Short-term scenario-based probabilistic load forecasting

A data-driven approach

Khoshrou, Abdolrahman; Pauwels, Eric J.

DOI
10.1016/j.apenergy.2019.01.155

Publication date
2019

Document Version
Accepted author manuscript

Published in
Applied Energy

Short-Term Scenario-Based Probabilistic Load Forecasting: A Data-Driven Approach

Abdolrahman Khoshrou[a,b,1,*], Eric J. Pauwels[a,2]

[a] *Centrum Wiskunde & Informatica, Science Park 123, 1098 XG, Amsterdam, The Netherlands*
[b] *Department of Mathematics and Computer Science, Delft University of Technology, The Netherlands*

Abstract

Scenario-based probabilistic forecasting models have been explored extensively in the literature in recent years. The performance of such models evidently depends to a large extent on how different input (temperature) scenarios are being generated. This paper proposes a generic framework for probabilistic load forecasting using an ensemble of regression trees. A major distinction of the current work is in using matrices as an alternative representation for quasi-periodic time series data. The singular value decomposition (SVD) technique is then used herein to generate temperature scenarios in a robust and timely manner. The strength of our proposed method lies in its simplicity and robustness, in terms of the training window size, with no need for subsetting or thresholding to generate temperature scenarios. Furthermore, to systematically account for the non-linear interactions between different variables, a new set of features is defined: the first and second derivatives of the predictors. The empirical case studies performed on the data from the load forecasting track of the Global Energy Forecasting Competition 2014 (GEFCom2014-L) show that the proposed method outperforms the top two scenario-based models with a similar set-up.

Keywords: Time-series analysis, Energy forecasting, Probabilistic forecasting, Time-varying effects, Singular value decomposition

1. Introduction

In the energy transition era (transition from conventional to non-conventional energy sources), the balancing of the power grid has become more challenging. It is mostly due to the inherently intermittent nature of renewable energy sources (RES), on the one hand, and shortcomings in bulk energy storage systems, on the other. The studies on probabilistic energy production and demand forecast have hence gained momentum, as they are highly valuable from both a technical and an economic point of view [18].

Generally speaking, load forecasting problems can be contemplated from two main perspectives: 1) time horizon; and 2) type of forecasting (point vs. probabilistic forecasting). The time-interval of interest, which can vary from the next few seconds or minutes to a couple of months or even years, categorizes the load forecasting problems into four groups [18]. The future of a grid and its expansion in long run is studied in the context of long term load forecasting (LTLF). Moreover, balance sheet calculations, risk management, purchasing energy and price planning purposes are most relevant from a few weeks up to a few months in advance; that is where medium term load forecasting (MTLF) techniques come into play. Short term load forecasting (STLF), as a dominant factor in electricity dispatching, scheduling and unit commitment, is mostly concerned with the load estimations for a few hours up to a few days ahead. Finally, very short term load forecasting (VSTLF) approaches are aimed to mitigate the possible mismatches between supply and demand, by generating highly accurate load prognoses for the coming half an hour or less. Although point (single-value) load forecasting methodologies have been implemented since the early days of

*Corresponding author: *a.khoshrou@cwi.nl*
[1] Abdolrahman Khoshrou is with the Intelligent and Autonomous Systems Group at CWI; he is also a guest PhD student at TU Delft.
[2] Eric Pauwels is a senior researcher and the leader of the Intelligent and Autonomous Systems Group at CWI.

modern grids, probabilistic load forecasting (PLF) studies have gained prominence in recent years [18]. Moreover, with the growing integration of weather-dependent and intermittent RES, different lines of research on the energy systems data have emerged. These include issues such as data-driven outlier detection, pre-processing, incorporating the dependency between different attributes, and so on demand further exploration.

Load forecasting methodologies can typically be classified into two groups: statistical and machine learning (ML) based techniques. The major argument in favor of the statistical approaches is the interpretability of their results; noteworthy is that some expert knowledge is usually needed to (partially) guide the learning process. On the other hand, ML based approaches are more independent in the sense that the user interventions are mostly limited to hyper-parameter tuning. Such models are generally more robust, easy to reconfigure, user-friendly and successful in addressing the non-linearity in the data. The major drawback of ML based methods, however, is that being a black box, it is often not clear how different attributes have contributed to the final results. A number of single-value and probabilistic load forecasting models using two different statistical and ML based approaches are summarized below.

Statistical approaches: Multiple linear regression (MLR) models are among the most fundamental and widely used models for both STLF and LTLF problems. Broadly speaking, such models are mostly aimed to learn a relationship between several explanatory variables and a dependent (target) variable. Typically, a *goodness-of-fit* function is used to estimate the target variable (load) based on other explanatory attributes (such as historical load and temperature data, calendar information or some interaction of them). The performance of such models, consequently, are only satisfactory if the dependent variables are well formulated based on explanatory variables. However, the most striking feature of such models is their need for some expert-knowledge to formulate the interaction between different variables - namely the recency effect, as it seems unworkable to incorporate the effects between different variables without some domain expertise. Usually, a large number of lagged temperature data are being used to account for the recency effect (which leads to an increase of the parameter-space). A solid ground for applying MLR analysis to STLF is provided in [34]. To include in part the interactions between the variables, 24 separate family regression models (one model for every hour of the day) have repeatedly been deployed in the literature to generate the day-ahead load prognoses (see e.g., [35]). Nonetheless, an interesting finding in [39] is that using 24 separate models (one for every hour of the day) might not necessarily ensure outperformance of such models over one interaction model for all 24 hours. Another category of regression based models are semi-parametric additive models, which generally are designed to address the nonlinear relationships and serially correlated errors. Auto regressive integrated moving average (ARIMA) models, as a class of ARMA models, are well-suited to capture different standard temporal structures in time series data. A family of such statistical models have been used numerously in the literature for time series forecasting problems, as they offer a baseline regression-based approach to account for the recency effect in the data. A comprehensive discussion on different types of autoregressive models, such as ARMA, ARIMA and ARMAX models is presented in [32]. Exponential smoothing is another capable and effective approach to systematically assimilate the recency effect in the data, by assigning weights to the previous data points inside a certain window [22]. However, the major drawback of most statistical models lies in their need for a lot of hyper-parameter tuning; specific factors such as threshold sets, the choice of lag sets, and also capturing the time-varying structures of the coefficients are crucial in the overall performance of the model. Furthermore, setting up a good fit enlarges the parameter space, which brings about longer computation time and raises the concern of over-fitting. To better accommodate the various non-linearity in the data, a least absolute shrinkage and selection operator (LASSO) estimation algorithm is introduced in [44].

Machine learning approaches : Machine learning (ML) based approaches are in fact a number of more advanced statistical methods for handling more complex regression and classification problems. Support vector machines (SVM), artificial neural networks (ANN), fuzzy regression models, classification and regression trees (CART), and $k-$nearest neighbours are among the most well-recognized ML based techniques. SVM is a powerful method for handling various regression and classification problems by recognizing patterns and constructing nonlinear decision boundaries. A generic model for STLF problem using SVM is developed in [8]; the strength of this work lies in its novel automatic feature selection algorithms and also the use of a particle swarm global optimization based technique to tweak the hyper-parameters. To enhance the performance of such models, different clustering approaches (e.g., based on hour of the day) have been proposed in the literature to group the data first, and apply an SVM model on each subset of data. In [14], an unsupervised self organizing map (SOM) is used for clustering the load profiles, first; an SVM regression model, for each group, is then applied to estimate the daily load profiles. As mentioned before, failing to

incorporate the recency effects in forecasting methodologies can lead to under performance. An SVM type hourly load forecasting model in [10] accounts for the recency effects by including a couple of previous ambient temperature values (several hours) as input variables. Nevertheless, the performance of such models relies on a suitable kernel function and hyper-parameter settings. A new approach for choosing an optimal kernel function for an SVM based model is proposed in [9]. Furthermore, fuzzy logic models have been developed to address some limitations of the linear models such as vague relation between different independent and dependent attributes, shortage in the number of observations, and error distribution verification [11], [23]. Such models provides a means to better incorporate the recency and cross effects in the data, and hence can outperform their corresponding MLR based counterpart models [20]. Over the last few years, artificial neural networks (ANN) have seen an explosion of interest in various types of prediction, classification and control applications across different fields. Typically, ANN do not perform based on modelling the underlying relation between predictors and the target variable; a mapping mechanism is instead in place to assign inputs to a target variable. Different variations of (hybrid) NN-based models for the STLF problems have been proposed [36], [41].

