Power Prediction through Energy Consumption Pattern Recognition for Smart Buildings

Ming Jin, Lin Zhang, Costas J. Spanos

Abstract— **In this paper, we propose a Non-negative Mixture of Experts (NME) model for smart buildings that is capable of making accurate power forecasting by recognizing characteristic consumption patterns. The model uses prediction error as a metric to guide the feature learning process subject to non-negativity constraints. The objective is to understand and model energy consumption behaviors in commercial buildings at the appliance level so as to facilitate dynamic pricing and demand response. Application of the NME model to a large dataset of device power measurements results in the discovery of meaningful energy usage patterns that are characteristic of the working and idle states of the building space, with the additional advantage that the learned features also optimize the energy prediction model. The model can be learned by stochastic gradient descent, which is suitable for large-scale problems, and an online version is also suggested.**

I. INTRODUCTION

Energy consumption relates to environmental issues, economic growth, and national security. Commercial and residential buildings account for 40% of the energy consumed in the US, compared with just 25% for transportation [1]. The huge potential of energy saving drives a growing interest in Energy Information Systems (EIS) that analyzes time-series data from meters, sensors, and external sources to perform anomaly detection, electric power tracking and prediction, efficiency evaluation, and load shaping to accommodate advanced demand-response schemes [2].

The ability of making reliable energy forecasts is crucial for future smart grids to implement demand-response (DR) schemes so as to detect potential demand-supply mismatch and ensure the overall stability of the power system. Electric energy forecasts for sensor-rich smart buildings, which are integral parts of the smart grid system, therefore plays a critical role in demand-side management and the implementation of dynamic pricing and other load curtailment strategies [3]. Amin-Naseri and Sorroush proposed a hybrid neural-network model based on a self-organizing map integrated with a feed-forward neural network to predict daily energy consumption of buildings [4]. Data-driven approaches were also investigated using Artificial Neural Networks (ANN) and Support Vector Machines (SVM) [5], [6], [7].

There is also much to learn about the underlying social structures and working patterns in the building space from the rich sensor data. As put forth by Nobel Laureate Kahneman in his book "Thinking, fast and slow", people are much more habitual and patterned than just acting randomly [8]. Recognizing the pattern of energy consumption from the non-intrusive device measurement, for instance, can help us understand the occupants and their behaviors while circumventing privacy issues [9]. It will lead to improvement of the energy usage efficiency and make the building "smarter" to understand and interact with its occupants.

It is, therefore, the objective of this paper to investigate the Non-negative Mixture of Experts (NME), which imposes practical constraints on the unsupervised learning process that is guided by some meaningful metric. Specifically, we apply the NME model to the sensor data collected in a study to understand the building's energy consumption behaviors, and demonstrate its capability of predicting future power consumption. In addition, we recognize interesting patterns that point to the underlying energy usage behaviors.

This paper is organized as follows. In Section II, we motivate the application of NME model and review relevant methods, including the Mixture of Experts model and non-negative matrix factorization. Section III introduces the NME model and its learning algorithm based on gradient descent rules. In Section IV, we apply the model to device measurement data and discuss the prediction and pattern recognition results. The paper is concluded in Section V with a discussion about possible future extensions.

II. RELATED WORK

Buildings can operate in several modes depending on the occupants activities, which make it difficult to perform prediction with a single model. The NME model improves traditional prediction methods by introducing a switching mechanism, which follows the intuition that buildings' operation modes can be observed through plug-load level consumption. Figure 1 (a) illustrates the working principle of the model: first, relevant information is fed into the first layer to decide which expert we should consult. Then each expert gives its opinion based on its own judgment, and the final decision is a weighted version of all the experts.

We now give a brief review of relevant works in the past that motivate the development of the NME model.

