Title
Building Electricity Load Forecasting via Stacking Ensemble Learning Method with Moving Horizon Optimization

Authors
Burger, Eric M.
Moura, Scott J.

Publication Date
2015-12-03

Building Electricity Load Forecasting via Stacking Ensemble Learning Method with Moving Horizon Optimization

Eric M. Burger, Scott J. Moura

***Abstract*—The short-term forecasting of building electricity demand is certain to play a vital role in the future power grid. Given the deployment of intermittent renewable energy sources and the ever increasing consumption of electricity, the generation of accurate demand-side electricity forecasts will be valuable to both grid operators and building energy management systems. The literature is rich with forecasting models for individual buildings. However, an ongoing challenge is the development of a broadly applicable method for electricity forecasting across geographic locations, seasons, and use-types. This paper addresses the need for a generalizable approach to electricity demand forecasting through the formulation of a stacking ensemble learning method. Rather than using a single model to predict electricity demand, our method uses a weighted linear combination of forecasts from multiple sub-models. By learning the model weights in real-time using electricity demand data streams and a moving horizon training technique, the method is more robust than a single model approach. By applying our method to electricity demand data sets for 8 different buildings, we show that this data-driven approach is capable of producing accurate multivariate forecasts for building level applications.**

***Index Terms*—building electricity load forecasting, stacking ensemble learning, moving horizon optimization, ordinary least squares (OLS) regression, least squares with L2 regularization (Ridge) regression, k-nearest neighbors (k-NN) regression**

I. Introduction

A. Motivation & Background

Commercial and residential buildings account for 74.1% of U.S. electricity consumption, more than either the transportation sector or the industrial sector (0.2% and 25.7%, respectively) [1]. Maintaining a continuous and instantaneous balance between generation and load is a fundamental requirement of the electric power system [2]. To reliably match supply with demand, the forecasting of grid-level electricity loads has long been a central part of the planning and management of electrical utilities [3]. The accuracy of these forecasts has a strong impact on the reliability and cost of power system operations. Trends, such as vehicle electrification and distributed generation, are expected to pose new challenges for grid operators. In particular, traditionally centralized power flow and generator dispatch tasks are becoming increasingly decentralized, creating a critical need for local electricity forecasting.

E. M. Burger and S. J. Moura are with the Energy, Controls, and Applications Lab (eCAL), 611 Davis Hall, Civil and Environmental Engineering Department, University of California Berkeley, Berkeley, CA, USA. E-mail: ericburger@berkeley.edu

To improve the accuracy of electricity demand forecasts and aid in power system management, recent attention has been placed on short-term building-level electricity demand forecasting using a wide range of models [4][5]. Accurate and adaptive forecasting of demand-side loads will play a critical role in maintaining grid stability and enabling renewables integration. Additionally, many novel optimal control schemes, under research umbrellas such as demand response and microgrid management, require short-term building electricity demand forecasts to aid in decision making [6].

B. Literature Review

Supply-side and demand-side electricity forecasting has been a topic of research for many decades. The literature is filled with a variety of well-cited modeling approaches, each differing in algorithmic complexity, estimation procedure, and computational cost. Of particular note are the variants of Artificial Neural Networks (ANN) [3][4][5][7][8][9][10], Support Vector Regression (SVR) [11][12][13][14] and Autoregressive Integrated Moving Average (ARIMA) models [3][12][13][15][16][17][18]. Lesser but nonetheless noteworthy attention has been given to approaches such as Multiple Linear Regression [3][11][19], Fuzzy Logic [3][20], Decision Trees [4], and k-Nearest Neighbors (k-NN).

These studies provide a broad catalog of use-cases and demonstrate the performance of certain forecasting algorithms when applied to specific building types. In particular, [3][4][10][17] provide a survey of electricity forecasting methods and a high-level comparison of techniques. In [8], the authors provide a detailed description of ANNs and their application to load forecasting, including data pre-processing and ANN architectures. The work in [5] details the development of a seasonal ANN approach and the advantage over a Seasonal ARIMA (SARIMA) model when applied to 6 building datasets. The focus of [18] is on the introduction of motion sensor data to improve the accuracy of an ARIMA model. In [9][11][15][18][20], the authors perform an in-depth analysis of the power demand patterns of a particular building in order to customize a forecasting model.

In papers with experimental results, the authors have generally applied their electricity demand forecasting technique to only a small number of datasets. Consequently, the literature is rich with forecasting algorithms tailored for individual buildings. This leads us to the following question: Is it possible to design a minimally-customized forecasting algorithm that is widely applicable across a diversity of building types, enabling scalability? We pursue this question by proposing a stacking ensemble learning method for electricity demand forecasting.