It is a common sense that no individual forecasting model is the best for all data sets. Therefore, it is highly appreciated to combine different forecasts to reduce the overall risk of making poor decisions. In [31], forecast combinations or ensemble models are classified into homogeneous and heterogeneous ensemble methods. In the former method, different forecast series are obtained by varying the hyper-parameters, input data, input features, or output targets for the same algorithm. The latter method, however, combines a number of forecasts with the hope that diversity help improving the results. Some examples of the application of different types of ensemble learning methods in energy forecasting tasks are presented in [4], [12], [27]. S. B. Taieb and R. J. Hyndman propose a robust component-wise gradient boosting model for STLF in [38]. The reported work incorporates different non-parametric effects such as calendar, temperature, lagged demand using an additive framework; it is done by considering a separate model for each hour of the day (24 different models for a day). Furthermore, univariate penalized regression spline functions is used to account for the recency effect in the data. However, caution should be exercised in the application of most exponential smoothing models, as the performance heavily relies on the window-size and hyper-parameter tuning [38]. Scenario-based probabilistic load forecasting models, as a subcategory of homogeneous models, have been exercised extensively, in recent years. Among the various methods, feeding simulated temperature scenarios to a single-value load forecasting model is being commonly accepted by the industry for its simplicity and interpretability. There are mainly three practical and popular methods for generating temperature scenarios, namely fixed-date, shifted-date, and bootstrap approaches. Nevertheless, these methods have mostly been used on an ad-hoc basis without being formally compared or quantitatively evaluated [43]. The performance of such models evidently depends to a great deal on how different temperature scenarios are being generated. Addressing the issues such as outliers, or gradual drifts in the data is essential in developing an accurate model. Another challenge in load forecasting problems (especially, short term) is the incorporation of recency effects of the data using time-varying models. The recency effect refers to the continuous nature of electricity consumption - at any moment it depends on the weather conditions and accordingly load, prior to that [28]. In other words, it mostly reflects the lagging effect associated with the thermal inertia of the buildings and facilities. In the literature, some heuristic, data-oriented lagging window of a couple hours, or subsetting of the data are typical approaches to incorporate the recency effect [28], [39]. The major concern regarding such methods is the generalizability of the results, as, e.g., the window size or the threshold in the subsetting step can affect the performance of the models drastically [44].

We herein propose two scenario-based probabilistic load forecasting models using an ensemble of regression trees. The contribution of this paper is as follow.

- An important distinction of the current work is the use of matrices as an alternative representation of the data. The singular value decomposition (SVD) technique is then used to generate temperature scenarios, in a robust and data-driven manner.
- In the second model, we extend the first one by adding the first and the second derivatives of the non-deterministic attributes (temperature and historical load data). This was done to partially account for the recency effects and interactions among the data. Unlike some family of time-varying models, our proposed approach is systematic, with no need for subsetting or thresholding the data.
- The experiment results show that special enhancements can consequently be obtained using this new set of

features (Section 4).

The rest of this paper is organized as follow. Section 2 provides a brief introduction to the data from the load forecasting track of the Global Energy Forecasting Competition (GEFCom2014-L). Section 3 is devoted to our methodology in this paper. A brief recapitulation of the Gradient Boosting method is presented, first, followed by out proposed models for point forecasting. After an introduction to the singular value decomposition (SVD), we explain how our proposed scenario based load forecasting models works. The proposed method in this paper is, in fact, a marriage between an SVD-based temperature scenario generator and an ensemble of trees (gradient boosting algorithm). The experimental results along with a comparison with the results of a number of benchmark models are presented in Section 4. We conclude this work in Section 5.

2. Data

To make the results of the proposed models replicable and accordingly comparable with the benchmark models, a case study was constructed based on the data from the load forecasting track of the Global Energy Forecasting Competition 2014 (GEFCom2014-L). The participants had been asked to develop a short term probabilistic load forecasting model, with the forecasting horizon of one month. The publicly available data set consists of the hourly temperature values from 25 anonymous weather stations and the aggregated hourly load profiles; for detailed description of data and the competition instructions see [19]. The electricity consumption patterns are subject to a variety of factors, such as meteorological conditions, calendar information, season, working schedules, energy cost and economic activities [30]. In the current work, however, consistent with the requirements in the GEFCom2014-L, only the temperature and the calendar information are considered as the available predictors (besides historic load profiles). Temperature is believed to be a major driving force behind the electricity demand; the non-linear effect of the temperature to the electricity demand is hence at the center of our attention. The left panel of Figure 1 provides an overview of the typical electricity consumption profiles on daily basis. This figure affirms that the consumption patterns differ notably during the weekends from the weekdays. Interestingly enough, on Friday afternoons, the demand profile gets close to the weekends, whereas, during the working hours, it is akin to other working days. Furthermore, the right panel in Figure 1, illustrates the evolution of daily load profiles in the year 2010; this figure was obtained by recasting the time series into a 24×365 matrix, where every column contains 24 hourly values for each daily profile [24]. As expected, in spring and fall, where the temperature is moderate, electricity demand tends to be lower than any other time of the year (winter and summer times). It underscores the fact that electricity demand is driven by climate conditions (e.g., air conditioning usage), and also the lifestyle changes followed by that. This figure also highlights the non-linear relation between load and temperature throughout the year.

In the literature, temperature is arguably the most dominant predictor of the load; however, in and of itself, it is not sufficient, for two main reasons:

- **Diurnal human activities:** As it is seen in the left panel of Figure 1, the typical electricity demand behaviour changes throughout the week. These diurnal human activities are plainly not reflected in the temperature data.

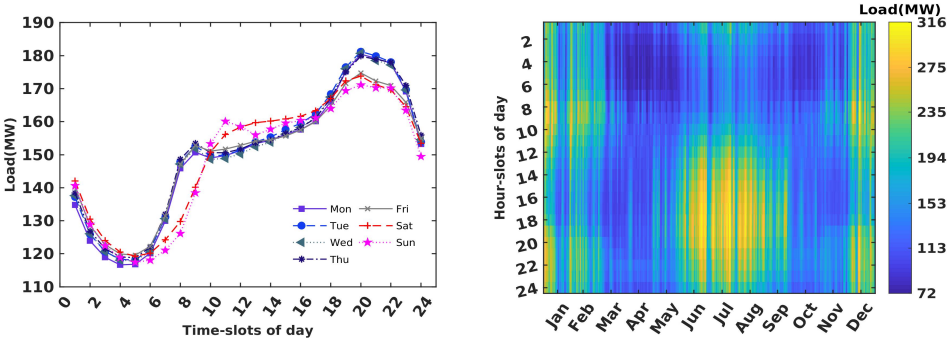

Figure 1: Left: Comparison of typical daily consumption patterns during the week. Right: An overall representation of the evolution of the load profiles throughout a year.

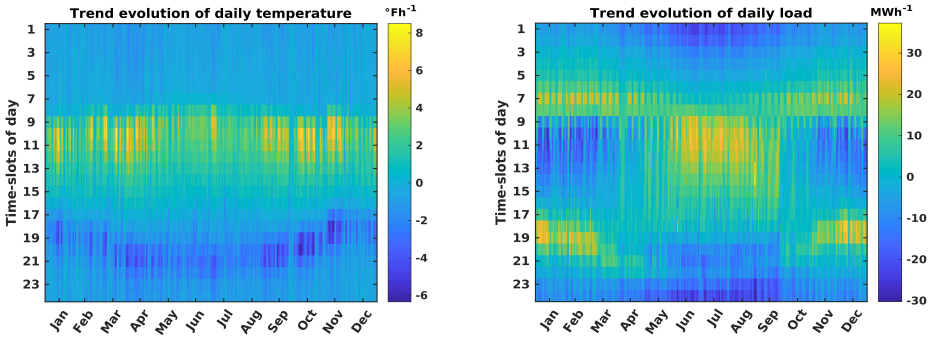

Figure 2: An overview of the evolution of the first derivative of the temperature (Left) and load (Right) profiles.

- **Recency and cross effects:** Even for similar days (weekends or weekdays), the trend of daily load profiles, corresponding to similar temperature data, might not be necessarily alike; the recency and cross effects can play a vital role. For instance, the rise of temperature in early spring might not necessarily lead to high electricity consumption, in comparison with summer times, as people might appreciate the rise in outside temperature after a cold winter. This, of course, can deviate across different seasons. Figure 2 illustrates an overview of the trend (first derivative) changes of daily temperature and load profiles in 2010. These figures were obtained by taking the first derivatives of daily temperature and load matrices. It is seen that, e.g., in early spring and summer time, with the rise of temperature, afternoon peak profiles start to disappear. Although, the overall relationship between load and temperature is clear; it is, however, non-trivial how to robustly address the non-linear effect between temperature and load profiles. Experiment results in Section 4 affirm that including the 1st and 2nd derivatives of the daily profiles can indeed enhance the performance of the forecasting model.