A. Mixture of Experts Model

One class of relevant models include the Mixture of Experts (ME) and the Hierarchical Mixture of Experts (HME), which are ensemble method that utilize a combination of simple learners to improve predictions [10], [11]. There are

*Author affiliations: M. Jin and C. J. Spanos are with the Department of Electrical Engineering and Computer Sciences at the University of California Berkeley, USA. Emails: {jinming, spanos}@berkeley.edu. L. Zhang is with the Department of Electronic Engineering at Tsinghua University, China. Email: linzhang@mail.tsinghua.edu.cn

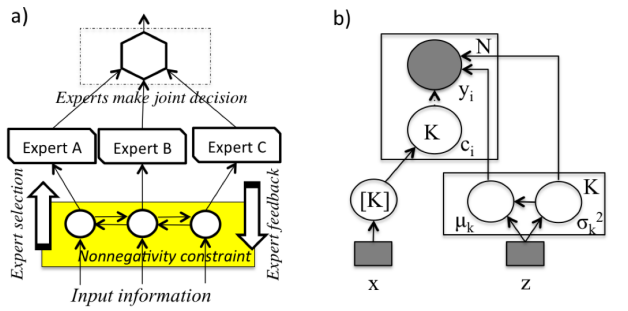

Fig. 1. a). Illustration of the NME model (left). b). Plate notation of the representation as a mixture model (right).

two basic components of the model: the gating network and the expert network. The gating network decides which expert to go to with the gating softmax functions:

$$g_t(\mathbf{x};\eta) = \frac{\exp\left(\eta_i^\top \mathbf{x}\right)}{\sum_{j=1}^{k}\exp\left(\eta_j^\top \mathbf{x}\right)} \tag{1}$$

The expert network gives its estimates of the query by maximizing the log-likelihood function obtained for the mixture probability. The learning of the model is usually achieved with EM algorithm, by augmenting the likelihood with a new variable, Z, an indicator variable that determines which expert to go to. The expectation of this indicator function can be shown to be the posterior probability of the expert.

The major drawback of the ME model is that the learning of the gating network parameters does not take into account constraints of data, such as the constraint of non-negativity on the learned parameters. Also the softmax function does not promote the discrepancy among different gating parameters, and can be difficult to optimize, which can also result in difficulty in explaining the gating parameters. Nevertheless, ME and HME models tackle the nonstationarity problem quite successfully by considering different input regimes.

B. Non-negative Matrix Factorization

Many algorithms for matrix factorization and dimension reduction have been studied, including principle component analysis (PCA) and singular value decomposition (SVD). Their common goal is to use as few features as possible to account for the most variance presented in the matrix. Non-negative matrix factorization (NMF), popularized by Lee and Seung, is a class of methods that are designed for matrices with only non-negative entries [12]. The method imposes a constraint on the factor matrices to be non-negative:

$$\min_{W,H} V = WH, \quad s.t.\ H_{ij} \geq 0, W_{ij} \geq 0 \tag{2}$$

This simple constraint leads to many interesting discoveries, such as discovery of metagenes, text mining, and spectral data analysis [13].

In the context of device power measurement data from buildings, such non-negativity constraint arises naturally, which suggests that we can use NMF to find possible building states. However, one problem with this approach is that the factor matrices are generally non-unique, lacking a metric to guide the learning process to generate meaningful patterns. Next we will show that the proposed NME model can deal with this issue effectively.

III. Non-negative Mixture of Experts Model

A. Model formulation

We set out the formulation of the NME model with two objectives: first, we want to train a model that makes accurate predictions of the energy consumption based on all the available data. Also, we want to learn interesting and useful patterns of device usage in the building, which help us understand the occupancy behaviors and possibly characterize the building state. Since the NME can be regarded as a derivative of the ME model with non-negative constraints, it is convenient to just adopt the terminologies, such as the gating and expert networks, introduced in the ME model.

The approach is a joint optimization over the gating and expert network parameters:

$$\begin{aligned} \underset{A,H,R}{\text{minimize}} \quad & \sum_{i=1}^{N}\left\| y_i - g\left(X^{(i)};A,H\right)^\top Rz^{(i)}\right\|^2 \\ \text{subject to} \quad & H \geq 0 \quad \text{(non-negativity constraint)} \end{aligned} \tag{3}$$

The objective function, $F(A,H,R;y,X,z)$, is essentially a quadratic cost function evaluating how well the model can represent the data with relevant information. A,H denote the model's gating parameters, R denotes its expert parameter, y_i is the target value, and $X^{(i)},z^{(i)}$ are relevant information, which are not necessarily of the same nature of y_i. For instance, y_i can be the next-moment power, while $X^{(i)},z^{(i)}$ can be the device power consumption data and previous energy data respectively. This formulation extends the model's capacity to include all possible relevant information.