Specifically, due to unique building characteristics, occupancy patterns, and individual energy use behaviors, the literature demonstrates that no single "silver bullet" model structure can accurately forecast electricity demand across all buildings. For example, some forecasting models may produce accurate predictions under identifiable conditions, such as a seasonal trend, a morning routine, or an extended absence. Other models may be ideal for buildings with energy use behaviors that are periodic over long periods of time. For buildings with frequent changes in occupancy patterns, recursively trained models may yield the highest accuracy.

C. Contributions

A key contribution of this paper is to develop an ensemble learning method that reduces the need for intensive model selection on a case-by-case basis. Rather, an engineer can select a set of different forecasting models that have proven effective in past case studies (e.g. the literature cited above). Once the models have been trained on building-specific electricity demand records, our ensemble method can learn, in real-time, which sub-models to favor and which to avoid for a particular building.

With our stacking ensemble method, we generate electricity demand forecasts using the weighted sum of predictions from multiple different forecasting sub-models. The sub-model weights are recursively learned using an electricity demand data stream and a moving horizon training technique. In this way, the ensemble method is able to learn in real-time and to produce short-term electricity demand forecasts that are automatically tailored to a particular building and instance in time. In addition to forecast accuracy, this paper will place an emphasis on method adaptability and ease of use. While we have implemented certain forecasting sub-models in this paper, the method is intended to allow the sub-models to be interchangeable.

D. Assumptions

In this paper, we will make the following assumptions with respect to the availability of building electricity demand data:

- A1. We have access to hourly historical building electricity demand at the meter.
- A2. We have access to hourly historical weather data near the building location.
- A3. We do not have access to submetered electricity demand data or building operations data, such a occupancy measurements or mechanical system schedules.

The limited access to input data with which to produce forecasts is representative of the challenge faced by grid operators. Accordingly, this paper will demonstrate the potential of our ensemble method to non-invasively forecast total electricity demand using data-driven methods. Additionally, unlike in [9][11][15][18][20], where the authors perform an in-depth analysis of the power demand patterns in order to customize a model to a particular building, this paper will focus on developing a forecasting approach that is generally applicable to all buildings with minimal customization.

E. Outline

This paper is organized into three sections: Methods, Results, and Conclusions. In Section II. Methods, we briefly present background theory for the two exemplary regression sub-models employed in this paper, Ordinary (Linear) Least Squares with ℓ_2 Regularization (Ridge) and k-Nearest Neighbors (k-NN). These regression models will compose the sub-models in our ensemble method. Additionally, Section II presents the stacking ensemble learning method for electricity demand forecasting with a moving horizon training technique. In Section III. Results, we apply and analyze the ensemble method to 8 commercial/university building electricity demand datasets. Key conclusions and future research directions are summarized in Section IV. Conclusions.

II. Methods

A. Regression Models

In this paper, we will consider one parametric regression model, Ordinary (Linear) Least Squares with ℓ_2 Regularization (Ridge), and one nonparametric model, k-Nearest Neighbors with uniform weights and binary tree data structure (k-NN), for use as sub-models in our stacking ensemble method. The structure of both regression models are briefly described in the following subsections. While we have elected to employ relatively simple regression models, our ensemble method is such that these models could easily be replaced with more complex regression models, such as Artificial Neural Networks (ANNs) or Seasonal Autoregressive Integrated Moving Average (SARIMA) models.

B. Ordinary Least Squares with ℓ_2 Regularization

Ordinary Least Squares with ℓ_2 Regularization (Ridge) fits a linear model with coefficients $w \in \mathbf{R}^n$ to minimize the sum of squared errors between the observed and predicted responses, while imposing a penalty on the size of coefficients measured by their ℓ_2-norm. The linear model of a system with univariate output is given by

$$\begin{aligned}\hat{y} &= w_0x_0 + w_1x_1 + \ldots + w_nx_n \\ &= \sum_k w_kx_k = w^Tx\end{aligned} \tag{1}$$

with variables $x \in \mathbf{R}^n$, the model input, $\hat{y} \in \mathbf{R}$, the predicted response, n, the number of inputs or features in x, and $k = 1, \ldots, n$.