The following section provides a brief to the ensemble of regression trees, followed by our proposed methodology for probabilistic STLF problem.

3. Methodology

In the present work, we opt to use an ensemble of regression trees (Gradient Boosting method) to predict day-ahead load prognoses, with forecasting horizon of one month, given hourly temperature profiles, historical (or estimated) load profiles, and calendar information. A brief recapitulation of an ensemble of regression trees is first provided in below.

3.1. Ensemble of regression trees

The use of "ensemble learning" methods in various classification and regression problems has taken off over the last few years. Ensembles generally rely on "resampling" techniques to obtain different training sets for each individual regression or classification model. Two popular methods for creating accurate ensembles are bootstrap aggregating (Bagging) and Boosting.

In Bagging method, the training data for each individual model is drawn randomly, i.e., n instances with replacement- where n is the number of observations in the training set. In other words, successive members (e.g., trees or neural networks) in this method are independent of each other; since each member of the ensemble is trained individually using a bootstrap sample of the data set [5]. In other words, Bagging methods control the generalization error through perturbation and averaging of sub-models. Worth noting that in this approach, to ensure that every training sample is predicted at least a few times, the number of trees needs to be large enough. Since the trees are independent of each other, the distribution function and the quantiles of each hourly forecast can be easily computed based on the output of all the trees [40].

In Gradient Boosting method, however, the training set for each member of the ensemble depends on the performance of the previous model(s). More precisely, in order to alleviate the error in earlier models, extra weights are assigned to samples with higher prediction error rates; those are hence more likely to take part in the training of

the next model [16, 15]. A comprehensive evaluation of both these techniques on 23 data sets, using two popular classifiers, i.e., decision trees and neural networks is presented in [33]. The applicability of Gradient Boosting method in quantile regression load forecasting application have been put into practice in [38, 40]. A brief recapitulation of the Gradient Boosting method, as the main methodology used herein to predict the hourly load values, is provided in below.

The goal in every typical prediction problem is to determine an estimate or approximation $\hat{\mathbf{F}}(\mathbf{x})$, of the true mapping function $\mathbf{F}^*(\mathbf{x})$ which assigns a $y \in \mathbb{R}$ to any given set of covariates $\mathbf{x} \in \mathbb{R}^p$. This process is optimized by minimizing the expected value of some specified loss function $L(y, \mathbf{F}(\mathbf{x}))$ over the set of the joint distribution of all $\{y, \mathbf{x}\}$ pairs. In mathematical parlance, we have:

$$\mathbf{F}^*(\mathbf{x}) = \underset{\mathbf{F}(\mathbf{x})}{\arg\min}\, \mathbb{E}_{y,\mathbf{x}} L(y, \mathbf{F}(\mathbf{x})) = \underset{\mathbf{F}(\mathbf{x})}{\arg\min}\, \mathbb{E}_{\mathbf{x}}[\mathbb{E}_y(L(y, \mathbf{F}(\mathbf{x})))|\mathbf{x}] \tag{1}$$

where $\mathbb{E}(.)$ is the expectation operator, and $L(y, \mathbf{F}(\mathbf{x}))$ is a *loss* function, e.g., the popular choice of squared-error $\{y - \mathbf{F}(\mathbf{x})\}^2$, for regression problems. $\mathbf{F}(\mathbf{x})$ is a member of "additive" class of functions of the form:

$$\mathbf{F}(\mathbf{x}; \{\lambda_k, \mathbf{a}_k\}_1^K) = \sum_{k=1}^{K} \lambda_k h(\mathbf{x}; \mathbf{a}_k). \tag{2}$$

where K is the number of members of the ensemble model, λ_k is the coefficient of the additive model, the generic

Algorithm 1: The Gradient Boosting Algorithm, with an squared-error loss function, in a nutshell.

Initialization: $\mathbf{F_0}(\mathbf{x}) = \bar{y}$

for *k=1 to K* **do**

$\tilde{y}_i = y_i - \mathbf{F}_{k-1}(\mathbf{x}_i), \quad i \in [1 : N]$

$(\rho_k, \mathbf{a}_k) = \underset{\rho,\mathbf{a}}{\arg\min} \sum_{i=1}^{N} [\tilde{y}_i - \rho\, h(\mathbf{x}_i; \mathbf{a})]^2$

$\mathbf{F}_k(\mathbf{x}) = \mathbf{F}_{k-1}(\mathbf{x}) + \rho_k h(\mathbf{x}; \mathbf{a}_k)$

end

function $h(\mathbf{x}; \mathbf{a})$ in (2) is called a *weak learner* or *base learner*-it is usually a simple parameterized function of the explanatory variables, specified by parameters $\mathbf{a} = \{a_1, a_2, \ldots\}$. In the present work, each $h(\mathbf{x}; \mathbf{a}_k)$ is a small regression tree as introduced in [7]. For a regression tree the parameters $\mathbf{a}_k$ are the splitting variables, split locations and means of the terminal node of the individual trees. An overview of the Gradient Boosting algorithm, with an squared-error loss function is presented in Algorithm 1, where the multiplier ρ_k is given by the line search:

$$\rho_k = \arg\min_{\rho}\, \mathbb{E}_{y,\mathbf{x}} L(y, \mathbf{F}_{k-1}(\mathbf{x}) - \rho \mathfrak{g}_k(\mathbf{x})), \tag{3}$$

and

$$\mathfrak{g}_k(\mathbf{x}) = \mathbb{E}_y \left[\frac{\partial \mathbf{L}(y, \mathbf{F}(\mathbf{x}))}{\partial \mathbf{F}(\mathbf{x})} \right]_{\mathbf{F}(\mathbf{x})=\mathbf{F}_{k-1}(\mathbf{x})}. \tag{4}$$

The rationale underpinning the choice of the ensemble of the trees is their ability in better handling the heterogeneous input data–which comprise both continuous and discrete variables. Additionally, tree models are effective, adaptive and modular, in that new predictors can be easily added. It is perceived that ensemble models, unlike their other ML based counterparts, are less prone to overfitting; they promise to strike a good trade-off between bias and variance [6].

3.2. Our proposed forecasting models

Generally speaking, a regression tree is an adaptive nearest neighbours like algorithm. However, it usually shows a better performance in comparison with other counterpart nearest neighbour-based methods; it tends to find the homogeneous portions of the sampling space locally, on the contrary to the other conventional methods which incline to treat all distances equally [29]. In the present work, we follow a homogeneous forecast combination framework,

Attribute	*Description*	*Model no.*
mn	month of year: 1,2,...,12	I,II
wk	day of week: 1,2,...,7	I,II
hr	hour of day: 1,2,...,24	I,II
$\mathbf{L(d-1)}$	previous day (estimated) hourly load	I,II
$\mathbf{L(d-7)}$	previous week (estimated) hourly load	I,II
$\mathbf{T(d)}$	hourly temperature (generated profiles)	I,II
$\mathbf{L'(d-1)}$	1st derivative of $\mathbf{L(d-1)}$	II
$\mathbf{L''(d-1)}$	2nd derivative of $\mathbf{L(d-1)}$	II
$\mathbf{L'(d-7)}$	1st derivative of $\mathbf{L(d-7)}$	II
$\mathbf{L''(d-7)}$	2nd derivative of $\mathbf{L(d-7)}$	II
$\mathbf{T'(d)}$	1st derivative of $\mathbf{T(d)}$	II
$\mathbf{T''(d)}$	2nd derivative of $\mathbf{T(d)}$	II
$\mathbf{L(d)}$	hourly load (forecast target)	I,II

Table 1: An overview of the attributes used in our proposed models.

i.e., we, first, train a single-value load forecasting model, then, vary the input data (different temperature scenarios) to obtain a series of forecasts, and accordingly, the quantiles.

Time series data in smart energy systems often comprise more than one distinct time scales. There are some conspicuous diurnal patterns in the data, reflecting typical patterns for daily or weekly human activities. Furthermore, the overall structure of the data is affected by a combination of those fast diurnal patterns superimposed on slower seasonal variations [24]. We herein consider two Gradient Boosting based methods to predict the day-ahead load prognoses. In the first model, only calendar information, temperature data along with the historical load data are the input variables. We proceed further in the second model to incorporate the daily dynamics of the temperature and load profiles. It is done using the first and second derivatives of the daily profiles.

