Improving Load Forecast Accuracy by Clustering Consumers using Smart Meter Data

Abbas Shahzadeh, Abbas Khosravi, and Saeid Nahavandi
Centre For Intelligent Systems Research
Deakin University

***Abstract*—Utility companies provide electricity to a large number of consumers. These companies need to have an accurate forecast of the next day electricity demand. Any forecast errors will result in either reliability issues or increased costs for the company. Because of the widespread roll-out of smart meters, a large amount of high resolution consumption data is now accessible which was not available in the past. This new data can be used to improve the load forecast and as a result increase the reliability and decrease the expenses of electricity providers. In this paper, a number of methods for improving load forecast using smart meter data are discussed. In these methods, consumers are first divided into a number of clusters. Then a neural network is trained for each cluster and forecasts of these networks are added together in order to form the prediction for the aggregated load. In this paper, it is demonstrated that clustering increases the forecast accuracy significantly. Criteria used for grouping consumers play an important role in this process. In this work, three different feature selection methods for clustering consumers are explained and the effect of feature extraction methods on forecast error is investigated.**

I. Introduction

Having an accurate estimate of the electric load for the next day is imperative for utility companies. Electricity providers and system operators need to know how much electricity is going to be needed at any hour so that they can plan for providing it. Any error in this forecast will result in either increased expenses and unnecessary pollution, or electricity shortage and reliability issues. Trying to avoid these negative consequences is the main reason for the huge interest in improving next day load forecast in the industry and academia.

As a matter of fact, literature on load forecast is extensive and many forecast methods have been applied to this problem. Regression Analysis [1], Artificial Neural Networks [2], Neuro-Fuzzy techniques [3], Fuzzy Modeling [4], Support Vector Machines [5] and Interval Type-2 Fuzzy Logic Systems [6] are among the methods used in this field. Even after decades of research, recent works still report improved performance and lower prediction errors.

However the global roll-out of smart meters has created new opportunities for further improvement of forecast accuracy. Before the mass adoption of smart meters, consumers' electricity usage was typically read in intervals ranging from one to six months. Consequently, usually there was a maximum of 12 readings per year per customer available. However, smart meters, can measure and record each consumer's energy usage, every 15, 30 or 60 minutes. As a result, a large amount of data is now available which was not accessible in the past. This data can be utilized for a variety of purposes. In this paper, the application of this newly emerged data in improving electricity load forecast will be discussed.

As will be explained shortly, the general method for improving the forecast using smart meter data is based on clustering. Assume that the next day electricity demand of a city with a number of consumers is to be predicted for the next 24 hours. A model can be trained to generate a forecast for the next day, using inputs like temperature, load at the same hour on the current day, load at the same hour on the same day in the last week, etc. Any of the methods mentioned above can be used for creating such a model.

Nonetheless, instead of developing and training a single model for the aggregated load of all of the consumers, they can be clustered in a number of groups. A model can be trained for each group. Thus a prediction model for the total load of each group can be developed. By adding the outputs of these models, a new prediction for the load of the whole city will be calculated. Then a comparison between the forecast error of the single model method and multiple cluster method will show the effectiveness or pointlessness of clustering.

The research about using smart data for improving load prediction is still in its early stages. As a result, there are not many papers addressing this issue. The existing works can be categorized according to their feature set selection methods, clustering methods and the applied prediction methods. Alzate and Sinn in [7] use wavelet analysis for feature extraction, a method called Kernel Spectral Clustering for clustering and a prediction method called PARX as their forecast method. They report a more than 20% improvement in forecast accuracy using these techniques.

Ilic et al. in [8], do not use a clustering method. Instead they first choose a group size and then randomly pick consumers for each group. By varying the group size and performing load forecast, they conclude that forecast accuracy increases for larger groups. However they do not compare the results of aggregated load forecast and load forecast after grouping the consumers. One odd observation in their paper is that complicated forecast methods work only slightly better than very simple prediction methods.

Misiti et al. in [9] use a three step strategy. First they preprocess each consumer's load data using wavelets. Then they divide the consumers into multiple clusters. Lastly they merge clusters to create larger clusters based on a set of criteria. They conclude that the best results are usually achieved with a rather small number of clusters. In their case study they show that the optimum results were obtained when 19 clusters were used.

Clustering based improvement of load forecast has many aspects that need to be studied. A very important factor in clustering is the set of criteria used for grouping the consumers. In this paper, three different sets of criteria will be discussed and the performances of the models based on these criteria will be compared. Number of clusters is also an important factor. However the main contribution of this paper is to show the effect of feature selection methods on the performance of forecast. Thus a fixed number of clusters has been assumed.