The linear model is trained on a set of inputs and observed responses by solving the quadratic program:

$$\min_{w} \sum_i \|w^Tx_i - y_i\|_2^2 + \lambda\|w\|_2^2 \tag{2}$$

with variables $x_i \in \mathbf{R}^n$, the model input for the i-th data point, $y_i \in \mathbf{R}$, the i-th observed response, $w \in \mathbf{R}^n$, the weighting coefficients, and $i = 1, \dots, N$, where N is the number of data samples and n is the number of features in x_i. Lastly, λ is a weighting term for the regularization penalty.

For a system with a multivariate output $\hat{y} \in \mathbf{R}^m$, we will treat the outputs as uncorrelated and define a set of coefficients $w_j \in \mathbf{R}^n$ for each predicted response $\hat{y}_j \in \mathbf{R}$ for $j = 1, \dots, m$. Thus, the multivariate linear model is

$$\hat{y}_j = w_j^T x, \ \forall j = 1, \dots, m \tag{3}$$

The weights of the multivariate model are determined by solving the quadratic program:

$$\min_w \sum_i \sum_j \|w_j^T x_i - y_{i,j}\|_2^2 + \sum_j \lambda \|w_j\|_2^2 \tag{4}$$

with variables $x_i \in \mathbf{R}^n$, the model input, $y_{i,j} \in \mathbf{R}$, the j-th response observed response, $w_j \in \mathbf{R}^n$, the weighting coefficients of the j-th response, $i = 1, \dots, N$, and $j = 1, \dots, m$, where N is the number of data samples, n is the number of features in x_i, and m is the number of observations in y_i.

C. k-Nearest Neighbors Regression

In k-Nearest Neighbors Regression (k-NN), an input $x \in \mathbf{R}^n$ is mapped to a continuous output value according to the weighted mean of the k nearest data points or neighbors, as defined by the Euclidean distance. In this paper, we will use uniform weights. In other words, each point in a neighborhood a contributes uniformly and thus the predicted univariate response $\hat{y} \in \mathbf{R}$ is the mean of the k-nearest neighbors.

$$\hat{y} = \frac{1}{k} \sum_{i=1}^{k} y_{a,i} \tag{5}$$

with variable y_a, the set of k observed responses $y \in \mathbf{R}$ in neighborhood a. For a system with multivariate output $\hat{y} \in \mathbf{R}^m$, the model is defined as the mean of each observation j over the k-nearest neighbors.

$$\hat{y}_j = \frac{1}{k} \sum_{i=1}^{k} y_{a,i,j} \ \forall j = 1, \dots, m \tag{6}$$

Given a new input x, it is possible to determine the neighborhood by computing the Euclidean distance (i.e. ℓ_2-norm of the difference) between the new input x and every data point in the training data set x_i for $i = 1, \dots, N$ and then ordering the distances to identify the nearest neighbors. However, this brute-force search is computationally inefficient for large datasets.

To improve the efficiency of the neighborhood identification, the training data points are partitioned into a tree data structure. A commonly used approach for organizing points in a multi-dimensional space is the ball tree data structure, a binary tree in which every node defines a D-dimensional hypersphere or ball. At each node, data points are assigned to the left or right balls according to their distance from the ball's center. At each terminal node or leaf, the data points are enumerated inside the ball. We refer the reader to [21] for a description of ball tree construction algorithms.

D. Stacking Ensemble Learning

In this section, we develop a regression method that produces a prediction according to the weighted sum of predictions from multiple sub-models. Ensemble learning methods which linearly combine the predictions of multiple models are generally referred to as stacking or stacked generalization methods and can often outperform any one of the trained sub-models (see e.g. [22][23]). To produce a multivariate prediction $\hat{y}_\Sigma \in \mathbf{R}^m$, the ensemble model is defined as the weighted sum of each prediction $\hat{y}_s \in \mathbf{R}^m$ from each sub-model s, as given by

$$\hat{y}_\Sigma = \sum_{s=1}^{M} \theta_s \hat{y}_s \tag{7}$$

with variable $\theta_s \in \mathbf{R}$, the weighting coefficient of sub-model s where M is the number of sub-models and subscript $s = 1, \dots, M$ indexes the coefficients (i.e. $[\theta_1, \dots, \theta_M] = \theta \in \mathbf{R}^M$). Note that we are not calculating the weighted mean of the sub-models. Therefore, we are not requiring that the values of the weighting coefficients sum to 1 or that the individual weights are positive.