The gating function is given by normalized linear terms:

$$g\left(X^{(i)};A,H\right) = \left[\frac{\left(H^\top X^{(i)}A^\top\right)_1}{\sum_{j=1}^{k}\left(H^\top X^{(i)}A^\top\right)_j},\cdots,\frac{\left(H^\top X^{(i)}A^\top\right)_k}{\sum_{j=1}^{k}\left(H^\top X^{(i)}A^\top\right)_j}\right]^\top \tag{4}$$

where $\left(H^\top X^{(i)}A^\top\right)_1 = \sum_{j=1}^{m} A_{1,j}\left\langle X^{(i)}_{\cdot,j}, H_{\cdot,j}\right\rangle$ is a weighted sum of the dot products of the input vectors, $X^{(i)}_{\cdot,j}$, with the 1^{st} column of the gating parameter $H_{\cdot,j}$. This form of gating function is amenable to gradient-descent methods, and is also easier to interpret: each component corresponds to the weight applied to each expert's opinion, given by:

$$Rz^{(u)} = \left[R_{1,\cdot}z^{(i)},\cdots,R_{p,\cdot}z^{(i)}\right]^\top \tag{5}$$

where each element in the vector is the opinion of the corresponding expert.

The structure of the model is best presented with a layered structure, similar to the artificial neural networks.

Layer 1: *Expert weight calculation.* In this layer, the absolute weights of each expert, ω_i, is given by:

$$\omega_p = \left(H^\top X^{(i)}A^\top\right)_p = \sum_{j=1}^{m} A_{1,j}\left\langle X^{(i)}_{\cdot,j}, H_{\cdot,p}\right\rangle \tag{6}$$

The inner product term dictates that *the closer the input information is to the feature vector $H_{\cdot,p}$, the higher the weight assigned to the corresponding expert*.

Layer 2: *Expert weight normalization*. In this layer, the weights are normalized so that they sum to 1, which is essentially the gating function in (4).

Layer 3: *Expert opinion formulation*. This layer is as simple as a linear regression for each expert given the input information, as given in (5).

Layer 4: Output layer, where we weight all the expert opinions to give a final result regarding the query:

$$\hat{y}_i = g\left(X^{(i)}; A, H\right)^\top R z^{(i)} \tag{7}$$

This layered structure view will facilitate the derivation of gradients in Section III below.

The switching layer, i.e., Layer 1, is jointly learned with the prediction experts, i.e., Layer 3, which are typically linear models such as the most commonly employed AutoRegressive Integrated Moving Average with exogneous variables (ARIMA–X). The overall model is a mixture of experts whose weights depend on plug-loads consumption patterns. The approach is closely related to the contextual bandits in [14], which selects the best arm to pull given a context in the form of feature vectors, whereas NME is a generalization to the continuous domain by weighting the opinions of experts for prediction.

The switching layer consists of multiple templates, i.e., device consumption patterns, corresponding to building operation modes. This is a dimensional reduction technique similar to PCA and its variants. Nevertheless, there are two main differences: 1) we impose nonnegative constraints on the template entries in recognition of the fact that we only have plug loads and no power sources , 2) NME achieves dimension reduction through directly minimizing prediction performance, whereas PCA aims at improving predictions by performing minimization of matrix approximation errors. The design considerations are aligned with prediction performance, especially when the buildings can exhibit several operation modes depending on occupant activities. The generalization from the 1h ahead prediction to the 24h ahead is straightforward by incorporating features from the previous days.

