between the predictor (input data) and the regressor (forecast load) can be locally approximated by linear functions or, at least, by less complex nonlinear structures. Clustering techniques are the most popular for both load characterization as well as joint profiling and forecasting [46, 35], due to

their easy implementation and direct interpretation.

Clustering is a classic idea in machine learning and it was first presented as an application for load forecast in [44], in which the authors use K-means clustering and linear regression for load forecasting. In fact, most of the works related to load profiling use clustering based on the K-means algorithm (see e.g. [3]). For example, [37] uses K-means to cluster load data and produce a detailed analysis of the results by grouping the clusters into yearly seasons. In [13], the authors assume that the data fits a linear ARMA model and show that, when the data is clustered, this assumption produces better prediction results for a large number of substations of the Belgian power grid. Nonlinear approaches have also been applied together with clustering. In [15] the authors use K-means to cluster aggregate data supplied by smart meters. The test data is then classified as belonging to one of such clusters and then processed through traditional neural networks to produce a forecast. In this work it is assumed that the input-output relationship is still nonlinear, but the clustering allows less complex structures.

K-means can be seen as a simplification of the Gaussian mixture model (GMM), which is trained with the well known Expectation Maximization algorithm [31]. An example of GMM used to identify typical daily electricity usage profiles of multiple buildings can be found in [24]. Authors actually use two clustering levels, the first one (intra-building) being a GMM and the second one (inter building) a hierarchical clustering.

Other authors, such as [33], take advantage of projections of the data into higher dimension Hilbert spaces through the use of kernelized versions of clustering techniques [39].

In all previous papers where clustering is used to enhance load prediction, the authors construct a cascaded scheme with two distinct stages: a clustering algorithm followed by a battery of regression functions that are trained independently. Alternative models attempt to train both stages at the same time. A variant of the GMM for its application to regression is the approach known as Gaussian Process Mixture Model [43]. A version of this method was applied to power-load forecasting in [25]. This model produces a probabilistic interpretation of the output but, since the input space is not clustered, it cannot provide a meaningful profiling of the data.

In this paper we present a new approach to simultaneous clustering and regression with the aim of providing good profiling characteristics while maintaining solid forecasting performance. The main theoretical novelty of the approach lies in the definition of a probabilistic model that performs a joint

training of the clustering and regression stages. This results in a clustering of the input data that is actively informed by the forecasting process. The clusters achieved by the model will therefore offer a better, more informative representation of the problem than if they were a function of the input data alone.

The remainder of this paper is organised as follows:

- Section 2 fully defines the theoretical framework for the proposed model. It also provides a simple synthetic example to illustrate its capabilities.
- Section 3 gives a detailed description of the experiments that have been carried out, in which we apply the model to power load forecasting. It includes a thorough analysis of the resulting profiles and predictive function distribution as well as a performance evaluation. Several reference models have been included in the experiments for comparison.
- In Section 4 we present our conclusions and final thoughts on the experiments as well as a brief discussion of future lines of work to improve and expand the models capabilities.

The full model, as used in this study, is implemented in python and is available at the following GitHub repository: `https://github.com/OGHinde/Clusterwise_Linear_Model`.

2. The Clusterwise Linear Model

Following on the idea of using an initial clustering phase to select groups of samples to later train separate regressors, we propose a unified probabilistic model that integrates regression into the clustering process. For this reason, we have called it the Clusterwise Linear Model (CWLM). We first assume that the observations, or input data, are generated by a standard Gaussian mixture model (GMM) with K components. Next, we consider that each component of the mixture model is associated to a linear regression model that generates the output targets. The novelty of this approach lies in the fact that both stages are coupled: not only does the input space clustering influence the linear regression on the output space, but also the regression process affects the overall clustering of the data.

Let's consider a regression problem defined by an observation data matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]^\top$ and a target vector $\mathbf{y} = [y_1, \ldots, y_N]^T$, where $\mathbf{x}_i \in \Re^D$ is the i-th observation and $y_i \in \Re$ is its corresponding target.

The proposed model starts by considering that the data distribution can be approximated by a mixture of K Gaussians,

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \tag{1}$$

where $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$ are, respectively, the mean and covariance matrix of the k-th Gaussian component, and π_k is the prior probability that sample $\mathbf{x}$ has been generated by the k-th Gaussian component.

Next, we introduce the following set of K linear models,

$$y = \mathbf{w}_k^\top \mathbf{x} + \epsilon_k, \tag{2}$$

where $\mathbf{w}_k$ are the linear regression weights of the k-th component, including the bias term[1], and ϵ_k is assumed to be Gaussian noise with zero mean and variance β_k^{-1}.

Thus, given that observation $\mathbf{x}$ has been generated by the k-th Gaussian component, its corresponding target value $\mathbf{y}$ will be generated by the k-th linear model. Therefore the probability distribution for y becomes

$$p(y|\mathbf{x}, \boldsymbol{\theta}) = \mathcal{N}(y|\mathbf{w}_k^\top \mathbf{x}, \beta_k^{-1}). \tag{3}$$

The mixture distribution for the target variables can therefore be stated as

$$p(y|\mathbf{x}, \boldsymbol{\theta}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(y|\mathbf{w}_k^\top \mathbf{x}, \beta_k^{-1}), \tag{4}$$

where $\boldsymbol{\theta}$ includes all model parameters: π_k, which are the prior cluster probabilities; $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, which contain all component mean vectors $\boldsymbol{\mu}_k$ and covariance matrices $\boldsymbol{\Sigma}_k$ for the input clustering stage; and $\mathbf{w}$ and $\boldsymbol{\beta}$ contain all regression weight vectors $\mathbf{w}_k$ and estimation noise precisions β_k.

2.1. Probabilistic representation

From a probabilistic standpoint, this model can be represented by the graph depicted in Figure 1.

[1] $\mathbf{x}$ is considered to be extended with a constant term of value 1 to account for the bias term

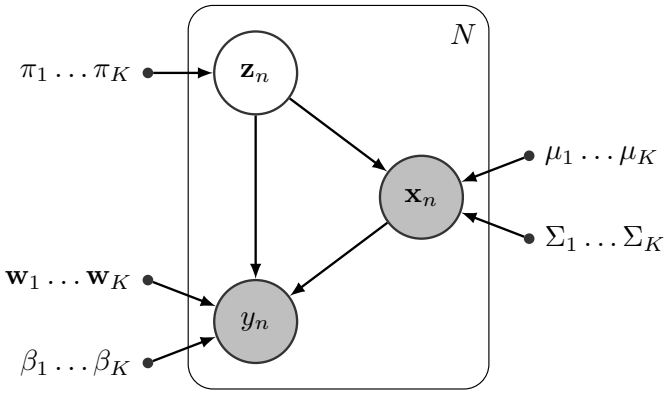


Figure 1: Graphical representation for the CWLM. Shaded nodes indicate variables that are observed during the training phase, whereas non-shaded nodes indicate latent variables. The smaller solid nodes indicate deterministic model parameters.

The model assumes that a set of latent variables $\mathbf{Z} = [\mathbf{z}_1, \dots, \mathbf{z}_N]^T$ exists, where each $\mathbf{z}_i = \{z_{i,k}\}_{k=1}^K$ is modeled such that only the k-th entry of these vectors equals 1 and the rest is zero, indicating that $\mathbf{x}_i$ has been generated by the k-th Gaussian mixture component and, consequently, y_i has been generated by the k-th linear regressor. The prior distribution of these variables is defined as

$$p(z_{i,k} = 1) = \pi_k. \tag{5}$$

where $0 \leq \pi_k \leq 1$ and $\sum_{\forall k} \pi_k = 1$.