We will employ an Ordinary Least Squares with ℓ_2 Regularization (Ridge) approach for learning the weighting coefficients θ. By solving a quadratic optimization problem, we can identify the weighting coefficients that minimize the error between the observations and the weighted sum of the sub-model predictions. The optimization problem and stacking ensemble model at time step t are given by

$$\theta^\star = \underset{\theta}{\operatorname{argmin}} \sum_{i=1}^{N} \sum_{j=1}^{m} \left(y_{i,j} - \sum_{s=1}^{M} \theta_s \hat{y}_{s,i,j} \right)^2 + \lambda \sum_{s=1}^{M} \theta_s^2 \tag{8}$$

$$\hat{y}_{\Sigma,t} = \sum_{s=1}^{M} \theta_s^\star \hat{y}_{s,t} \tag{9}$$

with variables $\theta^\star \in \mathbf{R}^\mathbf{M}$, the optimal weighting coefficients, $y_i \in \mathbf{R}^\mathbf{m}$, the i-th observed multivariate response, $\hat{y}_{s,i} \in \mathbf{R}^\mathbf{m}$, the i-th prediction from sub-model s, $\hat{y}_{\Sigma,t} \in \mathbf{R}^\mathbf{m}$, the ensemble model prediction at t, and $i = 1, \dots, N$, where N is the number of data samples used for training and m is the length of y_i. Subscript $j = 1, \dots, m$ indexes the $j-$th response. Lastly, λ is a weighting term for the regularization penalty.

E. Stacking Ensemble with Moving Horizon Training

To enable the ensemble model to learn and adapt to changes in the observed system, we will utilize time-varying weights, $\theta_{t,s} \in \mathbf{R}$ for $s = 1, \dots, M$ (i.e. $\theta_t \in \mathbf{R}^M$). These time-varying weights will be calculated by minimizing the error between the observations and the weighted sum of the sub-model predictions over a retrospective moving horizon of data samples. In other words, at each time step t, we will retrain the ensemble model using only the most recent T observations. Therefore, the observations used for calculating the weights θ_t will move with the current time step. The moving horizon optimization problem and stacking ensemble model at time step t are

$$\theta_t^\star = \underset{\theta_t}{\text{argmin}} \sum_{i=1}^{T}\sum_{j=1}^{m}\left(y_{t-i,j} - \sum_{s=1}^{M}\theta_{t,s}\hat{y}_{s,t-i,j}\right)^2 + \lambda\sum_{s=1}^{M}\theta_{t,s}^2 \tag{10}$$

$$\hat{y}_{\Sigma,t} = \sum_{s=1}^{M}\theta_{t,s}^\star \hat{y}_{s,t} \tag{11}$$

with variables $\theta_t^\star \in \mathbf{R}^{\mathbf{M}}$, the optimal weighting coefficients at time step t, $y_t \in \mathbf{R}^{\mathbf{m}}$, the observation at t, $\hat{y}_{s,t} \in \mathbf{R}^{\mathbf{m}}$, the prediction of sub-model s at t, and $\hat{y}_{\Sigma,t} \in \mathbf{R}^{\mathbf{m}}$, the ensemble model prediction at t where T is the number of observations in the moving horizon.

F. Data

For experimentation, this paper considers 2 years of metered hourly electricity demand (kW) data for 8 buildings on the University of California, Berkeley campus. This time-series data has been provided by the facilities team at the University of California, Berkeley and will be used as the observation data for the sub-models and ensemble model. Submetered electricity demand data and building operations data, such a occupancy measurements and mechanical system schedules, were not available. The 8 buildings were selected for their diversity. These buildings include classrooms, offices, libraries, and research facilities. We have also acquired hourly air temperature (°C) and relative humidity (%RH) data from a local weather station [24].

G. Ensemble Learning for Electricity Demand Forecasting

In this paper, we will apply the stacking ensemble model above to the building electricity forecasting problem. Given the many unpredictable behaviors of occupants and the unique physical and mechanical characteristics of every building, a single model approach to electricity demand forecasting may perform very well in one case and very poorly in another. Furthermore, the incorporation of exogenous signals like regional weather conditions may improve a model's accuracy but such benefits cannot be guaranteed. Only through observation and experimentation can the best regression models and input types be identified for a particular building.

By employing our stacking ensemble learning method with moving horizon training technique, we seek to improve the robustness of our electricity demand forecaster. Before we test the ensemble model, we must first train and test the sub-models. In this paper, we will use 8 sub-models, 4 using Ridge and 4 using k-NN. These models will be used to generate short-term multivariate electricity demand forecasts, specifically 6 consecutive hourly electricity demand predictions (i.e. $\hat{y} \in \mathbf{R}^6$).