The rest of this paper is organized as follows. In section II, the dataset used for testing different methods in this paper is introduced. In section III, the problem is formulated and the concept of clustering is explained. In section IV, a single model for forecasting the aggregated load is introduced and the results are reported to be used as a benchmark for comparison. In section V, three different feature selection methods are explained in detail. In Section VI, the introduced feature selection methods are applied on the dataset and the forecast results are compared. Finally, section VI concludes the paper.

II. Smart Meter Dataset

The dataset used in this paper contains smart meter records of more than 6000 consumers in Ireland. This data was gathered by Commission for Energy Regulation (CER) and was Accessed via the Irish Social Science Data Archive (ISSDA)[1].

The smart meter readings have been recorded from 14 July 2009 to 31 December 2010. There are two groups of consumers in this dataset: residential consumers and small to medium enterprises. In this paper, only the data of residential consumers is used. The data needed cleaning due to a large number of missing records for some consumers. Therefore, in this research, first the consumers with a large number of successive missing records were omitted. Then for single missing points, linear interpolation between the previous and next records was used. After the cleaning process, a total of 3176 consumers remained for analysis. One other initial problem with the dataset was the very large size of data files. After the cleaning phase, one file was created for each consumer in order to make the size of data files, manageable.

It should be noted that the dataset is completely anonymized and unfortunately no information about the locations of consumers in Ireland is given. Therefore, local temperature and weather data cannot be used. Instead, similar to [7], temperature and dew point records from Dublin airport were used for all consumers. It was assumed that all consumers experienced the same weather conditions. This assumption is not accurate but due to the lack of data, this was the only possible workaround.

III. Problem Statement

Smart meter records in the dataset are given at half hourly intervals. Consequently there are 48 records per day. Equation (1) shows the very simple relation between number of days (d) and number of time intervals (n).

$$n = 48 \times d \tag{1}$$

[1]www.ucd.ie/issda

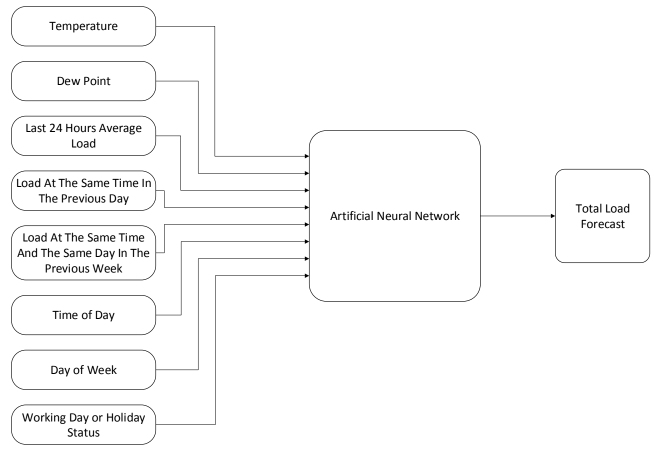

Fig. 1. Model used for generating the forecast for the total aggregated load of all consumers

Equation (2) shows the time series vector that represents energy consumption of consumer i during the period of interest. Henceforth, vectors will be displayed in bold and scalar values in plain font.

$$\mathbf{l}_i = \{l_i^1, l_i^2, ..., l_i^n\} \tag{2}$$

In (2), n denotes number of time intervals in the period of interest from (1). Total aggregated load can be displayed in a similar form as given in (3).

$$\mathbf{L}_{total} = \{L_{total}^1, L_{total}^2, ..., L_{total}^n\} \tag{3}$$

As stated in (4), at each time interval t, aggregated load of the network is the sum of energy consumption of all consumers.

$$L_{total}^t = \sum_{i=1}^{M} l_i^t \tag{4}$$

In (4), M denotes the number of consumers in the network. The goal of this paper is to generate a forecast for the aggregated load in the period of interest. A model like the one depicted in figure 1 can be used for this purpose. This model will be introduced in the next section. The forecast is again a vector with n elements for the n time intervals in the period of interest and is defined in (5).

$$\hat{\mathbf{L}}_{total} = \{\hat{L}_{total}^1, \hat{L}_{total}^2, ..., \hat{L}_{total}^n\} \tag{5}$$