B. Probability Perspectives

The NME bears many similarities with the Mixture of Experts and the Hierarchical Mixture of Experts model, and can be viewed as a mixture model with data generated from different latent processes under their individual probability distributions, $P(y_i|x_i, z_i, \theta_i)$, as illustrated in the plate notation in Figure 1 (b). The probability of y conditioned on all the information is given by:

$$P(y|X,Z) = \sum_{c_i} P_i(c_i|x_i) P(y|x_i, z_i, c_i, \theta_i) \tag{8}$$

where $P_i(c_i|x_i)$ is the gating network outputs (4), c_i is the realization of the latent variable C, an indicator variable that determines which expert to consult, and $\theta_i = \left[\mu_i, \sigma_i^2\right]$ is the expert network parameters, for instance, the Gaussian distribution with the mean determined by a regressive process, shown in (5). The posterior probabilities of the latent variable C, which incorporates the output y as well as the inputs, is given by:

$$P(c_i|y, x_i, z_i) = \frac{P_i(c_i|x_i) P(y|x_i, z_i, c_i, \theta_i)}{\sum_{c_k} P_k(c_k|x_i) P(y|x_i, z_i, c_i, \theta_i)} \tag{9}$$

A simple example would be the case where the output is a scalar, and the expert network distribution is Gaussian. Denote the gating outputs as $g = [g_1, ..., g_K]^\top$, the probability of the estimate is then given by:

$$P(y|X,Z) = \sum_{i=1}^{K} g_i \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{(y-\mu_i)^2}{2\sigma^2}\right) \tag{10}$$

This suggests the use of EM algorithm for model learning, which augments the above probability with the latent variable C and maximizes the "complete log-likelihood" over all the gating and expert network parameters [11].

C. Offline Learning: Batch Gradient Descent

The gradients of the objective function $F(A, H, R; y, X, z)$ can be derived with a back propagation method, which utilizes the chain rule of derivatives:

$$\frac{\partial E_i}{\partial O_{k,t}^l} = \sum_{j=1}^{\#(l+1)} \frac{\partial E_i}{\partial O_{j,i}^{l+1}} \frac{\partial O_{j,i}^{l+1}}{\partial O_{k,i}^l} \tag{11}$$

where $O_{k,i}^l$ denotes the output of the k-th node in the layer l for the i-th training sample, and $E_i = \sum_{j=1}^{\#(L)} (T_{j,i} - O_{j,i}^L)^2$ is the quadratic error measure, summed over all target outputs in the last layer. For instance, $\frac{\partial E_i}{\partial O_{k,t}^L} = 2(T_{k,i} - O_{m,i}^L)$ for the output layer; $\frac{\partial O_{k,i}^1}{\partial H_{\cdot,k}} = \sum_{j=1}^{m} A_{k,j}^{(i)} X_{\cdot,j}^{(i)}$ for the derivative of the first layer with respect to gating parameter $H_{\cdot,k}$, where we view the inputs layer with index 0. The partial derivatives of the cost function with respect to the parameters, A, R, H can be derived easily in this fashion.

For the batch method, the update of parameters is based on full gradient descents by considering all the training samples:

$$\begin{aligned} A_{ij} &= A_{ij} - \alpha \frac{\partial}{\partial} F(A, H, R; y, X, z) \\ R_{ij} &= R_{ij} - \alpha \frac{\partial}{\partial R_{ij}} F(A, H, R; y, X, z) \\ H_{\cdot,j} &= \mathrm{Proj}_+\left(H_{\cdot,j} - \alpha \frac{\partial}{\partial H_{\cdot,j}} F(A, H, R; y, X, z)\right) \end{aligned} \tag{12}$$

where α is the learning rate. In addition to the gradient descent, we also implement an extra non-negative projection for the parameter H to satisfy the non-negativity constraint, with $\mathrm{Proj}_+(v)$ taking the negative part of v to be zero and keep other parts unchanged.

Theoretically, one can prove that the sequence of parameters, $(A^{(k)}, H^{(k)}, R^{(k)})$ and $(A^{(k+1)}, H^{(k+1)}, R^{(k+1)})$ will converge to a stationary point $(A^{(*)}, H^{(*)}, R^{(*)})$ as the learning rate α shrinks down. The extra projection step maintains this good property because of the firm non-expansivity of projection operators [15].