The graphical model in Figure 1 leads us to the following complete-data likelihood function

$$p(\mathbf{Z}, \mathbf{X}, \mathbf{y}|\boldsymbol{\theta}) = p(\mathbf{Z}|\boldsymbol{\theta})p(\mathbf{X}|\mathbf{Z}, \boldsymbol{\theta})p(\mathbf{y}|\mathbf{Z}, \mathbf{X}, \boldsymbol{\theta}), \tag{6}$$

where

$$p(\mathbf{Z}|\boldsymbol{\theta}) = \prod_{i=1}^{N} \prod_{k=1}^{K} \pi_k^{z_{i,k}} \tag{7}$$

$$p(\mathbf{X}|\mathbf{Z}, \boldsymbol{\theta}) = \prod_{i=1}^{N} \prod_{k=1}^{K} \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)^{z_{i,k}} \tag{8}$$

$$p(\mathbf{y}|\mathbf{Z}, \mathbf{X}, \boldsymbol{\theta}) = \prod_{i=1}^{N} \prod_{k=1}^{K} \mathcal{N}(y_i|\mathbf{w}_k^\top \mathbf{x}_i, \beta_k^{-1})^{z_{i,k}}. \tag{9}$$

Therefore, the complete-data likelihood becomes

$$p(\mathbf{Z}, \mathbf{X}, \mathbf{y}|\boldsymbol{\theta}) = \prod_{i=1}^{N}\prod_{k=1}^{K}\left[\pi_k \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\mathcal{N}(y_i|\mathbf{w}_k^\top \mathbf{x}_i, \beta_k^{-1})\right]^{z_{i,k}}. \tag{10}$$

From here we can now compute the complete-data log likelihood,

$$\begin{aligned}&\ln p(\mathbf{Z}, \mathbf{X}, \mathbf{y}|\boldsymbol{\theta}) = \\ &= \sum_{i=1}^{N}\sum_{k=1}^{K} z_{i,k}\left[\ln \pi_k + \ln \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + \ln \mathcal{N}(y_i|\mathbf{w}_k^\top \mathbf{x}_i, \beta_k^{-1})\right]\end{aligned} \tag{11}$$

2.2. Model inference through Expectation Maximization

The Expectation Maximization algorithm (EM) is a common approach to find the optimal values for the parameters of a model that depends on latent variables. The EM algorithm iterates between two distinct stages: it first performs the expectation step (E-step), in which the current values of the model parameters are used to evaluate the posterior probabilities of the latent variables; it then applies the maximization step (M-step), in which these posterior probabilities are used to maximize the complete log-likelihood and update the values of the model parameters. These two steps are performed until a convergence criteria is met.

During the E-step the posterior distribution of the latent variables, known as the responsibilities, is computed as

$$\begin{aligned}\gamma(z_{i,k}) &= \mathbb{E}_{\mathbf{Z}}\left\{z_{i,k}|\boldsymbol{\theta}\right\} = p(z_{i,k}|\mathbf{x}_i, \mathbf{y}_i, \boldsymbol{\theta}) = \\ &= \frac{\pi_k \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\mathcal{N}(y_i|\mathbf{w}_k^\top \mathbf{x}_i, \boldsymbol{\beta}_k^{-1})}{\sum_{k'=1}^{K}\pi_{k'}\mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_{k'}, \boldsymbol{\Sigma}_{k'})\mathcal{N}(y_i|\mathbf{w}_{k'}^\top \mathbf{x}_i, \boldsymbol{\beta}_{k'}^{-1})}.\end{aligned} \tag{12}$$

Note that in this model the responsibilities depend on both $p(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ and $p(\mathbf{y}|\mathbf{w}^\top \mathbf{X}, \beta^{-1})$. Therefore the clustering process is informed by both the *input space* (the input variables contained in the observation data matrix, $\mathbf{X}$) and the *output space* (the output variables contained in the labels, $\mathbf{y}$).

During the M-step we update the model parameters by maximizing the expected value of the complete log-likelihood under the posterior of the latent variables. That is,

$$\boldsymbol{\theta}^{\text{new}} = \arg\max_{\boldsymbol{\theta}} \mathbb{E}_{\mathbf{Z}}\left\{\ln p(\mathbf{Z}, \mathbf{X}, \mathbf{y}|\boldsymbol{\theta})\right\}, \tag{13}$$

where

$$\begin{aligned}&\mathbb{E}_{\mathbf{Z}}\left\{\ln p(\mathbf{Z},\mathbf{X},\mathbf{y}|\boldsymbol{\theta})\right\} = \\ &= \sum_{i=1}^{N}\sum_{k=1}^{K}\gamma(z_{i,k})\left[\ln\pi_k + \ln\mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k) + \ln\mathcal{N}(y_i|\mathbf{w}_k^\top\mathbf{x}_i,\boldsymbol{\beta}_k^{-1})\right].\end{aligned} \tag{14}$$

Thus, the derivatives of (14) with respect to each parameter give us all the necessary update rules.

The cluster weight update rule becomes

$$\pi_k = \frac{1}{N}\sum_{i=1}^{N}\gamma(z_{n,k}), \tag{15}$$

considering the constraints $0 \leq \pi_k \leq 1$ and $\sum_{\forall k}\pi_k = 1$.

The cluster mean and cluster covariance matrix update rules are

$$\boldsymbol{\mu}_k = \frac{1}{N_k}\sum_{i=1}^{N}\gamma(z_{i,k})\mathbf{x}_i \tag{16}$$

and

$$\boldsymbol{\Sigma}_k = \frac{1}{N_k}\sum_{i=1}^{N}\gamma(z_{i,k})(\mathbf{x}_i\mathbf{x}_i^\top - \boldsymbol{\mu}_k\boldsymbol{\mu}_k^\top) \tag{17}$$

respectively, where N_k is the number of members belonging to component k, $N_k = \sum_{\forall i}\gamma(z_{i,k})$.

Whereas the update rules for the regression weights and noise precision become

$$\mathbf{w}_k = (\mathbf{X}^\top\boldsymbol{\Gamma}_k\mathbf{X})^{-1}\mathbf{X}^\top\boldsymbol{\Gamma}_k\mathbf{y} \tag{18}$$

and

$$\boldsymbol{\beta}_k^{-1} = \frac{1}{N\pi_k}\sum_{i=1}^{N}\gamma(z_{i,k})\left(y_i - \mathbf{w}_k^\top\mathbf{x}_i\right)^2 \tag{19}$$

respectively, where $\boldsymbol{\Gamma}_k$ is defined as $\boldsymbol{\Gamma}_k = diag\big(\{\gamma(z_{1,k}),\gamma(z_{2,k}),\ldots,\gamma(z_{N,k})\}\big)$.

2.3. Predictive distribution

The predictive distribution enables us to obtain an estimation of the output, $\hat{\mathbf{y}}^\star$, given a new test observation $\mathbf{x}^\star$. In this case, the output of the k-th regressor is given by $y_k^\star = \mathbf{w}_k^\top\mathbf{x}^\star + \epsilon_k$. Therefore, the probability

distribution of $\mathbf{y}^\star$ given the test observation together with the training data and the inferred model parameters is given by

$$p(y^\star|\mathbf{x}^\star, \mathbf{X}, \mathbf{y}, \boldsymbol{\theta}) = \sum_{k=1}^{K} p(\mathbf{z}_k^\star|\mathbf{x}^\star, \boldsymbol{\theta}) p(y^\star|z_k^\star = 1, \mathbf{x}^\star, \mathbf{X}, y, \boldsymbol{\theta}), \tag{20}$$

where

$$p(\mathbf{z}_k^\star|\mathbf{x}^\star, \boldsymbol{\theta}) = \frac{p(z_k^\star, \mathbf{x}^\star|\boldsymbol{\theta})}{\sum_{k'=1}^{K} p(z_{k'}^\star, \mathbf{x}^\star|\boldsymbol{\theta})} = \frac{\pi_k \mathcal{N}(\mathbf{x}^\star|\mu_k, \Sigma_k)}{\sum_{k'=1}^{K} \pi_{k'} \mathcal{N}(\mathbf{x}^\star|\mu_{k'}, \Sigma_{k'})} \tag{21}$$

and takes a similar role to that of the responsibilities, γ_k, in (12). The difference lies in the fact that here we don't have access to the real value of the target and therefore we can't incorporate this information to the cluster assignment.

