Watts App:
An Energy Analytics and Demand-Response Advisor Tool

Santiago Gonzalez, Case Western Reserve University, Electrical Engineering, *SUNFEST Fellow*

Dr. Rahul Mangharam, Electrical and Systems Engineering

Abstract— **Real-time electricity pricing and demand response has become a clean, reliable and cost-effective way of reducing peak demand on the electricity grid. Annual revenues to end-users from demand response markets are more than $700 million, making demand response the largest virtual generator in use [6]. DR-Advisor, an open source software tool created at the University of Pennsylvania, acts as a recommender system for building's facilities manager. Using historical data from a building, DR-Advisor uses data-driven models to suggest suitable control actions to meet the desired load curtailment during demand response events. Using data sets from several buildings on the University of Pennsylvania's campus, we enhance the capability of DR-Advisor by adding plug-ins for data-preprocessing and energy analytics.**

I. INTRODUCTION

Wholesale electricity markets in the United States all use some form of real-time locational marginal pricing, where prices are calculated based on the operating conditions of the electricity grid. During intervals of high electricity consumption or peak demand, electricity prices increase substantially, making power consumption both inefficient and extremely cost intensive for end-use customers. Figure-1 shows an example of the volatility in real-time pricing from the New England independent system operator. The nominal price of electricity starts out at $25/MWh but increases to $800/MWh on July 20th, 2015 [5]. In an effort to reduce peak power consumption and decrease electricity costs, customers have begun to depend on demand response (DR). DR programs involve a voluntary response of a building to real-time price signal. In such programs, end-users receive a notification from the utility requesting a reduction in their electricity load during periods of peak demand. Customers curtail power consumption during a predetermined amount of time and as a result receive a financial reward [2]. To be able to take advantage of real-time pricing and DR programs, the consumers must monitor electricity prices and be flexible in the ways they choose to use electricity. The challenge for large buildings lies in evaluating and taking control decision at fast time scales. Buildings are complex systems with many interconnected subsystems operating independently of each other. HVAC systems, chillers systems and lighting systems all operate independently of each other, making it difficult to analyze and synthesize to effect of any control action on system behavior.

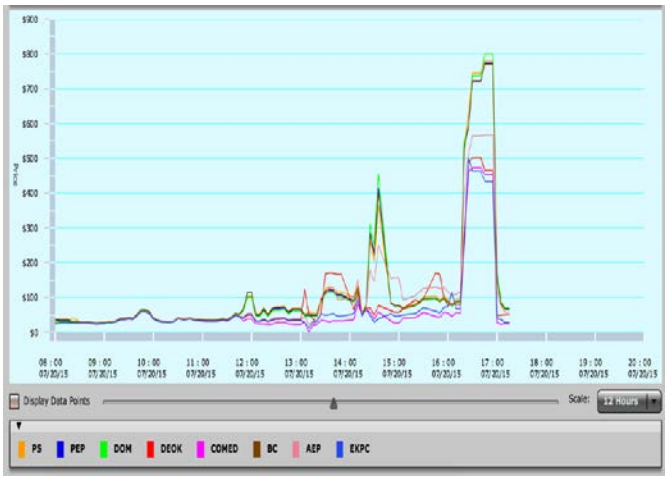

Figure 1- Real time electricity prices from New England ISO during 07/20/2015

DR-advisor uses regression tree-based algorithms to predict power consumption of large-scale commercial buildings in real time. These models are then used to create suitable control and scheduling strategies to meet the desired curtailment during a DR event. The problem is that data-driven model predictive accuracy depends of the quality of the data used for training the model. Building management systems consist of thousands of sensors embedded in the systems that control the internal environment, which often break or go offline causing noisy data. My work this summer consisted of creating a framework for data preprocessing which takes historical data files from any Penn building processes them to create suitable structure for training of data-

driven model. Processing includes outlier removal and interpolation. I also developed the capability to perform energy analytics on regression trees. This was the first step in designing a query system for the facilities managers. I then evaluated my contributions with data sets containing power consumption data for buildings on the University of Pennsylvania campus.

II. Background

Electricity generators and utility companies use real-time locational marginal pricing, making electricity costs exceptionally sensitive to human behavior and extreme weather conditions. This translates into electricity prices dozens of times more expensive for end-users. For instance, at the University of Pennsylvania, the nominal rate of $30/MWh increased to $817/MWh during a hot summer day in 2011, a 27-fold increase. The five most expensive days cost $1.47 million, accounting for 5.1% of the total bill [1].