As it was principled in [19] the forecasting horizon is one month, therefore, the estimated values for the first week of the month are being used to estimate the load profiles in the second week of the month and so on. In all cases, the aim is to predict $\mathbf{L(d)}$, 24 hourly load values for the target day $\mathbf{d}$. Power consumption is subject to a wide range of exogenous variables, including calendar effects, electricity price and so on. In the literature, the previous consumption patterns and calendar information have been extensively used in developing various load forecasting models. However, accounting for the interaction between different variables, namely the recency and cross effects can be an onerous task; it demands some domain expertise to be done sensibly [28, 39]. A number of common deterministic (categorical) explanatory variables used in our methodologies are as follows: month of the year $\mathbf{mn} \in \{1, 2, \ldots, 12\}$, day of the week $\mathbf{wk} \in \{1, 2, \ldots, 7\}$ (starting from Sunday=1), and hour of the day $\mathbf{hr} \in \{1, 2, \ldots, 24\}$. Below we discuss the models in more details, but for ease of reference, Table 1 summarizes all the common and distinctive attributes used in two proposed models.

Model I*:* The first model provides us with a benchmark to measure the credibility of our proposed method in incorporating the recency and cross effects in the data in Model II. Here, we introduce six different attributes to predict the hourly load values on the target day $\mathbf{d}$. The three above mentioned common discrete (categorical) values, namely, $\mathbf{mn}, \mathbf{wk}, \mathbf{hr}$, along with the (estimated) load value for a given hour on a day or a week before ($\mathbf{L(d-1)}$ and $\mathbf{L(d-7)}$, respectively). The intuition for this choice is the reflection of diurnal and weekly human activities on electricity consumption (Figure 1). The last covariate $\mathbf{T(d)}$ is the hourly temperature forecast for the target day $\mathbf{d}$. As it it explained in Section 3.3, we generate one hundred independent temperature profiles, using the singular value decomposition, to correspondingly obtain 100 independent load forecasts for each target day; the combination of these forecasts are then used to obtain the load quantiles for each hour.

Model II*:* To reflect the lagging effect of temperature on load changes, in the second model, we add the daily dynamics of the temperature and load profiles (1st and 2nd derivatives) [24], [25]. For a given hour slot h the corresponding first derivative of the variable $z \in \{\mathbf{L}, \mathbf{T}\}$ can be obtained by $z'(h) = 0.5[z(h+1) - z(h-1)]$; with obvious analogues for the 2nd derivative. As previously mentioned (see Figure 2), the thinking here is to include the daily dynamics and

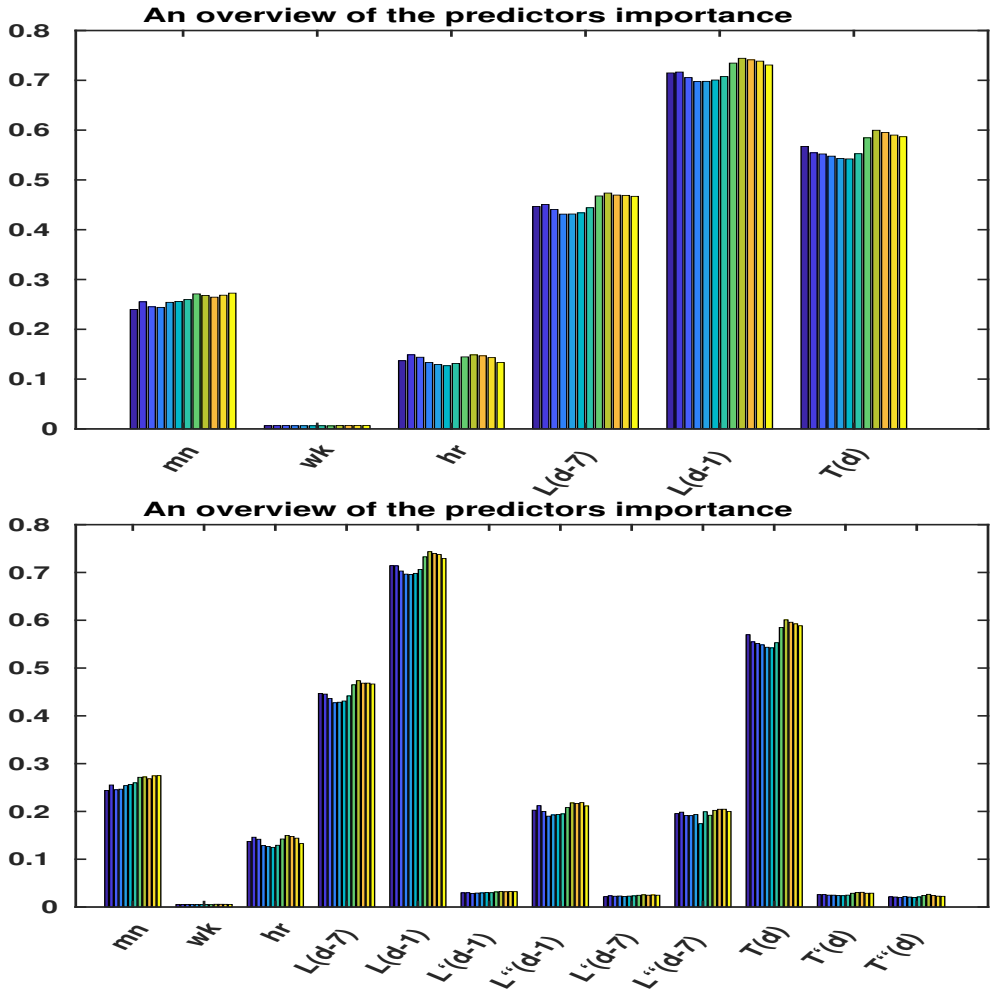

Figure 3: An overview of the importance of the predictors used in Model I (top), and Model II (bottom). The horizontal axis represents the predictors used in each model. The vertical axis illustrates their relative importance. 12 different colors represent 12 different models (one for each month, stating from the dark blue on the left for January 2011). Full description is followed in the text.

trends of load and temperature profiles as a new predictor. The reasoning for doing so is that oftentimes the actual values are not as important as the general underlying trends captured by the first or second derivatives of the covariates. In other words, load value at any moment is influenced by the variations of the other attributes (namely, temperature profiles) prior to that moment. Including the derivatives is, in fact, a relatively simple and generic means to account for the recency effect in the data. In comparison with most time-varying models, where the data is typically divided into subsets (based on thresholds), or a lagging window is optimized, our proposed approach is more straightforward and user-friendly. Figure 3 provides a comparison of the importance of the predictors used to train Model I and II. The importance was determined by summing up all the estimates over all weak learners in the ensemble [3]. Predictor with the highest value is the most important one. The bottom panel of Figure 3 highlights the fact that including the derivatives (especially the second derivatives) can indeed be helpful. As it become clear in Section 4, to predict the load values for every month of the year 2011, we train a new model using all the available data up to the beginning of that month (rolling window mechanism). Each shade of color in Figure 3, hence, corresponds to one experiment (dark blue on the left is for January 2011, and yellow, on the right for December 2011). A major contribution herein is the use of the singular value decomposition (SVD) to generate temperature scenarios $\mathbf{T}(\mathbf{d})$ for the target day $\mathbf{d}$; it is done to determine the distribution (99 percentiles) of the load profile in our proposed probabilistic forecasting models. A brief recapitulation to the singular value decomposition (SVD) technique is provided in below.

3.3. Singular value decomposition

As mentioned before, time series data in energy systems usually comprises at least two distinct time scales. Recasting such quasi-periodic time series as a matrix such that each column represents the values for a single day, could be helpful in gaining a better understanding of the data. The advantage of this approach is twofold. Primarily, representing a matrix as an image provides a thorough overview of the evolution of data in a certain time span; it hence can lead to better appreciation of indistinct or subtle features. Second, it makes it possible to draw on numerically stable matrix decomposition methods, such as SVD to elucidate the underlying data structure [24]. The SVD technique is used herein to generate new temperature profiles (matrices). To be more precise, we recast one year's worth of hourly temperature values as a matrix $\mathbf{T} \in \mathbb{R}^{24\times365}$ such that every column corresponds to 24 hourly values of a day. Consequently, the matrix $\mathbf{T}$ can conveniently be represented by a low-rank approximation. More specifically, given any arbitrary matrix $\mathcal{A} \in \mathbb{R}^{h\times d}$, there exist orthogonal matrices $\mathbf{U} \in \mathbb{R}^{h\times h}$ and $\mathbf{V} \in \mathbb{R}^{d\times d}$ (both with orthonormal columns) such that:

$$\mathcal{A} = \mathbf{U}\mathbf{S}\mathbf{V}^{\mathbf{T}} = \sum_{k=1}^{r} \sigma_{\mathbf{k}}\mathbf{u}_{\mathbf{k}}\mathbf{v}_{\mathbf{k}}^{\mathbf{T}} \tag{5}$$

where $\mathbf{S}$ is a diagonal matrix of the singular values $\sigma_{\mathbf{k}}$, such that its non-zero elements $\sigma_{\mathbf{1}} \geq \sigma_{\mathbf{2}} \geq \ldots \geq \sigma_{\mathbf{r}} \geq \mathbf{0}$ are positioned uniquely, in a descending order, on the main diagonal ($\mathbf{S}$ has the same size as $\mathcal{A}$ and $r = \min\{h, d\}$). Furthermore, $\mathbf{u}_{\mathbf{k}}$ and $\mathbf{v}_{\mathbf{k}}$, called the left and right singular vectors, denote the $\mathbf{k}^{th}$ column of $\mathbf{U}$ and $\mathbf{V}$, respectively [37]. If there are only a few dominant singular values (as it is the case for the temperature matrices, in Figure 4), the expansion of the matrix in (5) can be sufficiently truncated after just the first few K terms to yield $\mathcal{A}_K$, an adequate lower rank approximation of $\mathcal{A}$:

$$\mathcal{A}_K = \sum_{k=1}^{K} \sigma_{\mathbf{k}}\mathbf{u}_{\mathbf{k}}\mathbf{v}_{\mathbf{k}}^{\mathbf{T}} \quad \text{where} \quad K < r. \tag{6}$$