For measuring the accuracy of forecast, Mean Absolute Percentage Error (MAPE) is used in this work. MAPE is defined in (6).

$$MAPE = \frac{100}{n} \sum_{t=1}^{n} \frac{\hat{L}_{total}^t - L_{total}^t}{L_{total}^t} \tag{6}$$

Figure 2 shows the concept of forecast using clustering. The main idea of improving forecast by clustering is to divide the entire population of consumers into C clusters namely S_1, S_2, ..., S_C and then generate a forecast for each cluster. The forecast for a cluster S_k is given in (7) below.

$$\hat{\mathbf{L}}_{S_k} = \{\hat{L}_{S_k}^1, \hat{L}_{S_k}^2, ..., \hat{L}_{S_k}^n\} \tag{7}$$

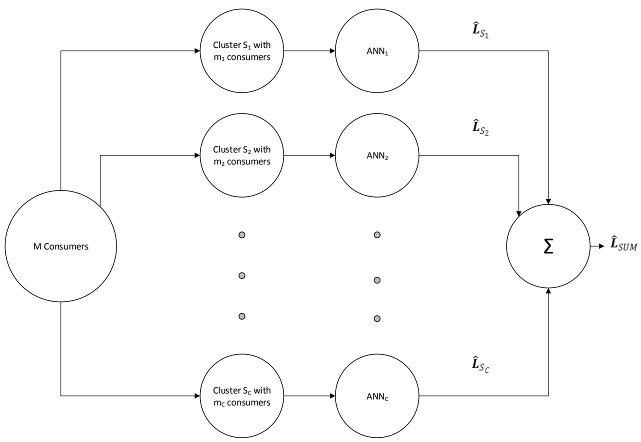

Fig. 2. The main concept of forecast using clustering. Instead of using only one model, C artificial neural network models are used for clusters S_1 to S_C

Each cluster will have a number of consumers in it. Number of consumers in any S_k is denoted by m_k. Adding up all m_k values will give the total number of consumers defined as M:

$$m_1 + m_2 + ... + m_C = M \tag{8}$$

A forecast for the total load can be generated by adding up all $\hat{\mathbf{L}}_{S_k}$ forecasts as formulated in (8).

$$\hat{\mathbf{L}}_{Sum} = \sum_{k=1}^{C} \hat{\mathbf{L}}_{S_k} \tag{9}$$

$\hat{\mathbf{L}}_{Sum}$ denotes the forecast for the total aggregated load using clustering method. A MAPE value can be calculated for this new forecast. A comparison between forecast error of $\hat{\mathbf{L}}_{Sum}$ and that of $\hat{\mathbf{L}}_{total}$, can show the improvement or the deterioration of forecast using clustering method.

IV. Forecast Using a Single Model

The model depicted in figure 1 was used to perform the forecast for the total aggregated load of all 3176 consumers. The model used in this work is based on a model developed and introduced in [10] and [11]. The Artificial Neural Network (ANN) in this model, is the default two-layer feed forward network in a MATLAB tool called nftool (the 2014a version of MATLAB was used for this work). This network uses sigmoid neurons in the hidden layer and linear neurons in the output layer. In this work, 20 neurons were used in the hidden layer. The network was trained using Levenberg-Marquardt back-propagation algorithm which was again the default algorithm in the MATLAB neural network fitting tool.

In this model a total of eight variables are utilized as the independent input variables to the forecast model. A regression analysis showed that among these variables, "load at the same hour in the previous day" and "load at the same hour in the same day in the last week" had the highest correlation with the actual load. For example if a prediction for the total load at 10:00 AM Wednesday 22 February 2010 is required, load at 10:00 AM on 21 February 2010 and load at 10:00 AM on 15 February 2010 will be used. Being a work day or a holiday also has a significant effect on the load pattern.

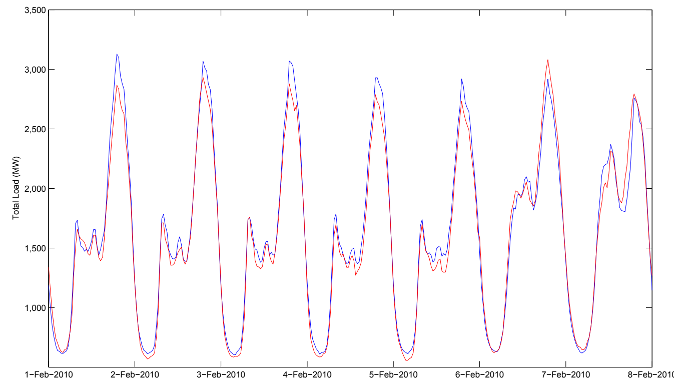

Fig. 3. Forecasted load profile (red) versus actual load (blue) for the week in question

It should be noted that in the initial model, only six parameters were used for generating the forecast. Later, temperature and dew point data were added to the model. Since local temperature data is not accessible for this dataset, there is a very small correlation between weather data and load. However adding these parameters decreased the forecast error by about 0.4%.