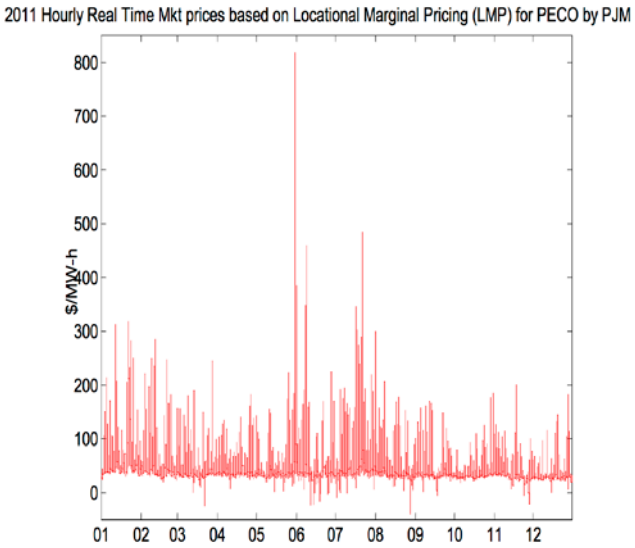

Figure 2: 2011 Hourly Real Time Market prices based on Location Marginal Pricing for PECO by PJM

In order to make effective use of DR programs, end users must be able to both predict power consumption and take appropriate actions in real time. Currently, rule-based and model-based strategies are the two most common approaches to responding to DR events. In rule-based strategies, curtailment is met through the implementation a pre-programmed plan. Although simple, the rule-based approach does not account for historical building or weather data. Rule-based strategies also lack any predictability capabilities. Model-based strategies rely on mathematically modeling a building and its equipment. The models are used to predict power consumption and to synthesize control strategies [3]. Buildings are complex systems with a large number of individual components that interact in a convoluted manner, making the creation of high fidelity models both time and cost intensive. In addition, complex models involve many factors that hinder the facility manager's ability to interpret and synthesize data in a meaningful manner.

DR-advisor uses a regression tree-based algorithm, whose innate characteristics make it a suitable strategy to meet the challenges that DR events pose. Below, I outline some of the unique advantages which make regression trees suitable for solving the challenges of demand response [4].

- Fast computation times
- Handle a lot of variables
- Robust to missing data and outliers
- Very easily interpretable

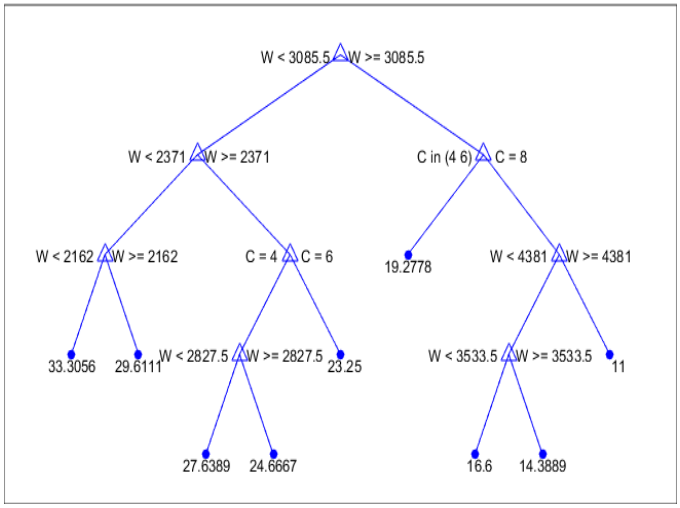

Figure 3: Regression Tree created by fitrtree method in MATLAB

A. *Data Description:*

Each regression tree needs to be trained on time-stamped historical data. I worked with data for nine buildings on the University of Pennsylvania Campus. The buildings that were included were Annenberg Center, Annenberg School, Clinical Research Building, College Hall, David Rittenhouse Laboratory, Goddard Labs, Huntsman Hall, and Vance Hall. Each building had comma separated value (CSV) files with weather, schedule and building data. The data included information for approximately 18 months at a resolution

of 1 hour time-steps. The CSV files consisted of 17 columns with proxy variables that included year, month, day of month, hour of day, building area and occupancy, weather variables including outside air temperature, dew point, relative humidity, incident solar radiation, wind speed, gusts speed, wind direction, heating and cooling degree days, and the power consumption in kW.

For some of the buildings, up to 25% of the data was either missing or considered an outlier. Instead of discarding valuable data, the goal was to create a plug-in for DR-Advisor that would remove outliers, and interpolate over missing data to improve model accuracy.