To elaborate more, Figure 5 illustrates the first three columns of $\mathbf{u}_{\mathbf{k}}$ (left) and $\mathbf{v}_{\mathbf{k}}$ (right) for temperature matrix for the year 2009. In geometrical terms, $\mathbf{u}_{\mathbf{k}}$ columns can be interpreted as the fundamental daily profile and its successive increments; $\mathbf{v}_{\mathbf{k}}$ values represent the corresponding scaling factors for each $\mathbf{u}_{\mathbf{k}}$ profiles for each day. In other words, SVD decomposes the original time series into a linear combination of a number of (data-driven) orthonormal profiles, specified by $\mathbf{u}_{\mathbf{k}}$ columns; each profile is then scaled up (or down) according to their corresponding weights in $\mathbf{v}_{\mathbf{k}}$. For instance, $\mathbf{u}_1$ in the top left panel of Figure 5 strikingly resembles the averaged daily temperature profile. Moreover, its corresponding $\mathbf{v}_1$ profile (top right panel) outlines the evolution of that profile throughout the year; it is in agreement with the fact that temperature is higher in summer time (middle part of the graph). Recall that these profiles are weighted based on the magnitude of their corresponding singular values which are sorted in descending order from left to right (Figure 4). The most dominant "corrective" incremental profile $\mathbf{u}_2$ and its corresponding coefficients $\mathbf{v}_2$ are displayed in the middle panel of Figure 5. This correction hence needs to be added to the first profile to get a better approximation, i.e., $K = 2$ in (6). Similar interpretations are valid for the third profile (bottom panels) and so on. It is worth noting that $\mathbf{v}_k$ profiles on the right-hand side of Figure 5 imply a distinct impression that temperatures are

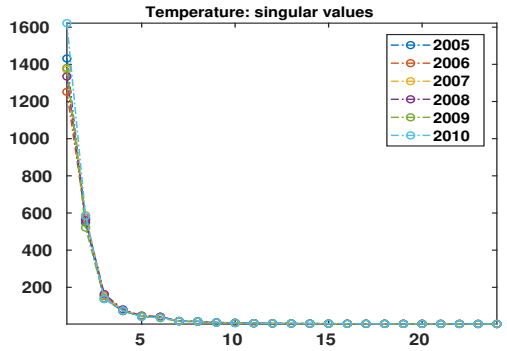

Figure 4: The evolution of the singular values of the temperature matrices over the years; it suggests that a reconstruction of rank-4 approximation would suffice, indicating that temperature is quite regular.

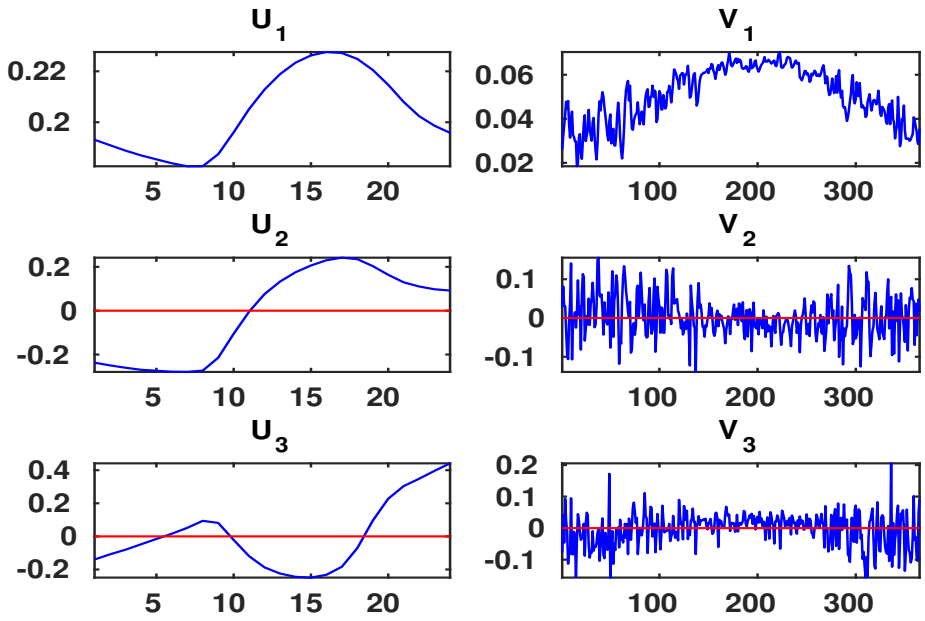

Figure 5: SVD-based decomposition of hourly temperature data for 2009. On the left, there are the first three $\mathbf{u_k}$ columns; whereas the right column displays their corresponding $\mathbf{v_k}$'s.

less variable during the summer (middle parts of the graph). In the following Section, SVD is practiced to simulate pragmatic temperature scenarios, in a systematic and data-driven manner. The generated profiles are accordingly fed to Models I and II to obtain the probability distribution (99 quantiles) of the load values for every given hour.

3.4. Temperature scenario generation

A common approach in probabilistic load forecasting problems is to vary the input (e.g., temperature profiles) to obtain a series of forecasts and combine them. One of the major challenges, however, is how to create realistic temperature profiles, as e.g., simply adding independent Gaussian noise to the hourly values of individual temperature curves results in some preposterously jagged profiles. In the literature, a number of solutions have been proposed to simulate temperature scenarios. In [42], it is proposed to combine different weather station measurements to generate new temperature profiles. Nonetheless, it can be argued that normal weather scenarios cannot precisely be simulated by averaging the temperature profiles, as they tend to underestimate the peaks. Furthermore, such approaches are not resilient toward outliers, as even one instance can change the whole profile as long as it takes part in generating new temperature scenarios; the performance of the forecasting model can consequently be diminished as a result of that. Worth noting that shifting the temperature data by one, two or even three days was initially used to generate temperature scenarios; it was later abandoned for obvious reasons. Some cumbersome approaches in terms of computational costs, such as Monte Carlo based methods are also popular, especially among utilities, to simulate thousands of temperature profiles - an approach which is used in scenario analysis in LTLF problems [21]. In [44], new temperature scenarios are generated, again, by averaging the temperature of stations 3 and 9 (GEFCom2014-L data was used). The reason for that is mentioned to be due to the existence of a good in-sample fit with a cubic relation between the temperature records of those two stations and the load data. Besides pre-processing there are not a lot of solutions in the literature on how to generate robust and pragmatic input (temperature) scenarios.