Now, since

$$p(y_k^\star|z_k^\star, \mathbf{x}^\star, \mathbf{X}, \mathbf{y}, \boldsymbol{\theta}) = \mathcal{N}(y_k^\star|\mathbf{w}_k^\top \mathbf{x}^\star, \boldsymbol{\beta}_k^{-1}), \tag{22}$$

we have that

$$p(y^\star|\mathbf{x}^\star, \mathbf{X}, \mathbf{y}, \boldsymbol{\theta}) = \sum_{k=1}^{K} p(\mathbf{z}_k^\star|\mathbf{x}^\star, \boldsymbol{\theta}) \mathcal{N}(y_k^\star|\mathbf{w}_k^\top \mathbf{x}^\star, \boldsymbol{\beta}_k^{-1}) \tag{23}$$

We can now obtain an estimation of $y^\star$ as the expected value of $p(y^\star|\mathbf{x}^\star, \mathbf{X}, \mathbf{y}, \boldsymbol{\theta})$

$$\begin{aligned} \hat{y}_{MSE}^\star &= \mathbb{E}\{y^\star|\mathbf{x}^\star, \mathbf{X}, \mathbf{y}, \boldsymbol{\theta}\} \\ &= \sum_{k=1}^{K} p(\mathbf{z}_k^\star|\mathbf{x}^\star, \boldsymbol{\theta}) \mathbb{E}\{y_k^\star|\mathbf{x}^\star, \mathbf{X}, \mathbf{y}, \boldsymbol{\theta}\} \\ &= \sum_{k=1}^{K} p(\mathbf{z}_k^\star|\mathbf{x}^\star, \boldsymbol{\theta}) \mathbf{w}_k^\top \mathbf{x}^\star \end{aligned} \tag{24}$$

This estimation is essentially the sum of the outputs of the K regressors weighted by $p(\mathbf{z}_k^\star|\mathbf{x}^\star, \boldsymbol{\theta})$, which translates to the minimum square error (MSE) estimator. The maximum a posteriori (MAP) estimator, $\hat{y}_{MAP}^\star$, can also be implemented by using the output of the regressor associated to the highest value of $p(\mathbf{z}_k^\star|\mathbf{x}^\star, \boldsymbol{\theta})$ as the estimated value for $y^\star$.

2.4. Model extensions

We now present some minor modifications to the standard version of CWLM that can be easily applied to obtain a more powerful, robust and expressive model.

Regularization term. An L^2 regularization [21] on the regression weights can be included in the model with barely any modifications to the algorithm. This regularization term is introduced to the model in (18), which now becomes

$$\mathbf{w}_k = (\mathbf{X}^\top \mathbf{\Gamma}_k \mathbf{X}\beta_k + \eta \mathbf{I})^{-1} \mathbf{X}^\top \mathbf{\Gamma}_k \beta_k \mathbf{y}. \tag{25}$$

The free parameter η acts simply as a regularization constant, and must be cross-validated to determine its optimal value without over-fitting the training data. This result is akin to assuming a Gaussian prior on the regression weights, where $p(\mathbf{w}|\eta) = \mathcal{N}(\mathbf{w}|0, \eta^{-1}\mathbf{I})$.

Multi-output prediction. The model can be easily extended to perform basic multi-output prediction, provided we assume complete independence of the output variables among themselves. In this case, the target vector $\mathbf{y}$ becomes target matrix $\mathbf{Y} = [\mathbf{y}_1, ..., \mathbf{y}_N]^\top$, where $\mathbf{y}_i \in \Re^T$ contains the T target variables for the i-th observation. Note that in this case the number of responsibilities computed in the E-step grows linearly with T (see (12)), meaning that the number of matrix inversions performed by the regression weight update rule (18) also grows linearly with T. This leaves us with per-task responsibilities

$$\gamma^{(t)}(z_{i,k}^{(t)}) = \frac{p(\mathbf{x}_i, \mathbf{y}_i^{(t)}, z_{i,k}|\boldsymbol{\theta})}{p(\mathbf{x}_i, \mathbf{y}_i^{(t)}|\boldsymbol{\theta})} \tag{26}$$

and per-task regression weight update rule

$$\mathbf{w}_k^{(t)} = (\mathbf{X}^\top \mathbf{\Gamma}_k^{(t)} \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{\Gamma}_k^{(t)} \mathbf{y}^{(t)}. \tag{27}$$

This severe increase in computational complexity can be avoided by averaging the responsibilities over all output targets for each observation,

$$\bar{\gamma}_{i,k} = \frac{1}{T} \sum_{t=1}^{T} \gamma^{(t)}(z_{i,k}^{(t)}) \tag{28}$$

resulting in a single regression weight update rule in which the new averaged responsibilities $\bar{\gamma}_{i,k}$ are arranged in matrix $\bar{\mathbf{\Gamma}}_k = diag\big(\{\bar{\gamma}_{1,k}, \bar{\gamma}_{2,k}, \ldots, \bar{\gamma}_{N,k}\}\big)$.

Multiple input-space views. Another modification of the model allows us to use different characterizations or views of the data for the input mixture of Gaussians and the output linear regressors, provided there is a one to one correspondence between the samples in each view. By defining two distinct input data matrices, $\mathbf{X}^{\dagger} \in \Re^{(N,D\dagger)}$ and $\mathbf{X}^{\ddagger} \in \Re^{(N,D\ddagger)}$, we can easily reformulate the M-Step and E-Step equations by replacing all appearances of $\mathbf{x}_i$ with either $\mathbf{x}_i^{\dagger}$ or $\mathbf{x}_i^{\ddagger}$ appropriately. As we shall see in Section 3, this multi-view approach can prove to be very useful when we wish to exploit different expressions of the input data in the Gaussian mixture and the linear regression portions of the model.

2.5. Model capabilities

To illustrate the capabilities of the CWLM we have generated a simple synthetic data-set with one-dimensional input and output spaces so we can visually analyze the problem. As can be seen in Figure 2, the data is arranged in three distinct groups, each associated to a different cluster and regressor, whose weights, bias terms and observational noise have been set randomly.

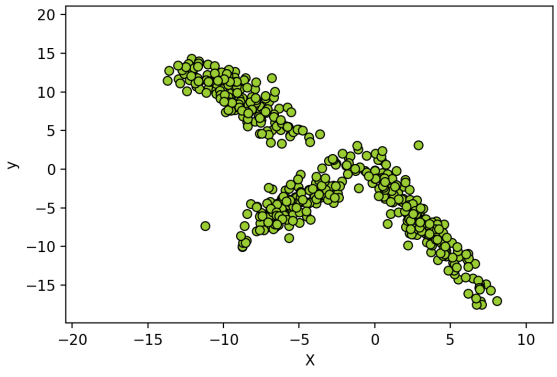


Figure 2: Synthetic data-set designed to illustrate the capabilities of CWLM.

We have fitted three models to this synthetic data-set: a ridge regression model, a K-Means clustering model with 3 components feeding 3 separate ridge regression algorithms, and the CWLM model described in Section 2.

Figure 3 summarizes the results. It's obvious that, while Ridge Regression does its best to fit a linear model to the data, it fails to capture any of its structure. Meanwhile, the K-Means + Ridge Regression approach does a little better at acknowledging the complex nature of the data, but it still

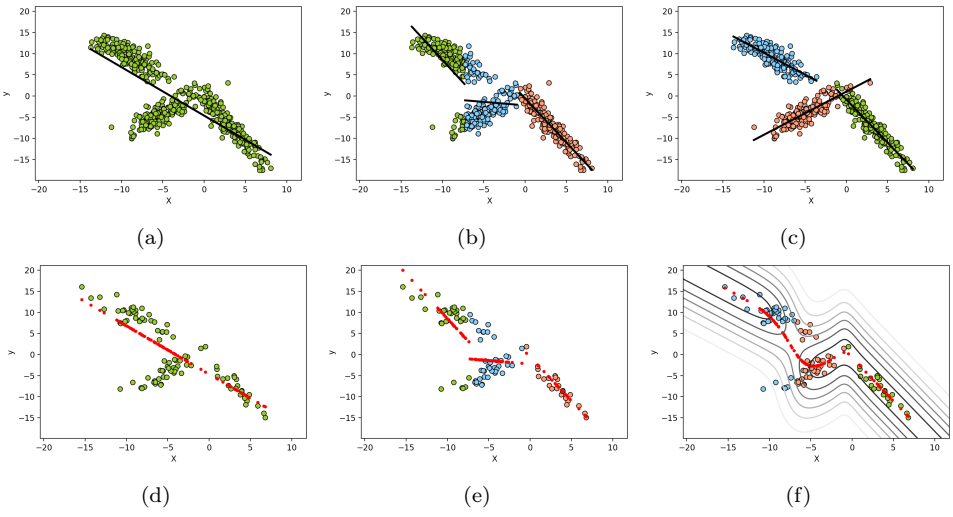


Figure 3: Model training and evaluation on the synthetic data-set. The subfigures on the top row show the results of fitting to the training set a Ridge Regression model (a), a K-means + Ridge Regression model (b) and the CWLM model (c), respectively. The subfigures on the bottom row show the predictions on a test set for the three models. Subfigure (f) also depicts the predictive distribution contour-plot for the CWLM model.

fails to provide an adequate description of the underlying structure due to its inability to use the information contained in the training targets. On the other hand, the CWLM model manages to correctly identify the three clusters and accurately estimate the values of the weights and bias terms. Given the mixed distribution nature of the model, and to the best of our knowledge, an analytical expression for the confidence intervals of the prediction cannot be explicitly obtained. However, the predictive model (see (23)) does provide a full probability distribution for the test target predictions. This can be used as an intuitive indicator of the confidence in the predictions for a given region of the output space. This can be seen in Subfigure 3f, in which predictions that fall within areas delimited by strong contours will offer higher confidence levels. We will illustrate the usefulness of this notion in Section 3.5 in the context of power load forecasting.