III. Case Study

I will use College Hall to present a comprehensive case study. College Hall was the first building on the West Philadelphia campus and currently home of the President, Provost, School of Arts and Sciences, the Department of History and the Undergraduate Admissions Office. College Hall has 6 floors with a total gross area of 110,266 square feet. The CSV file for College Hall had historical data starting on the July 18th, 2013 and ending on February 4th, 2015.

Figure 4: College Hall

A. Pre-Processing

Using the MATLAB statistics and machine learning toolbox, I created scripts for each of the buildings that contained functions for the importation, outlier removal, and interpolation of data. The script imported the data by parsing the CSV file and assigning each of the columns to a variable. Predictor features were assigned to X and the power consumption values assigned to Y. I then created a function that calculated the mean ($\bar{x}$) and the standard deviation (σ) of Y.

$$\sigma = \sqrt{\frac{\sum(x-\bar{x})^2}{n}} \qquad \bar{x} = \frac{\sum x_i}{n}$$

Data samples that were two standard deviations away from the mean were deemed outliers and removed from.

B. Interpolation

For the interpolation of data, we had to make several assumptions in attempt to characterize missing data. For non-proxy variables and power consumption values, we took values of zero to be missing data. For power consumption values, the assumption is justified by the observation that under normal operating conditions, any occupied building will always be consuming power. For non-proxy variables, we found that zeros could indicate both missing and actual values. In the case of missing value, the interpolation would act as intended. In the case that we interpolated over actual zero values, it was expected that neighboring values would be close to zero, since weather data exhibits linear behavior at high resolutions. Therefore, the interpolated value tended to be close to zero. Overall, we found that interpolation of non-proxy variables led to higher model accuracy. Interpolation was not applied to proxy variables, because it was not possible to distinguish in between missing data and samples that had values of zero.

After testing several interpolation methods, we found linear interpolation to be the most effective in handling long strings of missing values. I used the interp1 method from MATLAB, in which the interpolated value at a query point is based on linear interpolation of the values at neighboring grid points in each respective dimension. The equation is outlined below, where (x, y) is the query point to be interpolated and (x_0, y_0) (x_1, y_1) are the neighboring data samples.

$$y = y_0 + (y_1 - y_0)\frac{x - x_0}{x_1 - x_0}$$

Power consumption training data for College Hall before and after outlier removal and interpolation is shown in Figure 4.

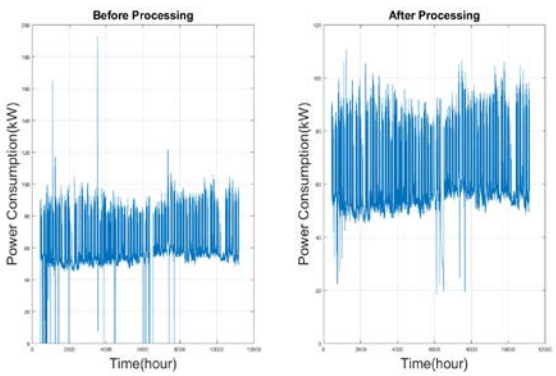

Figure 5: Left: College Hall Power Consumption before Processing. Right: College Hall Power Consumption after Processing

IV. Results

The metric for prediction accuracy was the normalized root mean square error (NMRSE). NRMSE is the RMSE divided by the mean of the data. The RMSE represents the sample standard deviation for the difference between predicted values and observed values. To test the efficacy of outlier removal and interpolation, I trained models for all of the buildings with both raw data and pre-processed data. I then calculated and compared NMRSE values for models trained with both the raw and processed data. The NMRSE values are shown in Table-1.

Regressions trees tend to have high variance and may sometimes over fit the data. It is a tradeoff to be paid for estimating a simple model. In order to grow more stable trees, DR-Advisor uses several ensemble methods. The effects of pre-processing were evaluated on the following algorithms: single regression tree, k-fold cross-validated trees, and random forests.

Building	Method	Before Processing NRMSE %	After Processing NRMSE %
College Hall	Single Tree	21.84	14.05
	Cross-Validated Tree	17.39	11.50
	Random Forest	12.27	11.23
Vance Hall	Single Tree	17.49	14.04
	Cross-Validated Tree	14.24	11.01
	Random Forest	10.39	9.26
DRL	Single Tree	13.12	11.80
	Cross-Validated Tree	9.93	10.57
	Random Forest	8.60	8.99
Goddard Labs	Single Tree	25.99	16.41
	Cross-Validated Tree	25.49	16.14
	Random Forest	17.73	15.55
Vagelos Labs	Single Tree	68.87	44.28
	Cross-Validated Tree	68.15	43.98
	Random Forest	68.38	43.18
Annenberg School	Single Tree	23.76	22.84
	Cross-Validated Tree	21.94	21.84
	Random Forest	19.91	19.88
Fisher and Duhring Wings	Single Tree	43.58	31.40
	Cross-Validated Tree	39.09	26.97
	Random Forest	34.77	23.61
CRB	Single Tree	19.36	5.20
	Cross-Validated Tree	10.61	4.54
	Random Forest	8.04	3.34
Annenberg Center	Single Tree	29.31	29.46
	Cross-Validated Tree	28.49	27.81
	Random Forest	27.88	28.05