Total load data from 1 February 2010 to 7 March 2010 was used for training and validating the network. Then the model was used for generating a 24 hour ahead forecast for the second week of March 2010. Actual load and a prediction generated by the model, are plotted in figure 3. As it can be seen in this figure, the forecast follows the actual load pattern closely. It should be noted that the actual load figures were used in this case and no normalization was carried out. Training and forecast procedure were performed 150 times to give a better performance measure of the model. In almost all of the trials, the training algorithm converged in less than 30 iterations. Forecast error was calculated using (6). Average forecast error using this model is 6.1479%. As mentioned earlier 20 neuron were used in the hidden layer of the ANN. Increasing the number of neurons did not result in any improvements in the accuracy of the forecast. The boxplot graph of the forecast error results of these 150 trials is given in figure 6. The boxplot corresponding to this method is the first bar from the left. The other five methods in that figure will be described in the two following sections.

V. Feature Selection Methods

In forecast by clustering method, at the first step, consumers should be divided into a number of clusters. Clustering algorithms need a vector of features to be assigned to each consumer. In this section three different methods for feature extraction will be discussed. In each of these methods a vector like $\mathbf{v}_i$ in (10) with a number of elements is assigned to each consumer like consumer i.

$$\mathbf{v}_i = \{v_i^0, v_i^1, ..., v_i^p\} \tag{10}$$

In (10), p shows the number of elements in the feature vector. $\mathbf{v}_i$ will be referred to as the "Feature vector" in the rest of this paper. The difference between the three methods that will be explained shortly is in the method that is used to extract feature

TABLE I. INDEPENDENT VARIABLES USED FOR REGRESSION ANALYSIS

Symbol	Description
x_1	Temperature
x_2	Dew Point
x_3	Average load in the previous 24 hours period
x_4	Load at the same hour in the previous day
x_5	Load at the same hour in the same day of week in the previous week
x_6	Time of day (1-48)
x_7	Day of week (1-7)
x_8	1 for working days, 0 for holidays and weekends

vectors from the data available for each consumer. This process is often referred to as feature extraction in the literature.

A. Regression Coefficients

As mentioned earlier, a total of 8 independent variables have been used as inputs to the forecast model(s). These eight variables are defined in more detail in table I.

A regression analysis of the electric load pattern of any consumer, will give eight regression coefficients corresponding to each of these eight variables. A load forecast at a time t can be generated for each consumer i as formulated in (11).

$$\begin{aligned}\hat{l}_i^t = b_i^0 + b_i^1 \times x_1(t) + b_i^2 \times x_2(t) + b_i^3 \times x_3(t) \\ +b_i^4 \times x_4(t) + b_i^5 \times x_5(t) + b_i^6 \times x_6(t) \\ +b_i^7 \times x_7(t) + b_i^8 \times x_8(t)\end{aligned} \tag{11}$$

In (11), b_i^0 to b_i^8 are the regression coefficients for consumer i. These coefficients are different for each consumer and can be used as a set of features for clustering. For this purpose, a vector with 9 elements can be assigned to each consumer as in (12).

$$\mathbf{v}_i = \{b_i^0, b_i^1, b_i^2, b_i^3, b_i^4, b_i^5, b_i^6, b_i^7, b_i^8\} \tag{12}$$

These regression coefficients reflect the correlation between the load and each of the eight independent variables introduced in table I. In other words, the coefficients in (12), indicate the sensitivity of the load to each of the affecting factors like temperature, day of week, etc. Thus clustering consumers using these vectors seems very reasonable. As a matter of fact, during the training of a neural network, using an algorithm like Levenberg-Marquardt back-propagation, parameters of the network are tuned to better model the relationship between each of inputs and the output. Thus if a network can be trained for consumers with similar responses to input variables, a higher overall accuracy can be expected.