3. CWLM applied to power load forecasting

The goal of this section is to evaluate the ability of the CWLM to gain valuable insight into the structure of the data in the context of power load profiling, while achieving competitive forecasting performance scores. For this

purpose we have worked with power load data-sets belonging to two North American Regional Transmission Organizations (RTO).

3.1. Data-set description

ISO New England (ISO) serves the states of Massachusetts, Connecticut, Maine, New Hampshire, Vermont and Rhode Island. It provides an online repository of historical power load data. We have used aggregated data from all its member utilities, spanning from January 2011 to December 2018. The data consists of hourly samples arranged as a time-series. The corresponding hourly measurements of ambient and dew-point temperatures were included as input variables to enhance the overall performance of the models. This meteorological data was obtained from the public NOAA repository.

PJM Interconnection LLC (PJM) serves all or part of Delaware, Illinois, Indiana, Kentucky, Maryland, Michigan, New Jersey, North Carolina, Ohio, Pennsylvania, Tennessee, Virginia, West Virginia, and the District of Columbia. As with ISO New England, PJM provides a public repository that includes data from all their partner utilities, with each utility serving a distinct zone. For this experiment we have selected four different zones: East Kentucky Power Cooperative (EKPC), Dayton Power an Light Company (DAY), Pennsylvania Electric Company (PN) and Commonwealth Edison Company (CE). In this case we used data from 2014 to 2017. Again, we are using hourly samples. Meteorological data was not available in this case.

For both data-sets we decided to focus on a daily structuring of the data to find out if specific daily behaviour patterns can be automatically identified and if these patterns can be exploited to improve the accuracy of our forecast. To achieve this we rearranged the yearly time-series into successive 24 hour time-series.

In the case of the ISO data, the ambient and dew-point temperature data are appended as input variables. Therefore each input sample for the ISO experiment is characterised by 72 variables: 24 successive power load values, 24 successive temperature values and 24 successive dew-point values.

In both experiments, the prediction targets for each daily sample are the 24 hours of the following day. We are therefore using the multi-target version of the model as described in Section 2.4.

Note that, since the main goal of this study is to gain interpretability, we have prioritised data that was relevant to this task. Specifically, we have focused on years that were as close to the test set as possible and that had the most complete daily samples, to ensure that seasonal and daily patterns

were as best represented as possible. This results in smaller data-sets than those used in other studies.

3.2. Baseline forecast models

In order to gauge the performance of the CWLM algorithm described in Section 2, we have established two distinct baseline model families. The first consists of standard models which do not feature any clustering components. The second consists of models that do introduce a clustering stage similar to that of the CWLM.

Beginning with the standard models, we first introduce the Ridge Regression (RR) algorithm [16]. This model was chosen in order to have a linear regression baseline for reference. The second standard model is the Support Vector Machine adapted to regression (SVR) using the epsilon method [12]. A Gaussian kernel was chosen to provide a non-linear baseline model. Since SVR can't perform multitarget regression, 24 SVR models were trained simultaneously while sharing the same parameter values.

As for the clustering baseline models, the first is the K-Means + Ridge Regression (KM-Reg) algorithm, a simple approach to clustered regression in which K-Means is used to separate the training data into K different groups, which are then fed to K independent ridge regression models. Note that this approach allows us to use multiple input-space views in the same manner that is described in 2.4 for CWLM. This model is also used in the synthetic example from section 2. The second of the clustering baselines is the Gaussian mixture model + Ridge Regression (GMM-Reg) algorithm. This is a more nuanced approach to clustered regression in which clustering is achieved by applying a Gaussian Mixture Model (GMM) to the input data and then training K ridge regression models with the full data-set, but using the likelihoods from the GMM model as sample weights. The final output is therefore the sum of the K regression outputs weighted by the GMM likelihoods for each sample and cluster. As is the case for the CWLM and KM-Reg approaches, this model admits multiple input-space views.

3.3. Experimental setup

The ISO data-set was split up into a training partition, containing all data from January 2011 to December 2016; a validation partition, containing all data from the year 2017, used to optimize all model hyperparameters; and a test partition, containing all data from the year 2018, used to compute the performance metrics.

As for the partitioning of the PJM data-set, samples from 2014 to 2016 were used for both training and validation. In this case, the validation set consists of randomly and uniformly selected samples, with a size equal to 20% of the total number of data-points from this time period. The rest of the samples from the 2014-2016 range were used as the training partition. All data from 2017 was used as the test partition.

The optimal values for all the relevant parameters were obtained after a thorough exploration using the validation partition:

- The regularization parameter, λ, for the Ridge Regression, KM-Reg and GMM-Reg models, was explored in the range $\lambda \in [10^{-4}, 10^{2}]$.
- The regularization parameter, η, for the CWLM algorithm, was explored in the range $\eta \in [10^{-4}, 10^{2}]$.
- The Gaussian kernel parameter, γ, for the SVR model was explored in the range $\gamma \in [10^{-4}, 10^{3}]$.
- The number of clusters for the CWLM algorithm as well as the KM-Reg and GMM-Reg models was explored in the range $K \in [2, 40]$.

All data was normalized row-wise so that every time-series lay between the values of 0 and 1. This row-wise normalization was applied in order to retain the shape of the time-series while ensuring an adequate scaling of the data.

For the ISO data-set, we applied both the standard and the multi-view versions of the CWLM, KM-Reg and GMM-Reg algorithms. The standard models received the full input data matrix, containing both the power load and meteorological data. For the multi-view models, we fed the full input-data matrix to the regression stage of the algorithms, whereas the input clustering stage of the algorithms only received the power load portion of the input-data.

Since the PJM data-set isn't augmented with meteorological data, only the standard version of the clustering algorithms was used.

To evaluate the performance of the models described above, we have chosen to use the following metrics, where y_n is the true target value for the n^{th} test sample, $\hat{y}_n$ is the predicted target value for the n^{th} test sample, $\bar{y} = \frac{1}{N_{test}} \sum_{n=1}^{N_{test}} y_n$ is the average of the true test target values and N_{test} is the size of the test set:

- Mean Absolute Percentage Error ($MAPE$), defined as:

$$MAPE = \frac{100\%}{N_{test}} \sum_{n=1}^{N_{test}} \left| \frac{y_n - \hat{y}_n}{y_n} \right| \tag{29}$$

  A lower $MAPE$ implies better performance.

- Root Mean Squared Error ($RMSE$) defined as:

$$RMSE = \left(\frac{1}{N_{test}} \sum_{n=1}^{N_{test}} (y_n - \hat{y}_n)^2 \right)^{1/2} \tag{30}$$

- Mean Absolute Error (MAE), defined as:

$$MAE = \frac{1}{N_{test}} \sum_{n=1}^{N_{test}} |y_n - \hat{y}_n| \tag{31}$$

- Coefficient of Determination (R^2), defined as:

$$R^2 = 1 - \frac{\sum_{n=1}^{N_{test}} (y_n - \hat{y}_n)^2}{\sum_{n=1}^{N_{test}} (y_n - \bar{y})^2} \tag{32}$$

  Where the best possible score is 1.0 and negative scores are possible (because the model can be arbitrarily worse).

3.4. Performance analysis

Tables 1 and 2 summarize the performance results for the ISO and PJM data-sets for all methods under study, with their respective parameters validated according to the procedure described in Section 3.3.

While the differences in performance aren't large, the clustered regression models do gain an edge over Ridge Regression. Overall, the CWLM algorithm comes on top in all metrics for both data-sets, with the multi-view version offering further performance gains in the case of the ISO data-set.