We herein propose a generic, data-driven and computationally efficient SVD-based approach for simulating temperature scenarios. SVD allows us to create hundreds of sensible and realistic temperature profiles for any target day $\mathbf{d}$, in a fairly fast and robust manner. Figure 4 affirms the fact that the singular values σ_k of the temperature matrices over the years have not changed much; similar conclusions can be drawn for the left singular vectors $\mathbf{u}_k$. Furthermore, it is plain to see in Figure 5 that the $\mathbf{v}_k$ coefficients implicate the variability of the temperature profiles throughout the year. Since the forecasting horizon is one month, hereafter temperature matrix is referred to a month worth of

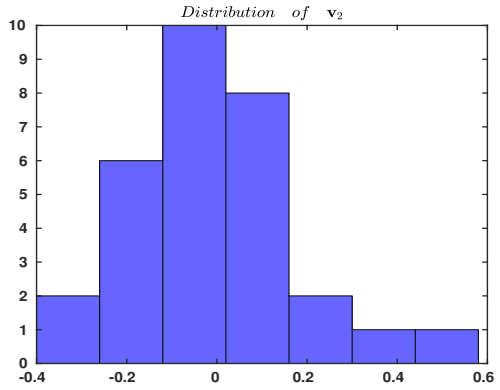

Figure 6: Histogram of the $\mathbf{v_2}$ coefficients for June 2011 temperature data (30 values). Note that the distribution is approximately normal with zero mean and $\mathbf{std}(\mathbf{v_2}) \approx \mathbf{0.18}$.

temperature data for the coming month (test data in Section 4). We, therefore, proceed with the following steps, to create temperature scenarios:

1. In the first step, we estimate the corresponding standard deviation $\mathbf{s}_k$ for a number of right singular vectors $\mathbf{v}_k$ ($\mathbf{v}_2, \ldots, \mathbf{v}_4$). Figure 6 illustrates the histogram for $\mathbf{v}_2$, which shows that $\mathbf{s}_2 \approx 0.18$. Interestingly enough, similar experiments on all three $\mathbf{v}_k$ columns ($k \geq 2$) yield similar results; however, their contribution to the final rank K reconstructed profile is scaled up or down by the magnitude of their corresponding singular values.
2. Next, for any given day $\mathbf{d}$ of the test month, for which a number of temperature scenarios are desired, we take the actual temperature profile for that day $\mathbf{T} = \mathbf{T}(\mathbf{d})$, find the corresponding $\mathbf{v}_k$ coefficients $(\mathbf{v}_2^0, \mathbf{v}_3^0, \mathbf{v}_4^0)$; then blend them with zero-mean Gaussian noise: $\mathbf{v}_k^n = \mathbf{v}_k^0 + \mathcal{N}(0, \epsilon^2)$. These perturbed $\mathbf{v_k}$ coefficients are then used to generate a new (noisy) temperature scenario (reconstruct the matrix).
3. According to the scheme outlined above, for each actual daily profile $\mathbf{T}(\mathbf{d})$, a hundred temperature scenarios are being generated. This new data set is then fed into the proposed prediction models; in the end, the forecasts are duly compared to the real load values. This enables us to determine the distribution of the hourly load values (99 quantiles) and compute the corresponding pinball error values (Section 4). Figure 7 illustrates an example of one hundred generated temperature profiles (Left), and their corresponding daily load profiles (Right).
4. For the sake of completeness, it should be noted that $\mathbf{v}_1$ is left unperturbed as this is a proxy of the average temperature on a particular day, for which the uncertainty is negligible. Similarly, there is not much to be gained from perturbing other right singular vectors ($\mathbf{v}_5$ etc), as their impact on the profile is insignificant (their corresponding σ_k are small).

In [25], a preliminary study was done to investigate how the effect of the perturbation variance in temperature profiles $\mathbf{T}(\mathbf{d})$ propagates into uncertainty on the target load profile $\mathbf{L}(\mathbf{d})$.

4. Experimental Results

Probabilistic forecasts can provide more comprehensive information about future uncertainties that what point forecasts can do [18]. As previously mentioned, the aim of the GEFCom2014-L was to estimate the quantiles of the hourly load values for a utility in the US, on a rolling basis [19]; the forecasting horizon was one month. Furthermore, it was expected from the contestants to investigate the weather scenario generation methods for probabilistic load forecasting. The scenario-based probabilistic forecasting methodology proposed by [21] was used by two top 8 teams (Jingrui Xie, top 3; Bidong Liu, top 8) in GEFCom2014-L. Therefore, we opt to compare our results with similar works.

A least absolute shrinkage and selection operator (LASSO) estimation based method is proposed in [44] for probabilistic load forecasting. This work is reported to outperform the methodology used by Bidong Liu to win a top 8 place in GEFCom2014-L. We hence have considered the proposed method in [44] as one of the benchmark models.

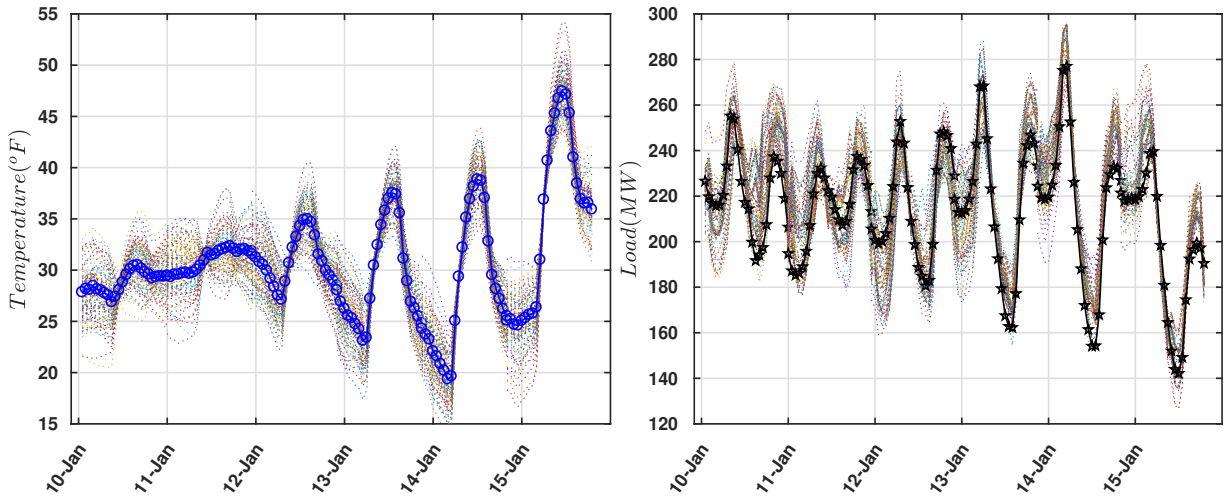

Figure 7: An illustrative example of 100 generated temperature scenarios for each day and their corresponding daily load profiles (obtained by Model II) for January 10-15, 2011. The spotted points are the actual values and the noise level is $\mathcal{N}(0, 0.09)$. These 100 different load values for every hour are then used to calculate the 99 quantiles.

The reported work uses a bivariate time-varying threshold autoregressive (AR) process for the hourly load $Y_{\mathcal{L},t}$ and temperature $Y_{\mathcal{T},t}$ data ($\mathcal{D} = \{\mathcal{L}, \mathcal{T}\}$). The time series of interest are accordingly modeled for $i \in \mathcal{D}$ as follow:

$$Y_{i,t} = \phi_{i,0}(t) + \sum_{j\in\mathcal{D}} \sum_{c\in C_{i,j}} \sum_{k\in I_{i,j,c}} \phi_{i,j,c,k}(t) \max\{Y_{j,t-k}, c\} + \epsilon_{i,t} \tag{7}$$

where $\phi_{i,0}$ are the time-varying intercepts and $\phi_{i,j,c,k}$ are time-varying autoregressive coefficients. Furthermore, $C_{i,j}$ are the set of all considered thresholds for the load and temperature data (all set manually). $I_{i,j,c}$ are the index sets of the corresponding lags and $\epsilon_{i,t}$ is the error term. The modelling process is done in three parts: 1) choice of thresholds $C_{i,j}$; 2) choice of lag sets $I_{i,j,c}$; and 3) time-varying structure of the coefficients. For further details see [44].

Another winning team (top 3) in the GEFCom2014-L was Jingrui Xie, who developed an integrated solution for probabilistic load forecasting [42]. Her proposed methodology consists of three parts: 1) pre-processing, which includes data cleaning and temperature station selection; 2) forecasting (which focuses on the development of point forecasting models), forecast combination, and temperature scenario generation; and 3) post-processing, which embodies the residual simulation for probabilistic forecasting purposes. Inspired by the *Vanilla* model in [28], their core forecasting model is as follow:

$$\begin{aligned}\mathcal{L}_t = \beta_0 + \beta_1 \text{Trend}_t + \beta_2 \mathcal{T}_t + \beta_3 \mathcal{T}_t^2 + \beta_4 \mathcal{T}_t^3 + \beta_5 \text{Month}_t + \beta_6 \text{Weekday}_t + \beta_7 \text{Hours}_t + \beta_8 \text{Hours}_t \text{Weekday}_t + \\ \beta_9 \mathcal{T}_t \text{Month}_t + \beta_{10} \mathcal{T}_t^2 \text{Month}_t + \beta_{11} \mathcal{T}_t^3 \text{Month}_t + \beta_{12} \mathcal{T}_t \text{Hour}_t + \beta_{13} \mathcal{T}_t^2 \text{Hour}_t + \beta_{14} \mathcal{T}_t^3 \text{Hour}_t\end{aligned} \tag{8}$$

It is in fact a multiple linear regression (MLR) model with the following main and cross effects:

- Main effects: a chronological Trend_t variable, first to third order polynomials of the temperature ($\mathcal{T}_t, \mathcal{T}_t^2, \mathcal{T}_t^3$), and a number of categorical variables namely, Month, weekday, and Hour.
- Cross effects: similar to [28], the cross effects are incorporated using the multiplications of different attributes such as $\text{Hour}_t\text{Weekday}_t$, $\mathcal{T}_t\text{Month}_t$, $\mathcal{T}_t^2\text{Month}_t$, $\mathcal{T}_t^3\text{Month}_t$, $\mathcal{T}_t\text{Hour}_t$, $\mathcal{T}_t^2\text{Hour}_t$, and $\mathcal{T}_t^3\text{Hour}_t$.