The regression models will employ 4 different input types or feature sets: electricity demand (D), time (T), electricity demand and time (DT), and electricity demand, time, and exogenous weather data (DTE). Thus, there is 1 Ridge model and 1 k-NN model for each of the 4 input types. The electricity demand input type (D) consists of the 24 hourly records that precede the desired forecast ($x \in \mathbf{R}^{24}$). The time input type (T) is the current weekday and hour represented as a sparse binary vector ($x \in \{0,1\}^{31}$). The demand and time input type (DT) combines the demand and time inputs ($x \in \mathbf{R}^{55}$). The demand, time, and exogenous weather data input type (DTE) is the demand and time input plus current air temperature (°C) and relative humidity (%RH) data retrieved from a local weather station ($x \in \mathbf{R}^{57}$)[24].

In this study, we train a set of 8 sub-models for each building. Training data from one building is not used to fit the models of another building. The sub-models are trained in an off-line batch manner (i.e. trained once on a large dataset) using 18 months of hourly input data from January 1st, 2012, to July 1st, 2013 (i.e. 13,128 training data points). The remaining 6 months of hourly data, July 1st, 2013, to January 1st, 2014 are reserved for testing of the sub-models and the ensemble method (i.e. 4,416 testing data points).

Testing of the stacking ensemble method is done by repeating the following procedure for each time step $t = 1, \ldots, 4416$ where t represents the integer-valued hour between July 1st, 2013, and January 1st, 2014.

1) Using each of the 8 sub-models, generate a 6 hour electricity demand forecast, $\hat{y}_{s,t} \in \mathbf{R}^6$ for $s = 1, \ldots, 8$, as given by either (3) or (6).
2) Learn the model weights $\theta_{t,s}^\star \in \mathbf{R}$ for $s = 1, \ldots, 8$ by minimizing (10) over a moving horizon of the previous $T = 168$ observations (i.e. 7 days)
3) Generate the ensemble model's forecast, $\hat{y}_{\Sigma,t} \in \mathbf{R}^6$, as given by (11).

Once a forecast has been generated for every data point in the testing set, we calculate the errors between the observation y_t and the forecast $\hat{y}_{\Sigma,t}$ for $t = 1, \ldots, 4416$. To evaluate the advantage of our ensemble method over a single model approach, we also calculate the errors between the observation y_t and the sub-model forecasts $\hat{y}_{\Sigma,t}$ for $s = 1, \ldots, 8$ and $t = 1, \ldots, 4416$. To enable the comparison of forecasting error between different buildings, the performance of the sub-models and the ensemble model will be reported as the mean absolute percent error (MAPE),

$$\text{MAPE} = \frac{100\%}{mN}\sum_{i=1}^{N}\sum_{j=1}^{m}\left|\frac{y_{i,j} - \hat{y}_{i,j}}{y_{i,j}}\right| \tag{12}$$

with variables $y_i \in \mathbf{R}^m$, the i-th observation, and $\hat{y}_i \in \mathbf{R}^m$, the i-th prediction, where m represents the number of outputs in the prediction ($m = 6$) and N, the number of predictions.

III. Results

The sub-model and ensemble model performances for each of the 8 building datasets are summarized in Fig. 1. In the figure, the marker color indicates the regression technique used by each model (Ridge, k-NN, or Ensemble) and the marker shape indicates the data type (D, T, DT, DTE, or Ensemble). The sub-model results (Ridge and k-NN) denote the forecast MAPE produced from that particular sub-model, over the testing dataset. The ensemble model results indicate the forecast MAPE produced by minimizing the moving horizon optimization problem and a weighted linear combination of the