Of interest is the performance difference of the SVR model in both data-sets: in the case of ISO, SVR achieves the worst scores while for PJM it comes very closely tied to CWLM in $MAPE$ and MAE, although CWLM achieves slightly better results in R^2 and $RMSE$. This suggests that the forecast problem for the ISO data is far more linear. In such a scenario, non linear models like the SVR with a Gaussian kernel will tend to overfit. Meanwhile CWLM is able to exploit the linearity of the ISO data to generalize better, while offering the interpretability gains which will be described in Section 3.6.

Model	Clusters	MAPE	R^2	RMSE	MAE
RR	-	4.53	0.81	661.1	135.7
SVR	-	5.14	0.79	654.2	148.5
KM-Reg	4	4.36	0.83	626.8	129.8
GMM-Reg	6	4.37	0.83	621.3	129.5
CWLM	6	4.34	0.84	620.5	128.8
Multi-View KM-Reg	6	4.36	0.83	626.8	129.7
Multi-View GMM-Reg	4	4.43	0.82	634.8	131.7
Multi-View CWLM	4	**4.26**	**0.85**	**619.6**	**127.7**

Table 1: ISO data-set - Test performance for all models.

Models	Clusters	MAPE	R^2	RMSE	MAE
Ridge Regression	-	5.97	0.978	1477.5	236.9
SVR	-	**5.48**	0.987	1392.5	214.9
KM-Reg	10	5.97	0.987	1467.0	233.2
GMM-Reg	7	5.70	0.989	1424.8	221.7
CWLM	13	5.49	**0.990**	**1385.6**	**214.7**

Table 2: PJM data-set - Test performance for all models.

3.5. Advantages of the predictive model

Figure 4 shows the real and forecast power load values for a randomly selected week from the ISO test partition. At the same time, it maps the probability density from the predictive distribution defined in equation (23). Darker regions indicate high probability density, which translates into higher predictive confidence. Lighter regions indicate a more diffuse probability density, therefore suggesting a lower predictive confidence.

The figure shows consistently high predictive confidence during the night and early mornings. This is backed by how closely the forecast curve follows the real curve during these time periods. On the other hand, the probability density disperses during the busier times of the day and the afternoon. Again, this is reflected by a worsening of the quality of the forecast values.

As the predictive function allows us to visually establish an intuition of the confidence in our predictions, our model can be used to determine times

of the day during which a higher volatility in the demand is to be expected, which in turn will influence the resource allocation strategies that need to be put in place.

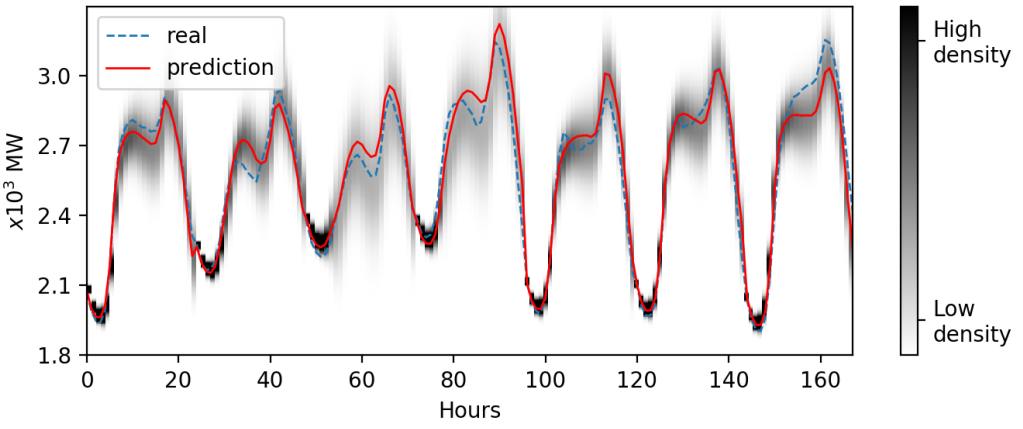


Figure 4: Power Load prediction using the CWLM algorithm: forecast vs real values for a randomly selected week in the ISO test partition. Also shown is the probability density map for the forecast values.

3.6. Interpretability analysis: daily load profiling.

The main goal of this study is to evaluate the interpretability gains for power load profiling provided by the CWLM algorithm. For this purpose, we focus on the nature of the data assigned to each cluster.

Beginning with the ISO data-set, Figures 5 and 6 allow us to evaluate the profiling capability of the CWLM algorithm in the multi-view configuration and of the basic KM-Reg approach[2]. Each figure shows us three different visualizations for each of the four clusters: the first graph represents the cluster centroid together with all its member samples; the second one is a histogram representing the frequency with which each day of the week is represented in the cluster; finally we represent a second histogram showing the frequency with which each month of the year is represented in the cluster.

[2]For simplicity's sake and to save space, we can't include figures for all the models under evaluation. We have chosen to only represent the multi-view CWLM and the KM-Reg algorithm, since they both selected the same number of clusters.

Figure 5 clearly shows that there are strong patterns that have been automatically identified by the multi-view CWLM algorithm. The first and second clusters are dominated by days belonging to the colder months of the year, whereas the third and fourth clusters mostly contain warmer days. Furthermore, the first and third clusters are very highly populated by weekend days, whereas cluster two and four are mostly composed of weekdays.

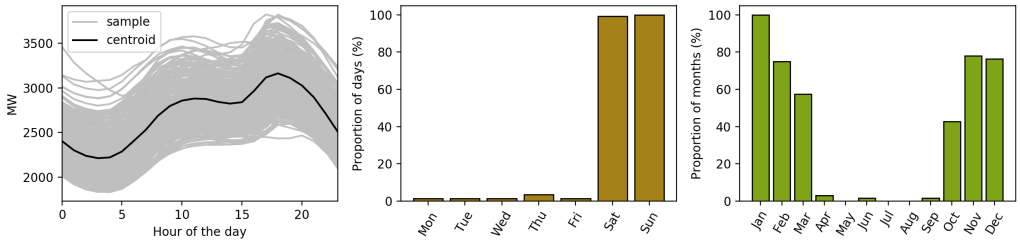
On the other hand, Figure 6 shows that the nature of the clusters selected by KM-Reg is far less clear cut. Some seasonal and daily patterns can be identified in these graphs, but we argue that the overall insight is not as "sharp" as that of the multi-view CWLM, which has very explicitly exploited seasonal and daily patterns, identifying interesting behavioural profiles in the data.

An analysis of the results obtained by the CWLM for the PJM data-set shows that most clusters are associated to specific utilities. For instance clusters 3 and 4 concentrate data from EKPC, whereas clusters 2 and 7 are mostly comprised of data from CE. We have therefore split our analysis between Figures 7, 8 and 9, each corresponding to a utility-specific set of clusters. Each figure features three subfigures, (a) to (c): the first shows the number of members for a given utility in each cluster, justifying the utility profile for each cluster; the second reflects the proportion of weekdays and weekends per cluster; the third subfigure shows how represented each month of the year is in each cluster. From these figures we can see that:

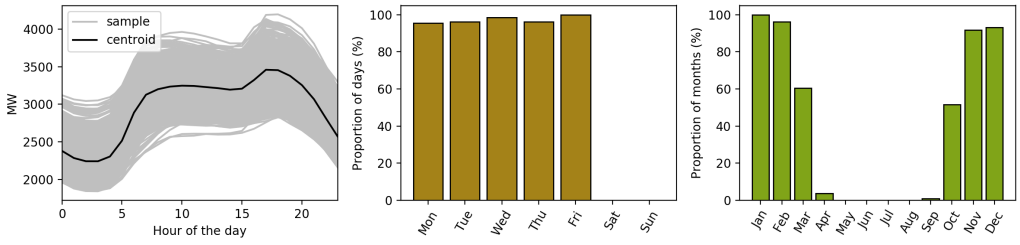
- Clusters associated to EKPC (Figure 7) don't seem to distinguish between weekends and weekdays. They do however show a clear seasonal dependence: cluster 3 is composed mostly of spring and autumn months, Cluster 4 is dominated by summer months and clusters 12 and 13 show a strong presence of winter months.

- In the case of clusters associated with CE (Figure 8), we do find a strong distinction between weekdays and weekends as well as a seasonal component. For instance, cluster 2 is dominated by the warmer months and, as is to be expected due to the presence of summer vacations, weekends and weekdays appear to be mixed. Clusters 7 and 8 clearly show weekday behaviours for the rest of the year. Finally, cluster 9 models weekends throughout the year.