B. Average Daily Load Pattern

The second method that was used for feature extraction is based on using average daily load pattern. The feature vector for each consumer will have 48 elements corresponding to the 48 half hourly time intervals in a day. Feature vector for consumer i is given in (13).

$$\mathbf{v}_i = \{al_i^1, al_i^2, ..., al_i^{48}\} \tag{13}$$

In (13) each al_i^k is the average load at time interval k of the day for consumer i and is calculated using (14).

$$al_i^k = \frac{1}{d} \sum_{j=0}^{j=d-1} l_i^{k+48\times j} \tag{14}$$

In (14), d denotes number of days in the training period.

C. Full Load Pattern

The third method that was tested for feature extraction is basically the simplest method in terms of mathematics. The feature vector in this case is merely consisted of smart meter records in the training period as stated in (15) where n denotes number of time interval in the training period.

$$\mathbf{v}_i = \{l_i^1, l_i^2, ..., l_i^n\} \tag{15}$$

One major drawback of this method of feature extraction is that feature vectors will have large dimensions. For example for a 30 day training period, each $\mathbf{v}_i$ will have $30 \times 48 = 1440$ elements which is a very large number in comparison to 48 elements for average daily load pattern and only 9 elements in sensitivity method. Larger dimensions for feature vectors will require longer times for the clustering algorithm.

VI. CASE STUDY

In this section the three methods explained in the previous section will be applied to the dataset introduced in section II.

A. Regression Coefficients

First a regression analysis was performed on the given dataset to find the regression coefficients for each consumer as explained in the previous section. The calculated coefficients were very different for each independent variable. Regression coefficients for x_1 which is the symbol for temperature variable ranged from -0.0015 to 0.0071 with a mean value of 0.002. While regression coefficients corresponding to x_8 for different consumers varied between 0 to 0.9454 (except for a few outliers) with an average value of 0.1665. As a result any clustering using these coefficients would be heavily dominated by b_8 values (which are regression coefficients corresponding to x_8). To prevent this problem a normalization preprocessing was carried out on the regression coefficients. Normalization can be carried out using various techniques. Two of the most common techniques were examined in this work. The first method is based on calculating standard score and is formulated in (16).

$$b_i'^k = \frac{b_i^k - \mu_k}{\sigma_k} \tag{16}$$

In which, μ_k and σ_k are mean value and variance of all b_i^k values for $i = 1...n$ respectively. The second normalization method that was tried in this study is given in (17). This method transforms the values to figures in the [0,1] range so that the minimum value is mapped to 0 and maximum value is mapped to 1.

$$b_i'^k = \frac{b_i^k - \min_i(b_i^k)}{\max_i(b_i^k) - \min_i(b_i^k)} \tag{17}$$