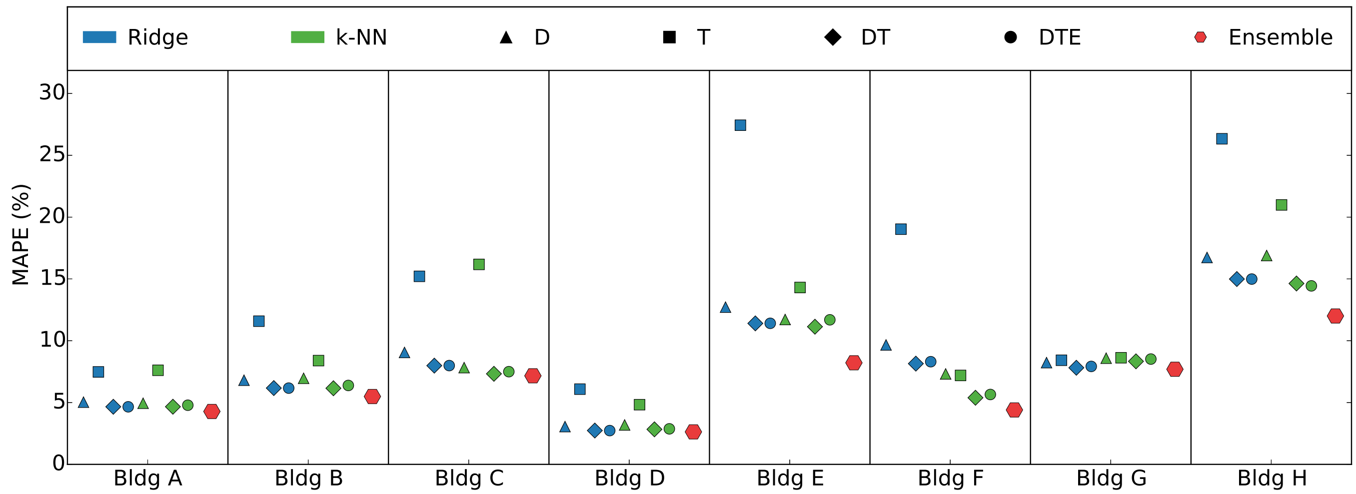

Fig. 1. **Ensemble Model and Sub-Model Performance Results.** The mean absolute percent error (MAPE) of the ensemble model and sub-models of each building for every 6 hour forecast between July 1st, 2013, and January 1st, 2014.

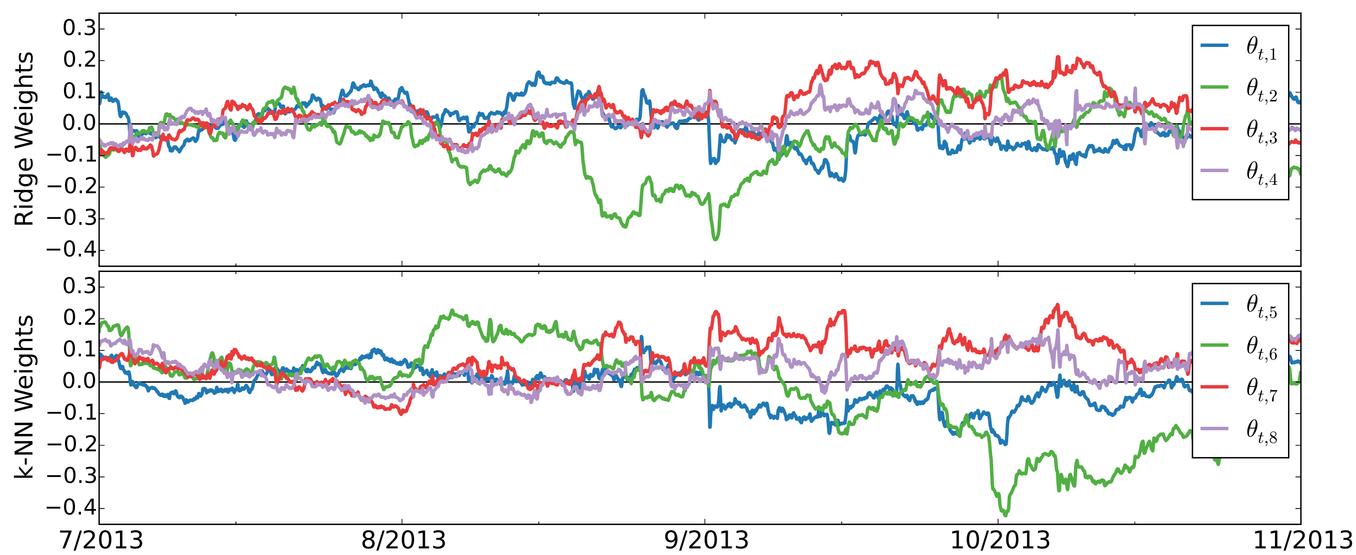

Fig. 2. **Time-Varying Sub-Model Weights.** Examples of time varying weights $\theta_{t,s}$ for building E from July 1st to November 1st, 2013.

sub-model forecasts. Examples of the multivariate electricity demand forecasts $\hat{y}_{\Sigma,t} \in \mathbf{R}^6$ produced by the ensemble model are presented in Fig. 3. Note that the figure plots $\hat{y}_{\Sigma,t}$ starting at but excluding the most recent observation.