D. Online Learning: Stochastic Gradient Descent

While batch gradient descent can lead to guaranteed convergence [15], it requires processing all the training samples in an epoch to update the parameters with a gradient, which can be memory and computationally demanding. So instead of using all the samples, it is necessary to make an update with fewer amounts of samples each time, which gives rise to the idea of stochastic gradient descent.

For the stochastic gradient descent, we are only considering $N^{(i)} < N$ training samples, where N is the entire set of data. Each update is then an accumulated sum of the gradients of these $N^{(i)}$ samples. Stochastic gradient descent can lead to a smaller objective function each time in expectation, and it also converges to a stationary point as the batch method. It is particularly suitable for online learning because each time we can update the gating and experts networks when new data are available, and the computation is very efficient. The learning algorithm is exactly the same as the batch method, so we leave it to the interested readers.

IV. NME FOR DEVICE POWER DATA

The NME model connects supervised learning to unsupervised learning by establishing a statistic based on prediction errors to guide the two processes. In this section, we consider the application of NME to the smart building device network. The dataset was gathered in a study of energy consumption of plug-in devices, which took place in the Building 90 (B90) of Lawrence Berkeley National Lab (LBNL) from July 2010 through February 2011. The measurement frequency is at a 15 minutes interval, for a span of 3 months. For the large-scale problem we use the measurement data of 35 devices in an office space, including computers, projectors, coffee makers, desk lamps, etc. Figure 2 shows some typical measurements in the data.

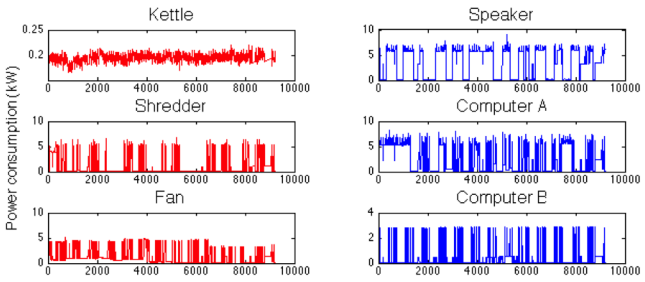

Fig. 2. Typical power measurement data in three months for devices, such as computers, speakers kettle, shredders, and fans.

We denote the device measurement data matrix as X, which is of size p by N, where p is the number of devices, and N is the number of samples. We also sum the columns of X to obtain the vector y, where each component corresponds to the total energy consumed by the device network. The knowledge discovery task is to learn about the energy use of workplace plug-in devices (also known as Miscellaneous and Electronic Loads – MELs) from the dataset. A sensible criterion on whether the discovered patterns are useful is whether we can make more accurate predictions based on these discovered patterns. NME exactly achieves these in a unifying framework of knowledge discovery and prediction.

As discussed in Section III, the training of NME is based on gradient projection method, which has been shown to converge to global optimal with a linear rate of convergence, $O(1/k)$, for convex problems. Due to the non-convexity of the objective function, we do not have the luxury of linear convergence rate and global optimality; nevertheless, with shrinking step-size and firm non-expansive projection operator, the method will eventually converge to a local optimal point following a "zigzag" path, as shown in Figure 3 (a). The convergence is compared with ANFIS, a popular neural network model with fuzzy logic rules [16]. For all the training cases where we start with random parameter values, NME converges not only faster, but also to a solution with lower training errors than ANFIS, indicating that NME can achieve a model with higher fidelity. The impressive point is NME has fewer parameters, and it is only limited to linear multiplication and scalar division, rather than Gaussian kernels as in ANFIS, which might be computationally intensive for large dataset.