Table 1: NMRSE values for buildings on the Penn Campus

Outlier removal and interpolation made significant improvements to the prediction accuracy of the models. College Hall saw an improvement of 35.7% for the single tree, 33.9% for the Cross Validated Tree, and 8.5% for the random forest. Figure-4 shows prediction for each algorithm compared to the ground truth both raw and processed data.

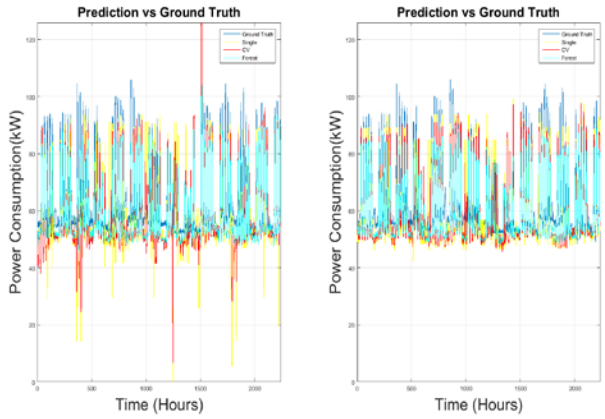

Figure 6: Left: College Hall Power Consumption Prediction with Raw Data. Right: College Hall Power Consumption Data with Processed Data.

It is important to note that proxy variables are important predictor of building power consumption. This is because they capture repeated patterns of occupancy and building operation. Figure-5 shows the importance of each of the predictor variables. Since neither outlier removal nor interpolation were performed on proxy variables, improvements were not quite as effective as they might have been had proxy variables been able to be processed.

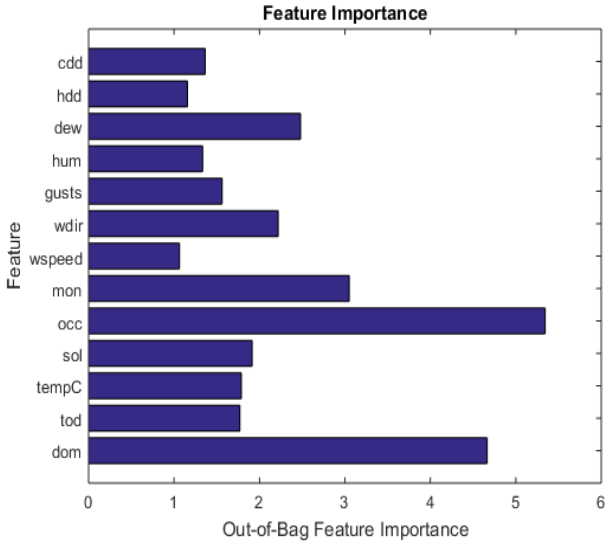

Figure 7: Feature Importance for training College Hall model

A. Filtering of Model Predictions:

Demand Response advisor uses the fitrtree MATLAB method to grow the regression trees. The fitrtree method works by recursively partitioning the feature space into a set of rectangles and then fitting a simple model in each one. The fitrtree method starts by considering input data and all possible binary splits on every predictor, and then selecting a split based on the best optimization criterion. It then repeats recursively until it meets a stopping criterion. Stopping criteria is met under two circumstances. The first condition is when the mean squared error (MSE) for the observed response in the node drops below some predetermined threshold. The second circumstance is when there are fewer than the minimum amount of observations in the node. The minimum amount of observations is predetermined by the user. A node that fits the stopping criteria is called a leaf node. Although a 1 data point-per-leaf minimum requirement can be assigned as the stopping criterion, a very large tree might over fit the data. Therefore, leaf nodes tend to have a set of data points within the partitioned space. By querying the data samples within each partitioned space, we can get insights into building behavior at selected levels of power consumption.

B. Procedure:

We find all the leaf nodes that lie within a user-specified power consumption range. We then find the data points that lie within each of the leaves. The value of each feature is extracted from each data sample and added to a data structure that groups together values for equivalent features for all of the data points within the specified range.