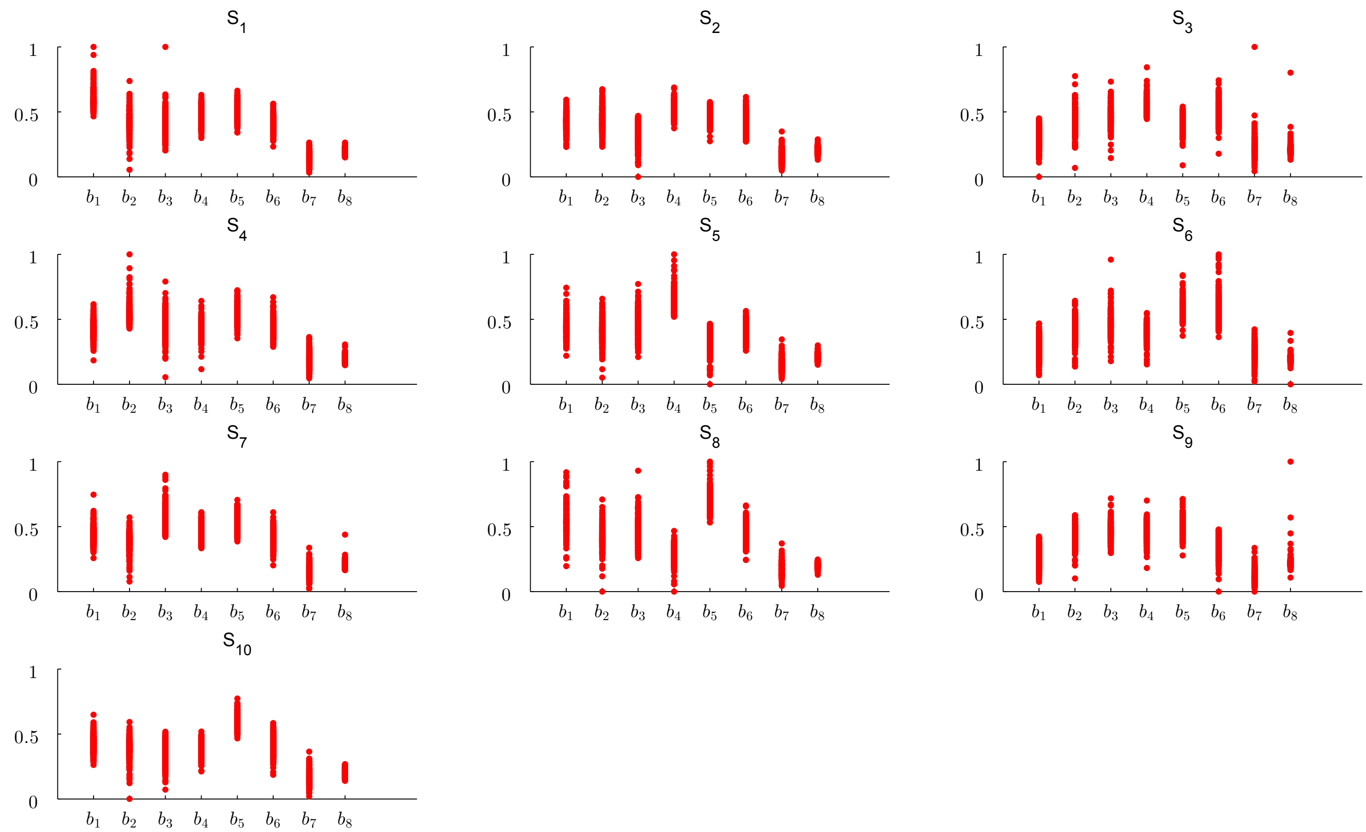

Fig. 4. Regression coefficients for clusters 1 to 10

Overall three sub-methods in respect to normalization were used to find the most efficient one:

1) No normalization.
2) Normalization using standard score (equation (16)).
3) Normalization by mapping to [0,1] range (equation (17)).

It turned out that the second normalization technique (sub-method 3) gives the best results. The results of applying the k-means clustering algorithm to the features extracted using this method are reported in figure 4. A comparison between clusters S_5 and S_8 in this figure, can give a better understanding of the clustering concept using regression coefficients as the feature set. According to the results portrayed in figure 4, in S_5, b_4 values are large and b_5 values are small. On the contrary in S_8, b_4 values are small and b_5 values are large. This means that for users in the 5$^{\text{th}}$ cluster, there is a high correlation between load at a time interval with the load at the same hour in the previous day (x_4 variable in table I) but little correlation with the load at the same hour and same day of week in the previous week (x_5 variable in table I). The other way around is true for the 8$^{\text{th}}$ cluster.

B. Average Daily Load Pattern

Average load patterns were calculated for each consumer. To make the results comparable, each load pattern was normalized by dividing the feature vector to the largest element in the vector as formulated in (18).

$$\mathbf{v}_i^{'} = \frac{\mathbf{v}_i}{\max(\mathbf{v}_i)} \tag{18}$$

Then k-means clustering algorithm was applied. To obtain better results, the algorithm was run 100 times on the collection of feature vectors and the best clustering choice was selected. The measure used for evaluating the quality of clustering is the sum of distances between cluster centres and the members of clusters. The results of the clustering phase are given in figure 5.

C. Full Load Pattern

Preparing the feature vectors for this method is easier than the other two methods. However, unlike the other two methods, clustering took a very long time because of high dimensionality of the feature vectors. Training and validation periods are 4 and one week long respectively. As a result, each feature vector has 35 days or according to (1), 1680 elements in it. Consequently running 100 iterations of k-means algorithm was not practical in this case and only 5 iterations were used. Figure 6 and table II show the forecast errors of the methods introduced earlier.

According to figure 6 and table II, clustering consumers using the regression coefficients normalized between 0 and 1, gives the highest forecast accuracy. This clustering method results in about 35% reduction in forecast error in comparison to the single model method.