By comparing the results in Fig. 1 for each building, we can distinguish sub-models that generally perform poorly (e.g. Ridge and k-NN with T input) from sub-models that generally perform well (e.g. Ridge and k-NN with DT input). We also observe dispersion among the results, particularly in Buildings E, F, and H. This dispersion represents a challenge for building level electricity forecasting. To produce the best results using a single model approach, an engineer must perform model selection for every deployment. This is difficult to scale. Just because a certain regression model and input type has performed well for one building does not guarantee it will do the same for another building.

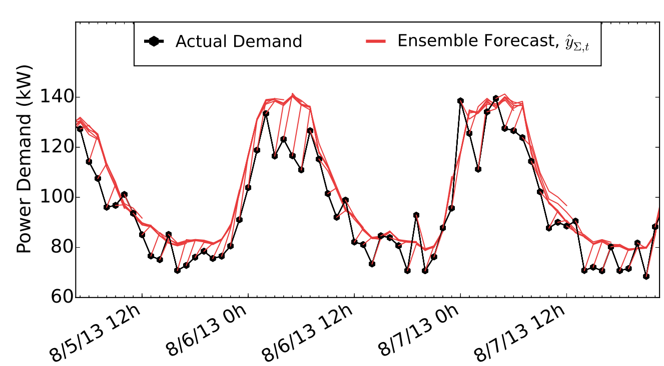

Fig. 3. **Building E Ensemble Forecasts.** Examples of 6 hour electricity demand forecasts for building E using the stacking ensemble learning method.

As indicated by the results, the ensemble model performs comparable to or better than the best sub-model for each building. Therefore, by minimizing the moving horizon optimization problem, the ensemble model is able to (i) learn the sub-model weights in an online manner, and (ii) produce a linear combination of sub-model forecasts that is comparable to or better than the best sub-model forecast. This characteristic is valuable to grid operators and building-level applications. Specifically, engineers need only identify a set of sub-models which generally perform well for demand-side electricity demand forecasting. Then, after training each sub-model on data from a particular building, the stacking ensemble learning method with moving horizon training technique can adaptively identify the weighting of each sub-model for that building.

Fig. 2 presents the sub-model weights $\theta_{t,s}$ of the Building E ensemble model from July 1st to November 1st, 2013. The weights $\theta_{t,1}$, $\theta_{t,2}$, $\theta_{t,3}$, and $\theta_{t,4}$ correspond to the Ridge models with D, T, DT, and DTE input types, respectively. Similarly, the weights $\theta_{t,5}$, $\theta_{t,6}$, $\theta_{t,7}$, and $\theta_{t,8}$ correspond to the k-NN models with D, T, DT, and DTE input types, respectively. As shown, the model weights do not converge but rather continuously evolve in time. Because the weights are determined by minimizing the moving horizon optimization problem, there are trends in the weighting values, but as the training data changes, so do the weights. Of particular note is the sharp change in the parameter values around September 1st, 2013. This can be attributed to the start of the fall academic semester at UC Berkeley and the corresponding change in electricity demand patterns.

IV. Conclusions

This paper presents a stacking ensemble learning method with a moving horizon training approach. We have applied the method to the short-term building-level electricity demand forecasting problem. These results demonstrate enhanced forecasting accuracy across a diversity of buildings due to two features: (i) applying a linear combination of sub-models, and (ii) adaptively learning the stacked model weights in real-time. The practical advantages are notable. Namely, the proposed method enables reliable forecasts over evolving use patterns across a wide diversity of buildings, in contrast to selecting and tailoring a single model for each building.

Additionally, the adaptability provided by the moving horizon training approach enables enhanced control applications. Rather than assuming that demand behaviors are time invariant, the proposed method responds to changes in electricity demand patterns. We have demonstrated this method on 8 buildings' datasets using 8 sub-models each. The results demonstrate that the stacking ensemble method produces equal or better accuracy than single models for multivariate electricity demand forecasts for building-level applications.

References

[1] U.S. Energy Information Administration. (2015, March) Monthly Energy Review. U.S. Department of Energy. [Online]. Available: http://www.eia.gov/mer

[2] E. Hirst and B. Kirby, "Separating and measuring the regulation and load-following ancillary services," *Utilities Policy*, vol. 8, pp. 75–81, 1999.

[3] H. K. Alfares and M. Nazeeruddin, "Electric load forecasting: literature survey and classification of methods," *International Journal of Systems Science*, vol. 33, no. 1, pp. 23–34, January 2002.

[4] G. K. F. Tso and K. K. W. Yau, "Predicting electricity energy consumption: A comparison of regression analysis, decision tree and neural networks," *Energy*, vol. 32, no. 9, pp. 1761–1768, September 2007.