For the College Hall data, we split power consumption data into 10 different bins of equal width. We then grouped all of the data samples that lied within each bin. Figure-6 shows the average prediction of each leaf.

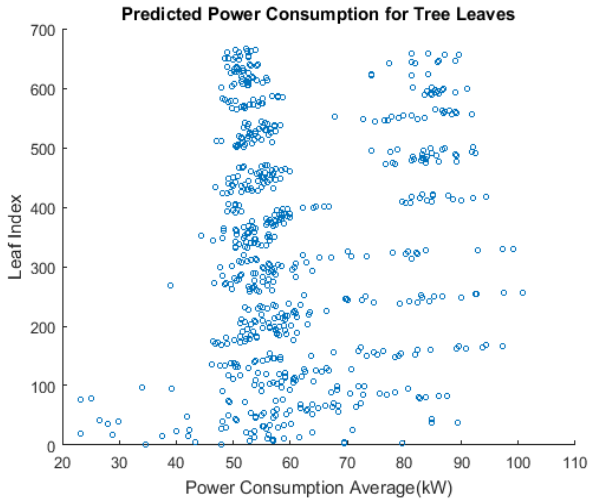

Figure 8: Plot of Power Consumption Averages for Leaf Nodes in College Hall

For each of the bins, we calculated the confidence interval and support. Confidence interval and support give insight into the frequency and certainty in which the building will consume a specific amount of power. We also use it to find any rare events, occasions in which an event has less support but higher confidence.

Figures with boxplots for each of the non-proxy predictor features were displayed. Each figure contained a boxplot for each of the bins. The boxplots provide the ability for users to see the distribution for each feature at a given power consumption. For example, the user could find under which temperature conditions the building would consume 90-100 kW of power. Figure-7 shows temperature feature distribution at each bin. For proxy variables, the three most frequently occurring values were calculated. The user can calculate, for example, what

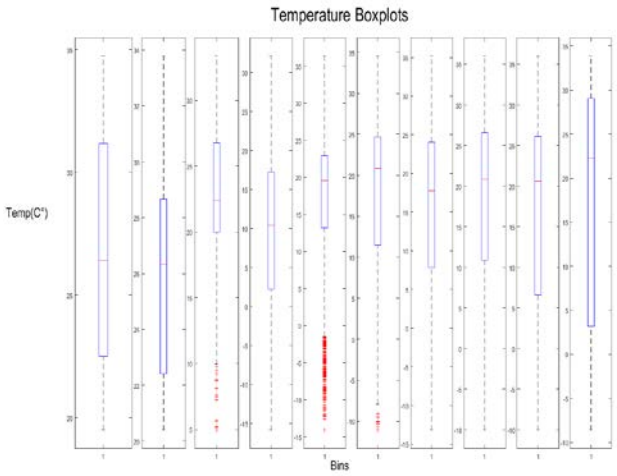

Figure 9: Boxplots of Temperature for each of the Bins in the College Hall Data.

V. Conclusion

Electricity costs are the single largest component of a large commercial and industrial building's operating budget. For such consumers, buying and reacting to real-time electricity prices is not as simple as paying a flat-rate monthly bill. Their power consumption demands are sensitive to both human behavior and weather conditions. DR- Advisor, a software tool that acts as a recommender system for the building's facilities manager, provides suitable control actions to meet the desired load curtailment while maintaining operations and maximizing the economic reward. We show that by preprocessing the incoming data, we dramatically improve the performance and accuracy of the models used by DR-Advisor. We also show that by querying the regression trees we make regression trees more interpretable by getting insight into building behavior not attainable otherwise. The developed plug-ins will be added to the DR-Advisor toolbox.

VI. Acknowledgment

I would like to thank Madhur Behl, graduate student at the University of Pennsylvania, with whom I collaborated extensively with on this project. I would also like to thank Dr.Rahul Mangharam for allowing me to work as a part of his research team and for his help and guidance on this project throughout the summer.

References

[1] Behl, Madhur. "Sometimes, Money Does Grow On Trees: Data-Driven Demand Response with DR-Advisor"
[2] C. Goldman. Coordination of energy efficiency and demand response. *Lawrence Berkeley National Laboratory, 2010.*
[3] D. B. Crawley, L. K. Lawrie, *et al.*, "Energyplus: creating a new generation building energy simulation program," *Energy and Buildings*, vol. 33, no. 4, pp. 319 – 331, 2001.
[4] L. Breiman, "Random forests," Machine learning, vol. 45, no. 1, pp. 5–32, 2001.
[5] New-England ISO. (2013) Real-time maps and charts, archives.
[6] P. Interconnection, "2014 demand response operations markets activity report," 2014.