One interesting observation is that no matter which clustering method is applied, forecast accuracy always is higher than

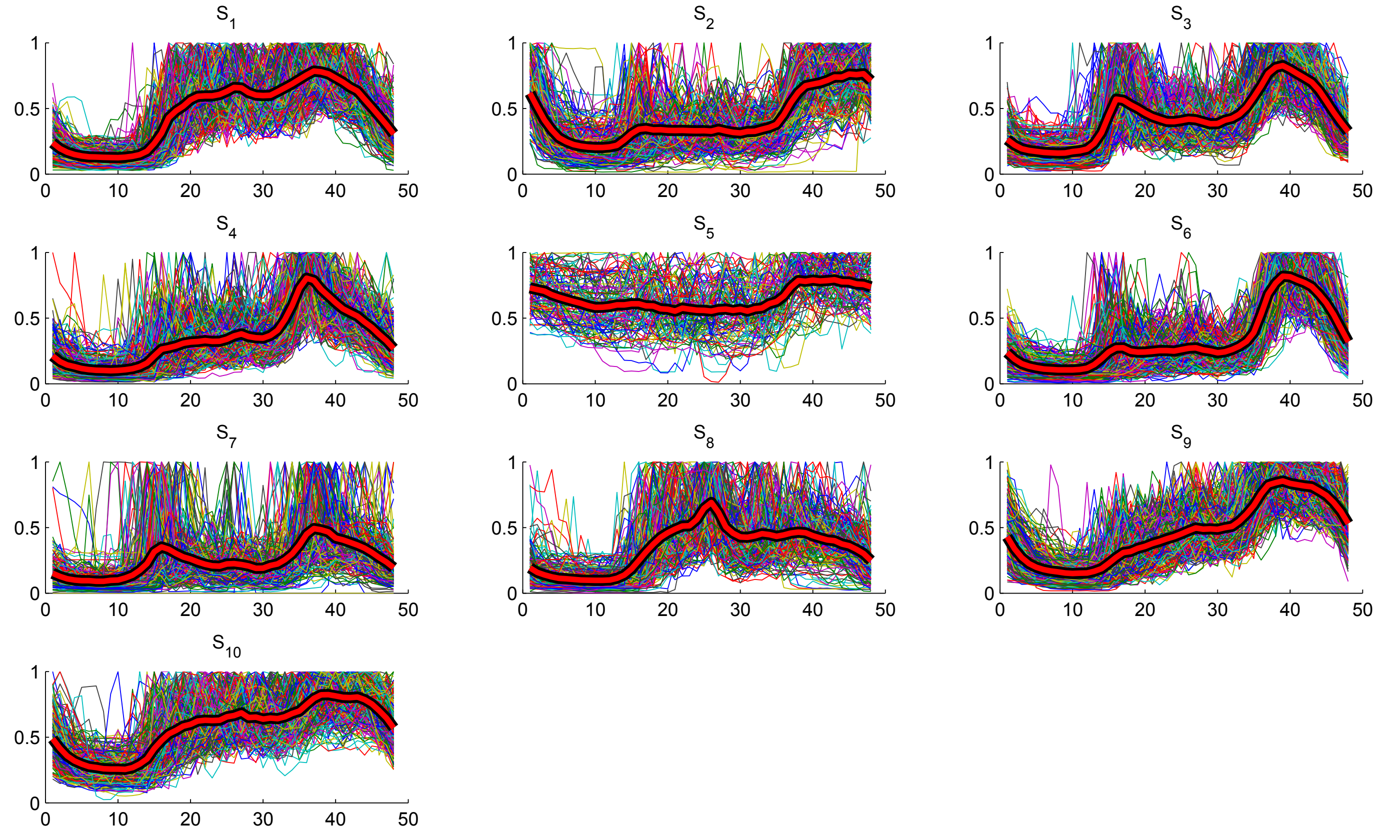

Fig. 5. Average load patterns for clusters 1 to 10. Consumers with similar average load patterns are put in the same cluster. The thick red line in each subplot shows the mean of all of the average load patterns in the corresponding cluster.

TABLE II. MEAN ABSOLUTE PERCENTAGE ERROR FOR DIFFERENT FORECAST AND CLUSTERING METHODS

Method	**MAPE**
Single Neural Network Model	6.1479
Clustering Based on Regression Coefficients No Normalization	4.5234
Clustering Based on Regression Coefficients Normalized using Equation (17)	3.9688
Clustering Based on Regression Coefficients Normalized using Equation (16)	4.2368
Clustering Based on Average Load Pattern	4.4594
Clustering Based on Full Load Pattern	4.7475