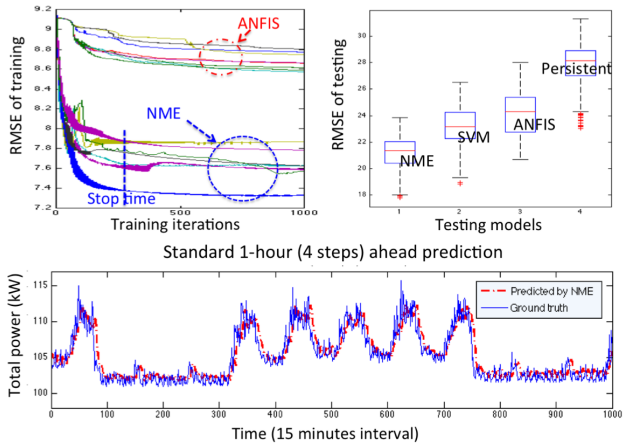

Fig. 3. a) Convergence behavior during model training for ANFIS and NME (top-left). b) The 1-hour (4 steps) ahead prediction performance of NME, SVM, ANFIS, and Persistent model (top-right). c) 4-step ahead prediction results by NME with ground truths (bottom).

The NME model can be naturally extended to predict further into the future by adding #*Ahead* (horizon) by #*Feature* number of parameters, so the model size still remains manageable. Figure 3 (b) illustrates prediction results for a 1-hour ahead prediction, compared with SVM and ANFIS, which are implemented by the standard MATLAB packages [17]. The persistent model is a naive approach of using the mean of the most immediate data to predict the future. The 1-hour ahead prediction is a standard prediction interval used by power aggregators such as the California Independent System Operator (CAISO). The testing was performed 1,000 times using different data sets for each model. The box plots illustrate the mean (red line), the 25th and 75th percentiles, the whiskers, and the outliers ('+'). As can be seen, the NME performs best on average in these testing cases, followed by

SVM and ANFIS, and the Persistent model gives the worst performance. The results are summarized in Table I.

TABLE I
SUMMARY OF PREDICTION ERRORS FOR DIFFERENT MODELS

	Persistent	**ANFIS**	**SVM**	**NME**
Root MSE (kW)	27.9171	24.0796	23.1256	**21.1799**
Std. Dev. (kW)	1.5618	1.6735	1.4437	**1.2155**

The result of the 1-hour ahead prediction is shown in Figure 3 (c). We observe that NME makes reasonably well predictions. It is worth mentioning that the model tends to make a conservative guess during the idle times, and a more aggressive guess during the working time, because of the segregation of model predictions by different experts.

In addition to the prediction power, the merit of NME compared with other prediction models is that it blends unsupervised learning into the supervised model, so that the knowledge learned from the model naturally arises from optimizing some metric, in this case, the prediction error. In the smart building context, we care about the building occupants' energy consumption behavior as much as the total power consumed. We can examine the feature vectors of the learned NME model, as shown in Figure 4 below.

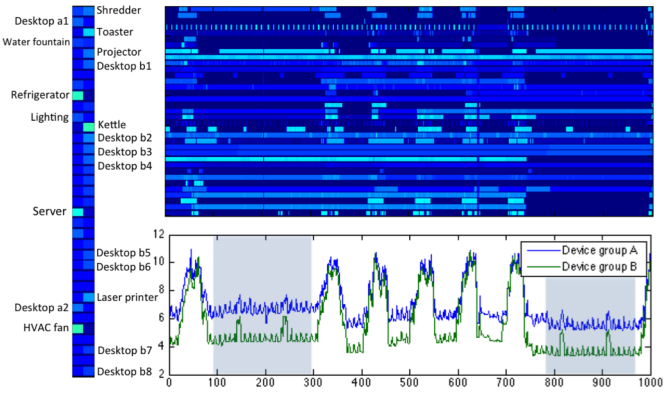

Fig. 4. Left: the learned feature vector whose active sites are labeled with devices. Top right: device measurement data matrix with lighter color indicating high power consumption. Bottom right: Corresponding feature weights throughout time.