In the next step, the residuals obtained from (8) is modeled using four different techniques, namely unobserved component models (UCM), exponential smoothing models (ESM), three-layer feedforward neural network (NN), and autoregressive integrated moving average models (ARIMA). Four different sets of point forecasts are accordingly generated by adding each set of residuals to the values obtained from the previous stage. The average of each four

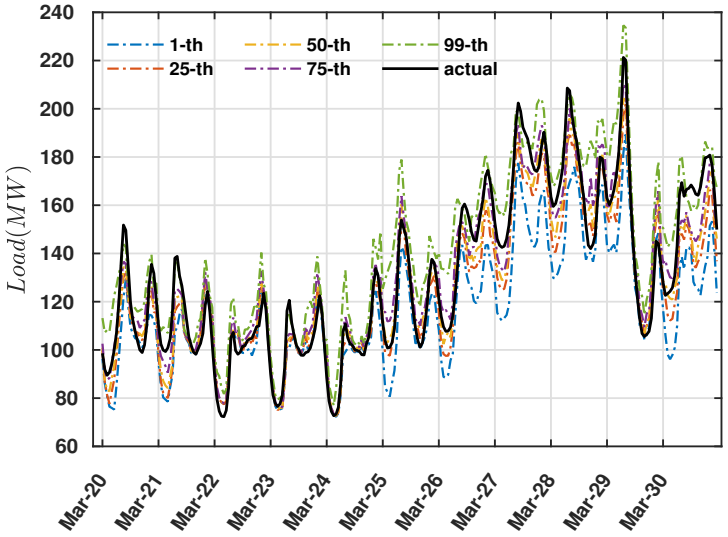

Figure 8: Probabilistic load forecast of 11 days from March 20, 2011 to March 30, 2011; the solid line in black is the actual value and the dash-dot lines are the forecast quantiles.

value is the final estimation for the load forecast for every given hour. In the end, 10 different temperature scenarios are generated according to [21], to obtain the 99 percentiles from the 10 point forecasts.

The regression-based models are arguably vulnerable towards outliers, especially in the scenario based applications. Due to the recency effects, outliers, e.g., in temperature scenarios, can affect the load forecasts for a longer time span. Our proposed SVD-based model is more robust and capable of handling this issue. As mentioned before, for every hour of the target day $\mathbf{L}(\mathbf{d})$, we obtain 100 different load values (right panel in Figure 7). The results are then being used to determine the distribution of the hourly load values (99 different quantiles) for any given hour by employing linear extrapolations [26]. An illustrative example of the predicted quantiles for 11 days, March 20-30, 2011, is provided in Figure 8.

Pinball loss is a comprehensive index to evaluate the reliability, sharpness, and calibration of the forecasts. It is an extensively used error measure for quantile forecasts in probabilistic forecasting problems. The performance of the forecasting models in GEFCom2014 was evaluated by the overall mean of the pinball loss values. Recall that the

Month	[44]	[42]	**Model I**	**Model II**
1	9.88	11.87	3.43	**3.23**
2	9.54	10.93	3.24	**2.89**
3	7.79	8.44	2.69	**2.56**
4	4.89	4.50	2.53	**2.30**
5	5.96	7.27	**3.33**	3.50
6	5.86	6.99	4.98	**4.66**
7	7.66	9.05	3.63	**3.42**
8	10.70	11.26	8.71	**8.58**
9	6.28	5.49	4.46	**4.05**
10	5.20	3.36	2.97	**2.76**
11	6.38	5.90	**3.50**	3.59
12	8.99	9.73	3.57	**3.36**

Table 2: The left two columns are the reported results in [44], and [42]. The results of our proposed two different models are presented on the right part. The results reported here are the average of 100 iterations (no. of trees is 100, and MaxNumSplits=128).

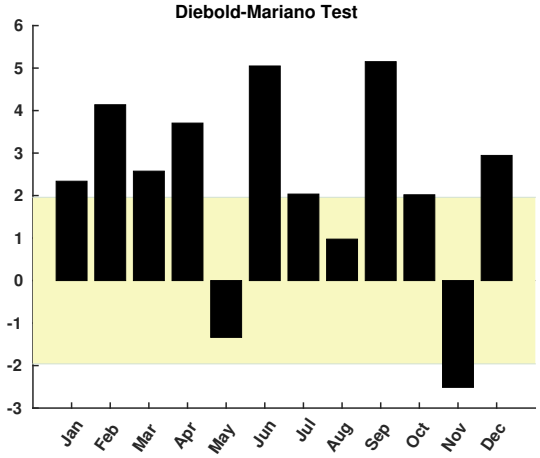

Figure 9: Comparison of the two models I and II, using Diebold-Mariano test ($h = 1$ and $k = 0$).

pinball loss function can be written as:

$$\text{Pinball}(\hat{y}_{t,q}, y_t, q) = \begin{cases} (1-q)(\hat{y}_{t,q} - y_t) & \text{if } \hat{y}_{t,q} > y_t \\ q(y_t - \hat{y}_{t,q}) & \text{if } \hat{y}_{t,q} \leq y_t \end{cases} \tag{9}$$

where y_t is the target hourly value of the load profile from [19], and $\hat{y}_{t,q}$ is the corresponding forecast value at the $q-$th quantile ($q \in \{0.01, 0.02, \ldots, 0.99\}$); it is obtained from one of the models specified above. To evaluate the full predictive densities, pinball scores obtained from (9) are averaged over the time horizon (99 quantiles for every hour, 24 hours of the day, n days of the month). A better forecast yield a lower pinball score. For more details on the pinball loss function and the evaluation methods used in GEFCom2014, see [19]. Table 2 contains the results of our proposed models along with two benchmark models. It is worth noting that all the data prior to the target month have taken part in the training of each model, i.e., the first eleven months in 2011 were used for training a model to predict the load profiles in December 2011. Furthermore, the average of 100 different hourly load values (right panel of Figure 7) is used as a proxy for the actual load value anytime needed; it is done because in the later days of the month, the earlier load profiles are needed in the form of $\mathbf{L}(\mathbf{d}-\mathbf{1})$ or $\mathbf{L}(\mathbf{d}-\mathbf{7})$. The results in Table 2, together with Figure 3 highlights the fact that including the derivatives (especially the 2nd derivatives) is indeed helpful in enhancing the performance of the forecasting model.

To investigate further, Figure 9 contains the Diebold-Mariano test [17], [13] to determine whether two models are significantly different. These results were obtained by comparing the error between the median ($q = 0.5$) of the forecasts from two models and the actual values. The results are the average of 100 iterations, calculated according to (10). Suppose that the significance level of the test is $\alpha = 0.05$. For a two-tailed test, therefore, the upper and lower tails would each be 0.025. Accordingly, the upper and lower $z-$values are 1.96 and -1.96, respectively [1]. The null hypothesis of no difference between the two models (forecasts) will be rejected if the computed Diebold-Mariano statistic falls outside the range of [-1.96 , 1.96]. Consistent with the results in Table 2, in February, June and September 2011, Models I and II are most significantly different. On the other hands, in August, where both models have the highest pinball score, the Diebold-Mariano (DM) test is low. Finally, DM tests in May and November 2011, are negative, as Model I outperforms Model II. We use the Diebold-Mariano test to determine whether forecasts are significantly different. Let e_{i1} and e_{i2} be the residuals for Model I and II, respectively ($i \in [1:n]$). n is the number of

data points, and k is the lagging variable [2].

$$
\begin{aligned}
d_i &= |e_{i1}|^2 - |e_{i2}|^2 \\
\bar{d} &= \frac{1}{n}\sum_{i=1}^{n} d_i \\
\gamma_k &= \frac{1}{n}\sum_{i=k+1}^{n}(d_i - \bar{d})(d_{i-k} - \bar{d}), \quad n > k \geq 1 \\
DM &= \frac{\bar{d}}{\sqrt{[\gamma_0 + 2\sum_{k=1}^{h-1}\gamma_k]/n}}, \quad h \geq 1
\end{aligned}
\tag{10}
$$

5. Conclusions

This paper proposes two generic scenario-based probabilistic load forecasting models using an ensemble of regression trees. An important distinction of the current work is in recasting quasi-periodic time series data as matrices. The singular value decomposition technique is then used to generate temperature scenarios, in a robust and data-driven manner. In the second model, we extend the first one by adding the first and second derivatives of the non-deterministic attributes (temperature and historical load data). It was done to partially account for the recency effects and interactions among the data. The empirical case studies performed on the data from the load forecasting track of the Global Energy Forecasting Competition 2014 (GEFCom2014-L) show how the proposed models outperform two benchmark scenario-based models with a similar set-up.