[5] J. G. Jetcheva, M. Majidpour, and W.-P. Chen, "Neural network model ensembles for building-level electricity load forecasts," *Energy and Buildings*, vol. 84, pp. 214–223, December 2014.

[6] F. Javeda, N. Arshada, F. Wallinb, I. Vassilevab, and E. Dahlquistb, "Forecasting for demand response in smart grids: An analysis on use of anthropologic and structural data and short term multiple loads forecasting," *Applied Energy*, vol. 96, pp. 150–160, August 2012.

[7] A. Azadeh, S. F. Ghaderi, S. Tarverdian, and M. Saberi, "Integration of artificial neural networks and genetic algorithm to predict electrical energy consumption," *Applied Mathematics and Computation*, vol. 186, no. 2, pp. 1731–1741, March 2007.

[8] H. S. Hippert, C. E. Pedreira, and R. C. Souza, "Neural networks for short-term load forecasting: A review and evaluation," *IEEE Transactions on Power Systems*, vol. 16, no. 1, pp. 44–55, February 2001.

[9] S. Karatasou, M. Santamouris, and V. Geros, "Modeling and predicting building's energy use with artificial neural networks: Methods and results," *Energy and Buildings*, vol. 38, no. 8, pp. 949–958, Aug 2006.

[10] K. Metaxiotis, A. Kagiannas, D. Askounis, and J. Psarras, "Artificial intelligence in short term electric load forecasting: a state-of-the-art survey for the researcher," *Energy Conversion and Management*, vol. 44, no. 9, pp. 1525–1534, June 2003.

[11] J. Massana, C. Pous, L. Burgas, J. Melendez, and J. Colomer, "Short-term load forecasting in a non-residential building contrasting models and attributes," *Energy and Buildings*, vol. 92, pp. 322–330, April 2015.

[12] P. F. Pai and W. C. Hong, "Support vector machines with simulated annealing algorithms in electricity load forecasting," *Energy Conversion and Management*, vol. 46, no. 17, pp. 2669–2688, October 2005.

[13] A. Kavousi-Fard, H. Samet, and F. Marzbani, "A new hybrid modified firefly algorithm and support vector regression model for accurate short term load forecasting," *Expert Systems with Applications*, vol. 41, no. 13, pp. 6047–6056, October 2014.

[14] W.-C. Hong, "Electric load forecasting by support vector model," *Applied Mathematical Modelling*, vol. 33, no. 5, pp. 2444–2454, May 2009.

[15] H. T. Yang, C. M. Huang, and C. L. Huang, "Identification of ARMAX model for short term load forecasting: An evolutionary programming approach," *IEEE Transactions on Power Systems*, vol. 11, no. 1, pp. 403–408, February 1996.

[16] G. A. Darbellay and M. Slama, "Forecasting the short-term demand for electricity - do neural networks stand a better chance?" *International Journal of Forecasting*, vol. 16, no. 1, pp. 71–83, Jan-Mar 2000.

[17] J. W. Taylor, L. M. de Menezes, and P. E. McSharry, "A comparison of univariate methods for forecasting electricity demand up to a day ahead," *International Journal of Forecasting*, vol. 22, no. 1, pp. 1–16, Jan-Mar 2006.

[18] G. R. Newsham and B. J. Birt, "Building-level occupancy data to improve arima-based electricity use forecasts," in *Proceedings of the 2nd ACM workshop on embedded sensing systems for energy-efficiency in building*. ACM, 2010, pp. 13–18.

[19] S. Rahman and O. Hazim, "A generalized knowledge-based short-term load-forecasting technique," *IEEE Transactions on Power Systems*, vol. 8, no. 2, pp. 508–514, May 1993.

[20] K. Li, H. Su, and J. Chu, "Forecasting building energy consumption using neural networks and hybrid neuro-fuzzy system: A comparative study," *Energy and Buildings*, vol. 43, pp. 2893–2899, 2011.

[21] S. M. Omohundro, *Five balltree construction algorithms*. International Computer Science Institute Berkeley, 1989.

[22] D. H. Wolpert, "Stacked generalization," *Neural networks*, vol. 5, no. 2, pp. 241–259, 1992.

[23] L. Breiman, "Stacked regressions," *Machine learning*, vol. 24, no. 1, pp. 49–64, 1996.

[24] R. L. Snyder, "California irrigation management information system," *American Journal of Potato Research*, vol. 61, no. 4, pp. 229–234, 1984. [Online]. Available: http://www.cimis.water.ca.gov/