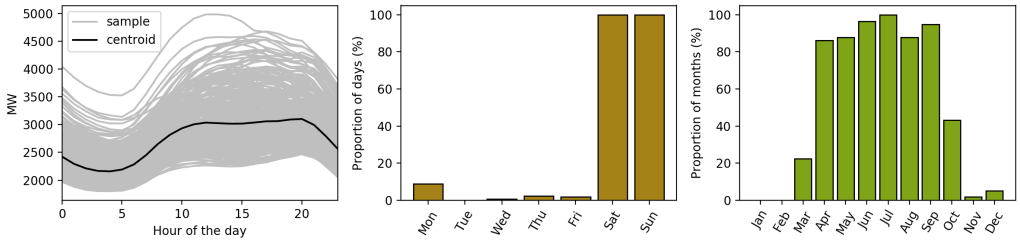
- Interestingly, Figure 9 shows that utilities PN and DAY tend to share similar behaviours and therefore appear grouped in the same clusters.



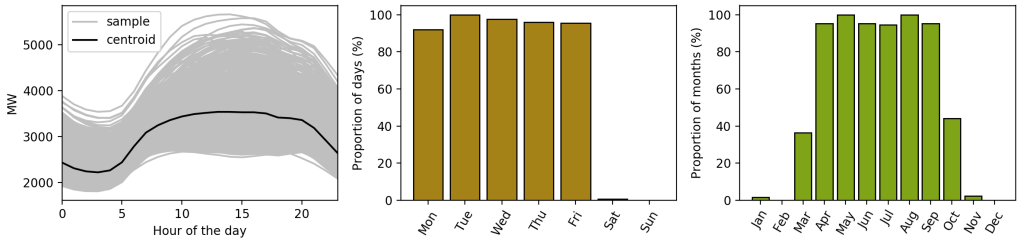
(a) Cluster 1: Cold Months, Weekends.



(b) Cluster 2: Cold Months, Weekdays.

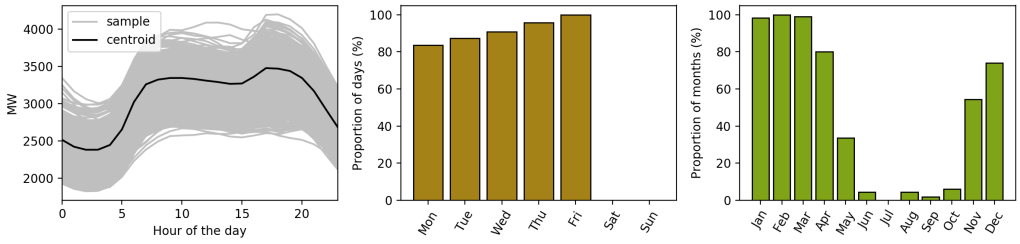


(c) Cluster 3: Warm Months, Weekends.

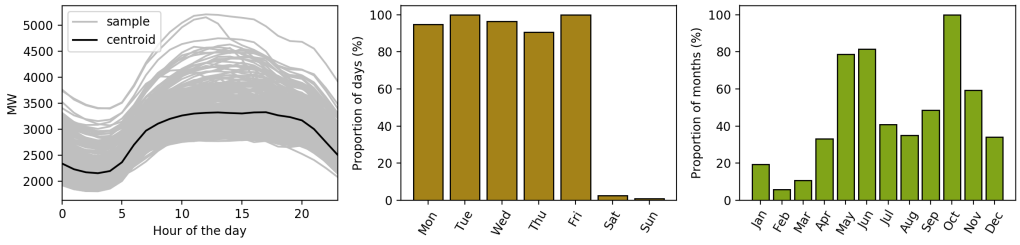


(d) Cluster 4: Warm Months, Weekdays.

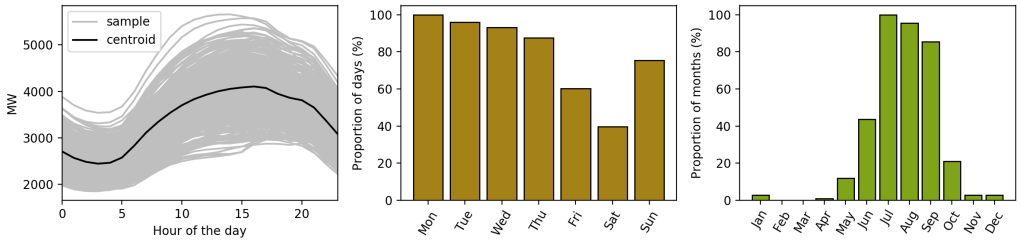
Figure 5: Multi-view CWLM profile visualization for the ISO New England data-set.



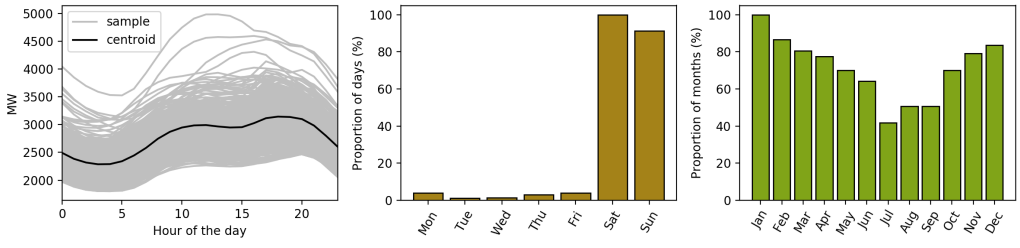
(a) Cluster 1.



(b) Cluster 2.

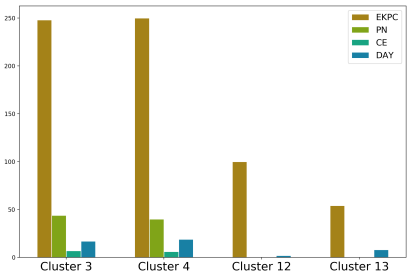


(c) Cluster 3.

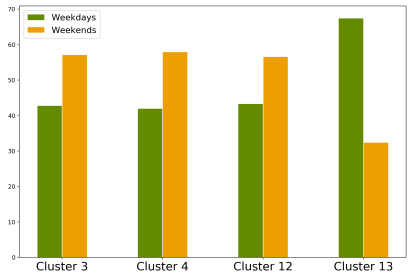


(d) Cluster 4.

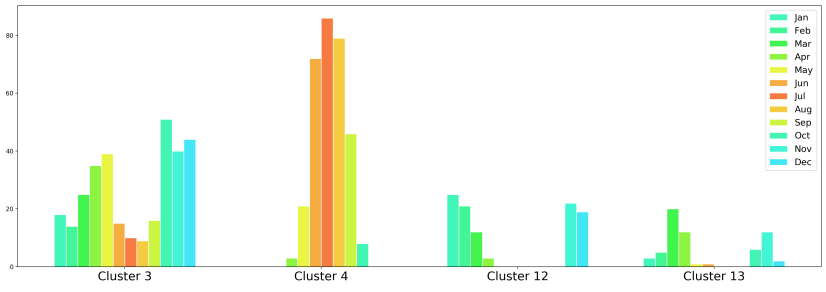
Figure 6: K-Means + Ridge Regression profile visualization for the ISO New England data-set.



(a) Areas per cluster.



(b) Weekdays vs. weekends per cluster.

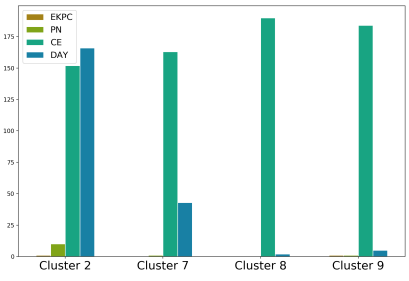


(c) Months per cluster.

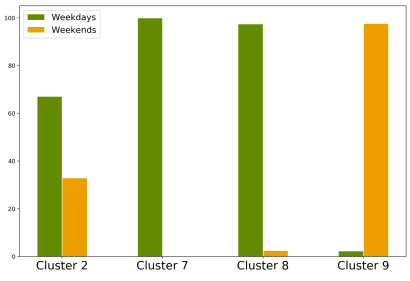
Figure 7: CWLM model visualization for the PJM data-set - EKPC.

Going into more detail, clusters 1 and 11 again show hot summer months and spring-autumn months, respectively. Cluster 2 also models hotter months and cluster 5 focuses on summer and autumn weekends. Finally, clusters 6 and 10 are associated to cold months.