Acknowledgment

The authors gratefully acknowledge partial support by the Dutch NWO-TTW under project grant *Smart Energy Management and Services in Buildings and Grids (SES-BE)*. The authors also would like to thank the anonymous reviewers for their fruitful comments.

References

[1] , a. Comparing predictive accuracy of two forecasts: The diebold-mariano test. `http://www.phdeconomics.sssup.it/documents/Lesson19.pdf`.

[2] , b. Diebold-mariano test statistic. `https://nl.mathworks.com/matlabcentral/fileexchange/33979-diebold-mariano-test-statistic?focused=7267180&tab=function`.

[3] , . Estimates of predictor importance. `https://nl.mathworks.com/help/stats/compactregressionensemble.predictorimportance.html`.

[4] Alobaidi, M.H., Chebana, F., Meguid, M.A., 2018. Robust ensemble learning framework for day-ahead forecasting of household based energy consumption. Applied Energy 212, 997–1012.

[5] Breiman, L., 1996. Bagging predictors. Machine learning 24, 123–140.

[6] Breiman, L., 2001. Random forests. Machine learning 45, 5–32.

[7] Breiman, L., Friedman, J., Stone, C., Olshen, R., 1984. Classification and Regression Trees. The Wadsworth and Brooks-Cole statistics-probability series, Taylor & Francis.

[8] Ceperic, E., Ceperic, V., Baric, A., 2013. A strategy for short-term load forecasting by support vector regression machines. IEEE Transactions on Power Systems 28, 4356–4364.

[9] Che, J., Wang, J., 2014. Short-term load forecasting using a kernel-based support vector regression combination model. Applied energy 132, 602–609.

[10] Chen, Y., Xu, P., Chu, Y., Li, W., Wu, Y., Ni, L., Bao, Y., Wang, K., 2017. Short-term electrical load forecasting using the support vector regression (svr) model to calculate the demand response baseline for office buildings. Applied Energy 195, 659–670.

[11] Coelho, V.N., Coelho, I.M., Coelho, B.N., Reis, A.J., Enayatifar, R., Souza, M.J., Guimarães, F.G., 2016. A self-adaptive evolutionary fuzzy model for load forecasting problems on smart grid environment. Applied Energy 169, 567–584.

[12] Craparo, E., Karatas, M., Singham, D.I., 2017. A robust optimization approach to hybrid microgrid operation using ensemble weather forecasts. Applied energy 201, 135–147.

[13] Diebold, F.X., Lopez, J.A., 1996. 8 forecast evaluation and combination. Handbook of statistics 14, 241–268.

[14] Fan, S., Chen, L., 2006. Short-term load forecasting based on an adaptive hybrid method. IEEE Transactions on Power Systems 21, 392–401.

[15] Freund, Y., Schapire, R., Abe, N., 1999. A short introduction to boosting. Journal-Japanese Society For Artificial Intelligence 14, 1612.
[16] Freund, Y., Schapire, R.E., et al., 1996. Experiments with a new boosting algorithm, in: Icml, Citeseer. pp. 148–156.
[17] Harvey, D., Leybourne, S., Newbold, P., 1997. Testing the equality of prediction mean squared errors. International Journal of forecasting 13, 281–291.
[18] Hong, T., Fan, S., 2016. Probabilistic electric load forecasting: A tutorial review. International Journal of Forecasting 32, 914–938.
[19] Hong, T., Pinson, P., Fan, S., Zareipour, H., Troccoli, A., Hyndman, R.J., 2016. Probabilistic energy forecasting: Global energy forecasting competition 2014 and beyond.
[20] Hong, T., Wang, P., 2014. Fuzzy interaction regression for short term load forecasting. Fuzzy optimization and decision making 13, 91–103.
[21] Hong, T., Wilson, J., Xie, J., 2014. Long term probabilistic load forecasting and normalization with hourly information. IEEE Transactions on Smart Grid 5, 456–462.
[22] Hyndman, R., Koehler, A.B., Ord, J.K., Snyder, R.D., 2008. Forecasting with exponential smoothing: the state space approach. Springer Science & Business Media.
[23] Keshtkar, A., Arzanpour, S., 2017. An adaptive fuzzy logic system for residential energy management in smart grid environments. Applied Energy 186, 68–81.
[24] Khoshrou, A., Dorsman, A.B., Pauwels, E.J., 2017. Svd-based visualisation and approximation for time series data in smart energy systems, in: Innovative Smart Grid Technologies Conference Europe (ISGT-Europe), 2017 IEEE PES, IEEE. pp. 1–6.
[25] Khoshrou, A., Pauwels, E.J., 2017. Propagating uncertainty in tree-based load forecasts, in: 2017 10th International Conference on Electrical and Electronics Engineering (ELECO), pp. 120–124.
[26] Langford, E., 2006. Quartiles in elementary statistics. Journal of Statistics Education 14.
[27] Li, S., Goel, L., Wang, P., 2016. An ensemble approach for short-term load forecasting by extreme learning machine. Applied Energy 170, 22–29.
[28] Liu, B., Nowotarski, J., Hong, T., Weron, R., 2017. Probabilistic load forecasting via quantile regression averaging on sister forecasts. IEEE Transactions on Smart Grid 8, 730–737.
[29] Loh, W.Y., 2011. Classification and regression trees. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 1, 14–23.
[30] Lusis, P., Khalilpour, K.R., Andrew, L., Liebman, A., 2017. Short-term residential load forecasting: Impact of calendar effects and forecast granularity. Applied Energy 205, 654–669.
[31] Mendes-Moreira, J., Soares, C., Jorge, A.M., Sousa, J.F.D., 2012. Ensemble approaches for regression: A survey. ACM Computing Surveys (CSUR) 45, 10.
[32] Misiorek, A., Trueck, S., Weron, R., 2006. Point and interval forecasting of spot electricity prices: Linear vs. non-linear time series models. Studies in Nonlinear Dynamics & Econometrics 10.
[33] Opitz, D., Maclin, R., 1999. Popular ensemble methods: An empirical study. Journal of artificial intelligence research 11, 169–198.
[34] Papalexopoulos, A.D., Hesterberg, T.C., 1990. A regression-based approach to short-term system load forecasting. IEEE Transactions on Power Systems 5, 1535–1547.
[35] Ramanathan, R., Engle, R., Granger, C.W., Vahid-Araghi, F., Brace, C., 1997. Short-run forecasts of electricity loads and peaks. International journal of forecasting 13, 161–174.
[36] Singh, P., Dwivedi, P., 2018. Integration of new evolutionary approach with artificial neural network for solving short term load forecast problem. Applied Energy 217, 537–549.
[37] Strang, G., 1993. Introduction to linear algebra. volume 3. Wellesley-Cambridge Press Wellesley, MA.
[38] Taieb, S.B., Hyndman, R.J., 2014. A gradient boosting approach to the kaggle load forecasting competition. International journal of forecasting 30, 382–394.
[39] Wang, P., Liu, B., Hong, T., 2016. Electric load forecasting with recency effect: A big data approach. International Journal of Forecasting 32, 585–597.
[40] Wang, Y., Zhang, N., Tan, Y., Hong, T., Kirschen, D.S., Kang, C., 2018. Combining probabilistic load forecasts. IEEE Transactions on Smart Grid .
[41] Xiao, L., Shao, W., Yu, M., Ma, J., Jin, C., 2017. Research and application of a hybrid wavelet neural network model with the improved cuckoo search algorithm for electrical power system forecasting. Applied energy 198, 203–222.
[42] Xie, J., Hong, T., 2016. Gefcom2014 probabilistic electric load forecasting: An integrated solution with forecast combination and residual simulation. International Journal of Forecasting 32, 1012–1016.
[43] Xie, J., Hong, T., 2018. Temperature scenario generation for probabilistic load forecasting. IEEE Transactions on Smart Grid 9, 1680–1687.
[44] Ziel, F., Liu, B., 2016. Lasso estimation for gefcom2014 probabilistic electric load forecasting. International Journal of Forecasting 32, 1029–1037.