As can be seen, using data of 10 days, we successfully identified patterns that are typical of working hours as well as idle times. In particular, the two feature vectors essentially segregated the devices into two groups. Close inspection reveals that these two groups have characteristic devices: devices that are active in group A (left feature vector in Figure 4) include refrigerators, water fountains, lightings, servers, HVACs, that generally operate with and without occupant presence; group B includes mostly desktops, as well as shredders, projectors, printers, etc., that are used frequently during working ours, but left idle after work. We also identified some active desktops in group A, which might indicate some working patterns of the owners.

The discovered group of feature vectors can assist in making power predictions as follows. With the corresponding learned expert models, if presented with a past history of data, it first decides which expert it should turn to by calculating the similarities of the past data with each of these feature vectors, and then makes a prediction based on the weighted sum of experts' opinions.

To test the statistical significance of the device patterns discovered by the NME model, we apply the classical idea of permutation test based on a statistic that measures the effect of the claimed expert model selection mechanism. The null hypothesis H_0 is that the selection of expert model based on the learned device pattern has no effect on NME's prediction error. The original time series of the device network have specific labels, such as projectors, coffeemakers, refrigerators and computers. Correspondingly, the learned feature vectors have labels attached to each of its entries as illustrated in Figure 4. If we randomly permute the labels of the original time series and use the same feature vector, under the null hypothesis, this should not make a difference in the model's prediction power. To make the added variation due to resampling as small as possible, this random permutation was conducted 5,000 times, each time with a randomly permuted time-series data matrix, where we only permuted the labels of the time series rather than the points within. Figure 5 shows the permutation distribution of the test statistic.

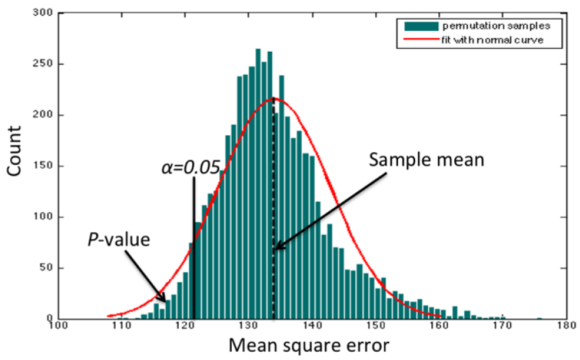

Fig. 5. The distribution of the statistic of MSE prediction error in the permutation test.

The test statistic is summarized in Table II. In Figure 5, the dashed line marks the mean of the distribution. The solid black line is the .05 significance level for one-tailed test. The sampling distribution is close to normal, as indicated by the red fitting line. The P-value of this permutation test is .0144, which is the probability that we would observe a statistic value as extreme or more extreme than the one we did observe under the null hypothesis. Obviously, we are confident to reject the null hypothesis and adopt the alternative: the feature vector learned by the NME model did make a difference in the prediction performance. This suggests that the learned features are very likely describing the underlying energy usage patterns.

TABLE II

SUMMARY OF STATISTICS FOR PERMUTATION TEST

Observed	Mean	SE	Permutation test P-value
118.4331	134.0764	8.7313	0.0144*

V. CONCLUSION

Reliable and decentralized electric power forecast based on local energy consumption behavior is crucial to the implementation of demand response and dynamic pricing. Modeling consumer behaviors in commercial buildings is of particular interests because of the potential to significantly contribute to the accurate estimation of demand side in the power system [18].

NME explores a connection of supervised and unsupervised learning by establishing a statistic based on prediction errors to guide the two learning processes. It has particular strengths of making accurate prediction of total energy consumption of a building space, and also recognizing statistically significant energy consumption behaviors that reveal the underlying social structures and working patterns of occupants for large-scale problems. The knowledge learned from the model is valuable input to device-level clustering, building state classification and local management decisions.

NME can be implemented at the building embedded system level by adding an extra layer to the existing prediction methods for switching, assuming that the prediction method is a linear model and can be adapted to the mixture of experts framework. The computation complexity is not limiting the performance. For the building level, the building manager can take plug-loads consumption of each floor or centers for building modes identification, and the prediction of building energy consumption can be used to facilitate demand response scheduling. At the utility, each building can be associated with a collection of features; nevertheless, for the concern of privacy, only the consumption of few devices and public information such as weather forecasts can be accessible.