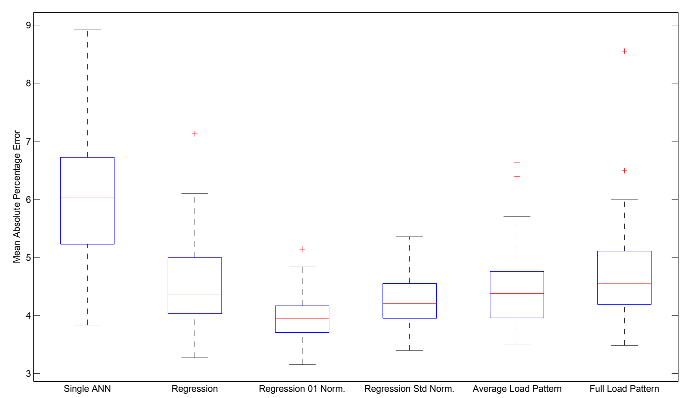

Fig. 6. Boxplot Diagram For Methods Listed in Table II. For each method, the experiment was run 150 times and the graph shows the distribution of the results. The red lines show the median of each set, the blue rectangle shows the range between the first and the third quartile. The whiskers (black line segments) mark the area in which about 98% of the results are lying and the red plus markers show the outliers.

the case of a single ANN model. It was observed that the single ANN model cannot provide the same accuracy regardless of the number of neurons used in the hidden layer. However the computation load is higher in comparison with when the single ANN model is used, since the training phase has to be repeated for each cluster separately. The improvement in the quality of forecast is a result of the introduction of new data which is the smart meter records of a large number of consumers. Intuitively, by using more data, higher accuracy can be expected. The validity of this expectation is confirmed by the experiment results reported in this work.

VII. CONCLUSION

Three different feature selection methods for clustering were examined in this paper. For one of these methods two different normalization techniques were also tried. It was demonstrated that clustering increases the forecast accuracy in all of the studied methods. However using normalized regression coefficients yields the best results. Forecast error decreased by 35% in comparison to a single ANN model. In a future work, other factors affecting the accuracy of forecast like number of clusters, forecast model, etc. will be investigated.

REFERENCES

[1] A. Papalexopoulos and T. Hesterberg, "A regression-based approach to short-term system load forecasting," *IEEE Transactions on Power Systems*, vol. 5, no. 4, pp. 1535–1547, Nov. 1990.

[2] D. Park, M. El-Sharkawi, I. Marks, R.J., L. Atlas, and M. Damborg, "Electric load forecasting using an artificial neural network," *IEEE Transactions on Power Systems*, vol. 6, no. 2, pp. 442–449, May 1991.

[3] K.-H. Kim, J.-K. Park, K.-J. Hwang, and S.-H. Kim, "Implementation of hybrid short-term load forecasting system using artificial neural networks and fuzzy expert systems," *IEEE Transactions on Power Systems*, vol. 10, no. 3, pp. 1534–1539, Aug. 1995.

[4] P. Mastorocostas, J. Theocharis, and A. Bakirtzis, "Fuzzy modeling for short term load forecasting using the orthogonal least squares method," *IEEE Transactions on Power Systems*, vol. 14, no. 1, pp. 29–36, Feb. 1999.

[5] 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, 2005.

[6] A. Khosravi, S. Nahavandi, D. Creighton, and D. Srinivasan, "Interval type-2 fuzzy logic systems for load forecasting: A comparative study," *IEEE Transactions on Power Systems*, vol. 27, no. 3, pp. 1274–1282, Aug. 2012.

[7] C. Alzate and M. Sinn, "Improved electricity load forecasting via kernel spectral clustering of smart meters," in *Proceedings - IEEE International Conference on Data Mining, ICDM*, 2013, pp. 943–948.

[8] D. Ili, S. Karnouskos, P. G. Da Silva, and M. Jacobi, "Impact assessment of smart meter grouping on the accuracy of forecasting algorithms," *IEEE Transactions on Smart Grid*, pp. 673–679, 2013.

[9] M. Misiti, Y. Misiti, G. Oppenheim, J.-M. Poggi, and others, "Optimized clusters for disaggregated electricity load forecasting," *Revstat*, vol. 8, pp. 105–124, 2010.

[10] A. Deoras, "Electricity load and price forecasting webinar case study." [Online]. Available: http://au.mathworks.com/matlabcentral/fileexchange/28684-electricity-load-and-price-forecasting-webinar-case-study

[11] Ameya Deoras, MathWorks, "Electricity load and price forecasting with matlab." [Online]. Available: http://au.mathworks.com/videos/electricity-load-and-price-forecasting-with-matlab-81765.html