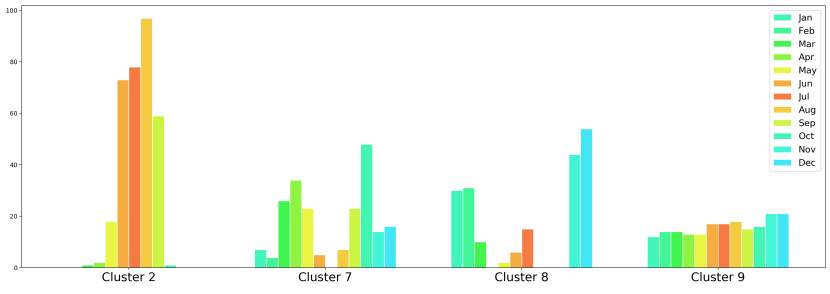
Visualizations for the PJM data-set results of the GMM-Reg and the KM-Reg models can be seen in Figure 10 and Figure 11, respectively. These figures represent the number of samples from each utility assigned to each cluster. Here we can see that GMM-Reg tends to also discriminate utilities quite well, particularly in the case of EKPC and PN. However, its lack of predictive performance (Table 2) suggests that these clusters aren't as relevant as those obtained by CWLM. Finally, KM-Reg is incapable of even discriminating utilities in a meaningful way, as well as having the worst performance out of all the clustering models.



(a) Areas per cluster.

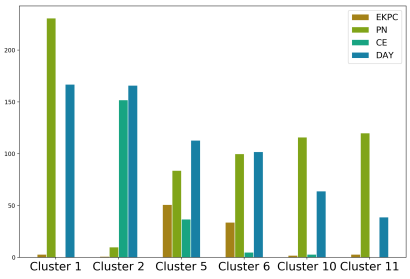


(b) Weekdays vs. weekends per cluster.

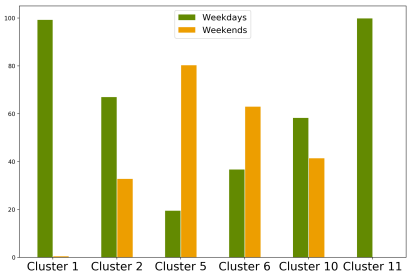


(c) Months per cluster.

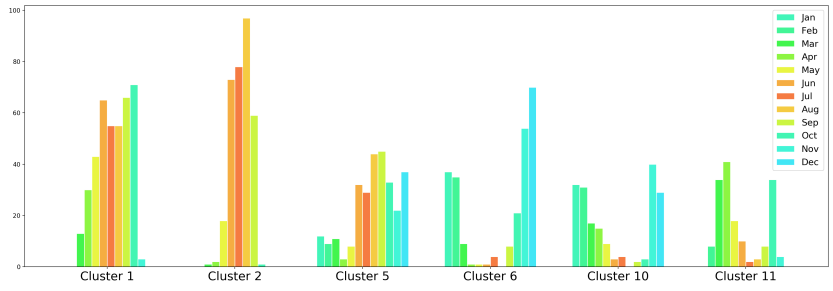
Figure 8: CWLM model visualization for the PJM data-set - CE.



(a) Areas per cluster.



(b) Weekdays vs. weekends per cluster.



(c) Months per cluster.

Figure 9: CWLM model visualization for the PJM data-set - PN DAY.

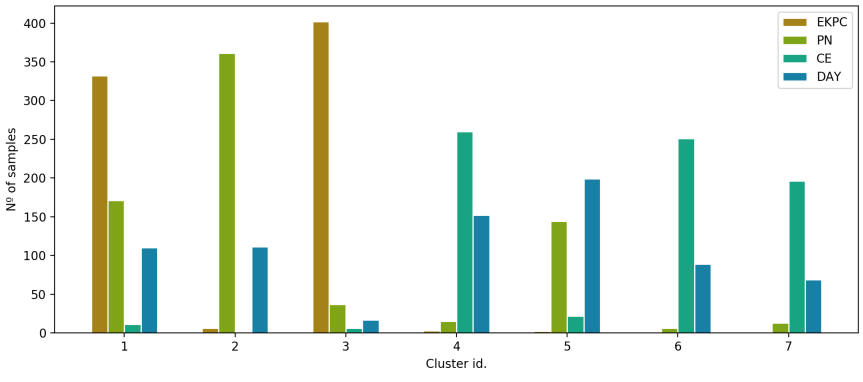


Figure 10: GMM-Reg model visualization for the PJM data-set.

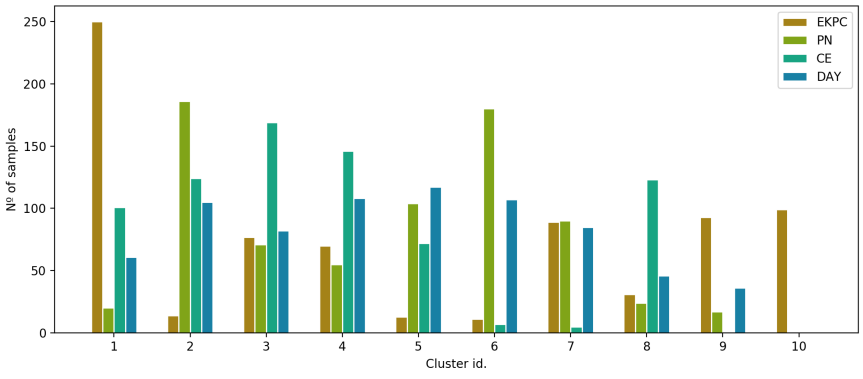


Figure 11: KM-Reg model visualization for the PJM data-set.

4. Conclusions

In this paper we propose a novel theoretical framework leading to a probabilistic model that features simultaneous data clustering and regression. The clustering of data samples is informed by the regression process, which enables the model to achieve a better characterisation of the underlying nature of the data. Therefore, we obtain better, more informative clusters while maintaining or improving predictive performance. We propose that this model can improve problem interpretability in the context of day-ahead electric power-load forecasting, by generating useful and insightful daily load profiles.

At the same time, the model presents a probabilistic predictive function that is capable of providing an intuition for the forecasting confidence, which in turn can be used to improve the interpretation of the predicted power-load values. We suggest that this could be of great use in the context of power-grid management and efficiency.

Experimental results in the context of power load forecasting applied to data from two major Regional Transmission Organizations confirm the usefulness of our model in terms of interpretability, as it is shown to generate insightful load profiles, while obtaining competitive forecasting performance when compared to other prediction models. In both data-bases, the automatically generated profiles reflect the relevance of regional and seasonal patterns, as well as the influence of weekdays and weekends: for the first data-set, very clear seasonal and daily patterns were obtained, with four strong clusters that

segregated data into weekday and weekends during warm months and cold months; for the second data-set, strong regional separation was automatically achieved, with dedicated clusters for specific utilities at different times of the year. Meanwhile the predictive function points to very high confidence during the early hours of the morning, with said confidence dropping during the busiest hours of the day.

Two key improvements to CWLM can be introduced as future work. First, a fully Bayesian approach can be formulated in which the regression weights become a new latent variable with their own prior distribution. This can be solved using the Variational Inference approach known as Mean Field Approximation, which will result in a very similar formulation to the one presented in this work. The main advantage is that the need to validate the regularization term, η, would disappear, as it would become a part of the iterative model optimization algorithm. The second improvement we propose is the implementation of a complete multi-target model in which the possible correlations between the output variables are taken into account, while maintaining the same integration with the clustering process as the model presented in this paper.

Acknowledgements

This work is partially supported by the National Science Foundation EPSCoR Cooperative Agreement OIA-1757207 and the Spanish MINECO grants TEC2014-52289R and TEC2017-83838-R

References

[1] Joana M. Abreu, Francisco Cámara Pereira, and Paulo Ferrão. Using pattern recognition to identify habitual behavior in residential electricity consumption. *Energy and Buildings*, 49:479–487, 2012.

[2] U.S. Energy Information Administration. Renewable sources, 2019.

[3] Christopher M. Bishop. *Pattern recognition and machine learning*. Springer, 2017.

[4] George E. P. Box, Gwilym M. Jenkins, Gregory C. Reinsel, and Greta M. Ljung. *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[5] Christopher J. C. Burges, Bernhard Scholkopf, and Alexander J. Smola. *Advances in kernel methods: support vector learning*. MIT press Cambridge, MA, USA:, 1999.

[6] Gustavo Camps-Valls, Luis Gómez-Chova, Jordi Muñoz-Marí, José Luis Rojo-Álvarez, and Manel Martínez-Ramón. Kernel-based framework for multitemporal and multisource remote sensing data classification and change detection. *IEEE Transactions on Geoscience and Remote Sensing*, 46(6):1822–1835, 2008.