One future research direction is to extend the capability of the model to identify key characteristic devices that are critical to the local decision of energy consumption, as well as to detect possible inefficient energy consumption behaviors. It is also interesting to apply the model in the optimal control task by designing a decision metric that maximizes the expected utility based on the learned feature patterns.

ACKNOWLEDGMENTS

This research is funded by the Republic of Singapore's National Research Foundation through a grant to the Berkeley Education Alliance for Research in Singapore (BEARS) for the Singapore-Berkeley Building Efficiency and Sustainability in the Tropics (SinBerBEST) Program. BEARS has been established by the University of California, Berkeley as a center for intellectual excellence in research and education in Singapore.

REFERENCES

[1] L. Pérez-Lombard, J. Ortiz, and C. Pout, "A review on buildings energy consumption information," *Energy and buildings*, vol. 40, no. 3, pp. 394–398, 2008.

[2] J. Granderson, M. A. Piette, G. Ghatikar, and P. Price, "Building energy information systems: State of the technology and user case studies," *Handbook of web based energy information and control systems*, 2009.

[3] J. L. Mathieu, D. S. Callaway, and S. Kiliccote, "Examining uncertainty in demand response baseline models and variability in automated responses to dynamic pricing," in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*. IEEE, 2011, pp. 4332–4339.

[4] M. Amin-Naseri and A. Soroush, "Combined use of unsupervised and supervised learning for daily peak load forecasting," *Energy Conversion and Management*, vol. 49, no. 6, pp. 1302–1308, 2008.

[5] A. H. Neto and F. A. S. Fiorelli, "Comparison between detailed model simulation and artificial neural network for forecasting building energy consumption," *Energy and Buildings*, vol. 40, no. 12, pp. 2169–2176, 2008.

[6] B. Dong, C. Cao, and S. E. Lee, "Applying support vector machines to predict building energy consumption in tropical region," *Energy and Buildings*, vol. 37, no. 5, pp. 545–553, 2005.

[7] S. A. Kalogirou and M. Bojic, "Artificial neural networks for the prediction of the energy consumption of a passive solar building," *Energy*, vol. 25, no. 5, pp. 479–491, 2000.

[8] D. Kahneman, *Thinking, fast and slow*. Macmillan, 2011.

[9] M. Jin, R. Jia, Z. Kang, I. C. Konstantakopoulos, and C. Spanos, "Presencesense: Zero-training algorithm for individual presence detection based on power monitoring," in *BuildSys14, November 5–6, 2014, Memphis, TN, USA*, 2014, pp. 1–10.

[10] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, "Adaptive mixtures of local experts," *Neural computation*, vol. 3, no. 1, pp. 79–87, 1991.

[11] M. I. Jordan and R. A. Jacobs, "Hierarchical mixtures of experts and the em algorithm," *Neural computation*, vol. 6, no. 2, pp. 181–214, 1994.

[12] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Advances in neural information processing systems*, 2001, pp. 556–562.

[13] M. W. Berry, M. Browne, A. N. Langville, V. P. Pauca, and R. J. Plemmons, "Algorithms and applications for approximate nonnegative matrix factorization," *Computational statistics & data analysis*, vol. 52, no. 1, pp. 155–173, 2007.

[14] N. Cesa-Bianchi and G. Lugosi, *Prediction, learning, and games*. Cambridge University Press, 2006.

[15] D. G. Luenberger and Y. Ye, *Linear and nonlinear programming*. Springer Science & Business Media, 2008, vol. 116.

[16] J.-S. Jang, "Anfis: adaptive-network-based fuzzy inference system," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 23, no. 3, pp. 665–685, 1993.

[17] C.-C. Chang and C.-J. Lin, "Libsvm: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, p. 27, 2011.

[18] Z. Kang, M. Jin, and C. J. Spanos, "Modeling of end-use energy profile: An appliance-data-driven stochastic approach," in *The 40th Annual Conference of the IEEE Industrial Electronics Society, Dallas, TX, USA*, 2014, pp. 5382 – 5388.