[7] Enrico Carpaneto, Gianfranco Chicco, Roberto Napoli, and Mircea Scutariu. Customer classification by means of harmonic representation of distinguishing features. In *2003 IEEE Bologna Power Tech. Conference Proceedings*, volume 3. IEEE, 2003.

[8] European Commission. Renewable energy statistics, 2019.

[9] Tamás Csoknyai, Jeremy Legardeur, Audrey Abi Akle, and Miklós Horváth. Analysis of energy consumption profiles in residential buildings and impact assessment of a serious game on occupants' behavior. *Energy and Buildings*, 196:1–20, 2019.

[10] Longquan Diao, Yongjun Sun, Zejun Chen, and Jiayu Chen. Modeling energy consumption in residential buildings: A bottom-up analysis based on occupant behavior pattern clustering and stochastic simulation. *Energy and Buildings*, 147:47–66, 2017.

[11] Xishuang Dong, Lijun Qian, and Lei Huang. Short-term load forecasting in smart grid: A combined CNN and K-means clustering approach. In *2017 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pages 119–125. IEEE, 2017.

[12] Harris Drucker, Christopher J. C. Burges, Linda Kaufman, Alex J. Smola, and Vladimir Vapnik. Support vector regression machines. In *Advances in Neural Information Processing Systems*, pages 155–161, 1997.

[13] 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 Systems*, 20 (3):1622–1630, Aug 2005.

[14] Marcelo Espinoza, Caroline Joye, Ronnie Belmans, and Bart De Moor. Short-term load forecasting, profile identification, and customer segmentation: a methodology based on periodic time series. *IEEE Transactions on Power Systems*, 20(3):1622–1630, 2005.

[15] Fateme Fahiman, Sarah M. Erfani, Sutharshan Rajasegarar, Marimuthu Palaniswami, and Christopher Leckie. Improving load forecasting based on deep learning and K-shape clustering. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 4134–4141. IEEE, 2017.

[16] Gene H. Golub, Per Christian Hansen, and Dianne P. O'Leary. Tikhonov regularization and total least squares. *SIAM Journal on Matrix Analysis and Applications*, 21(1):185–194, 1999.

[17] George Gross and Francisco D. Galiana. Short-term load forecasting. *Proceedings of the IEEE*, 75(12):1558–1573, 1987.

[18] Simon Haykin. *Adaptive filter theory*. Prentice-Hall, Inc., 1996.

[19] Schalk W. Heunis and Ron Herman. A probabilistic model for residential consumer loads. *IEEE Transactions on Power Systems*, 17(3):621–625, 2002.

[20] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[21] Arthur E. Hoerl and Robert W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

[22] José Antonio Jardini, Carlos M. V. Tahan, Marcos R. Gouvea, Se Un Ahn, and F. M. Figueiredo. Daily load profiles for residential, commercial and industrial low voltage consumers. *IEEE Transactions on Power Delivery*, 15(1):375–380, 2000.

[23] Weicong Kong, Zhao Yang Dong, Youwei Jia, David J. Hill, Yan Xu, and Yuan Zhang. Short-term residential load forecasting based on LSTM recurrent neural network. *IEEE Transactions on Smart Grid*, 10(1): 841–851, 2017.

[24] Kehua Li, Zhenjun Ma, Duane Robinson, and Jun Ma. Identification of typical building daily electricity usage profiles using Gaussian mixture model-based clustering and hierarchical clustering. *Applied Energy*, 231: 331–342, 2018.

[25] Ling-Ling Li, Jin Sun, Ching-Hsin Wang, Ya-Tong Zhou, and Kuo-Ping Lin. Enhanced Gaussian process mixture model for short-term electric load forecasting. *Information Sciences*, 477:386–398, 2019.

[26] Yuan-cheng Li, Ting-jian Fang, and Er-keng Yu. Study of support vector machines for short-term load forecasting. *Proceedings of the CSEE*, 23 (6):55–59, 2003.

[27] Huaiwei Liao and Dagmar Niebur. Load profile estimation in electric transmission networks using independent component analysis. *IEEE Transactions on Power Systems*, 18(2):707–715, 2003.

[28] Andrea Mammoli, Matthew Robinson, Victor Ayon, Manel Martínez-Ramón, Chien-Fei Chen, and Joana M. Abreu. A behavior-centered framework for real-time control and load-shedding using aggregated residential energy resources in distribution microgrids. *Energy and Buildings*, 198:275–290, 2019.

[29] Manel Martínez-Ramón, José Luis Rojo-Álvarez, Gustau Camps-Valls, Jordi Muñoz-Marí, Ángel Navia-Vázquez, Emilio Soria-Olivas, and Aníbal R. Figueiras-Vidal. Support vector machines for nonlinear kernel ARMA system identification. *IEEE Transactions on Neural Networks*, 17(6):1617–1622, 2006.

[30] Hiroyuki Mori and Masatarou Ohmi. Probabilistic short-term load forecasting with Gaussian processes. In *Proceedings of the 13th International Conference on, Intelligent Systems Application to Power Systems*. IEEE, 2005.

[31] Kevin P. Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[32] Hongzhan Nie, Guohui Liu, Xiaoman Liu, and Yong Wang. Hybrid of ARIMA and SVMs for short-term load forecasting. *Energy Procedia*, 16: 1455–1460, 2012.

[33] E. Pan, H. Li, L. Song, and Z. Han. Kernel-based non-parametric clustering for load profiling of big smart meter data. In *2015 IEEE Wireless Communications and Networking Conference (WCNC)*, pages 2251–2255, March 2015.

[34] Alex D. Papalexopoulos and Timothy C. Hesterberg. A regression-based approach to short-term system load forecasting. *IEEE Transactions on Power Systems*, 5(4):1535–1547, 1990.

[35] Amin Rajabi, Mohsen Eskandari, Mojtaba Jabbari Ghadi, Li Li, Jiangfeng Zhang, and Pierluigi Siano. A comparative study of clustering techniques for electrical load pattern segmentation. *Renewable and Sustainable Energy Reviews*, Vol. 120:109628, 2020.

[36] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006.

[37] Joshua D. Rhodes, Wesley J. Cole, Charles R. Upshaw, Thomas F. Edgar, and Michael E. Webber. Clustering analysis of residential electricity demand profiles. *Applied Energy*, 135:461–471, 2014.

[38] José Luis Rojo-Álvarez, Manel Martínez-Ramón, Mario de Prado-Cumplido, Antonio Artés-Rodríguez, and Aníbal R. Figueiras-Vidal. Support vector method for robust ARMA system identification. *IEEE Transactions on Signal Processing*, 52(1):155–164, 2004.

[39] José Luis Rojo-Álvarez, Manel Martínez-Ramón, Jordi Muñoz Marí, and Gustavo Camps-Valls. *Digital signal processing with Kernel methods*. Wiley Online Library, 2018.

[40] John Shawe-Taylor and Nello Cristianini. *Kernel methods for pattern analysis*. Cambridge university press, 2004.

[41] Dipti Srinivasan, Swee Sien Tan, C. S. Cheng, and Eng Kiat Chan. Parallel neural network-fuzzy expert system strategy for short-term load forecasting: System implementation and performance evaluation. *IEEE Transactions on Power Systems*, 14(3):1100–1106, 1999.

[42] Harry G. Stoll and Leonard J. Garver. *Least-cost electric utility planning*. J. Wiley, 1989.

[43] Volker Tresp. Mixtures of Gaussian processes. In *Advances in Neural Information Processing Systems*, pages 654–660, 2001.

[44] H. L. Willis, A. E. Schauer, J. E. D. Northcote-green, and T. D. Vismor. Forecasting distribution system loads using curve shape clustering. *IEEE Transactions on Power Apparatus and Systems*, PAS-102(4):893–901, April 1983.

[45] H. Lee Willis. *Power distribution planning reference book*. CRC press, 2004.

[46] Selin Yilmaz, Jonathan Chambers, and Martin Kumar Patel. Comparison of clustering approaches for domestic electricity load profile characterisation-implications for demand side management. *Energy*, 180: 665–677, 2019.

[47] Hyeonjoong Yoo and Russell L. Pimmel. Short term load forecasting using a self-supervised adaptive neural network. *IEEE Transactions on Power Systems*, 14(2):779–784, 1999.