FORECASTING CUSTOMER'S ENERGY DEMAND USING MACHINE LEARNING

SAIFUL ABU

Department of Computer Science

APPROVED:

Christopher Kiekintveld, Chair, Ph.D.

Luc Longpré, Ph.D.

Mohamed Amine Khamsi, Ph.D.

Pablo Arenaz, Ph.D.
Dean of the Graduate School

to my

MOTHER and FATHER

with love

Acknowledgements

I would like to express my deep-felt gratitude to my advisor, Dr. Vladik Kreinovich of the Computer Science Department at The University of Texas at El Paso, for his advice, encouragement, enduring patience and constant support. He was never ceasing in his belief in me (though I was often doubting in my own abilities), always providing clear explanations when I was (hopelessly) lost, constantly driving me with energy (*Where does he get it?!*) when I was tired, and always, *always* giving me his time, in spite of anything else that was going on. His response to my verbal thanks one day was a very modest, “It’s my job.” I wish all students the honor and opportunity to experience his ability to perform at that job.

I also wish to thank the other members of my committee, Dr. Luc Longpré of the Computer Science Department and Dr. Mohamed Amine Khamsi of the Mathematics Department, both at The University of Texas at El Paso. Their suggestions, comments and additional guidance were invaluable to the completion of this work. As a special note, Dr. Longpré graciously volunteered to act as my advisor while Dr. Kreinovich was working abroad in Europe. He was extremely helpful in providing the additional guidance and expertise I needed in order to complete this work, especially with regard to the chapter on NP-hard problems and the theory of NP-completeness.

Additionally, I want to thank The University of Texas at El Paso Computer Science Department professors and staff for all their hard work and dedication, providing me the means to complete my degree and prepare for a career as a computer scientist. This includes (but certainly is not limited to) the following individuals:

Dr. Andrew Bernat

> He made it possible for me to have many wonderful experiences I enjoyed while a student, including the opportunity to teach beginning computer science students the basics of UNIX and OpenWindows (something I wish I had been taught when I first started), and the ability to present some of my work at the University of Puerto Rico, Mayagüez Campus.

Dr. Michael Gelfond

> His influence, though unbeknownst to him, was one of the main reasons for my return to UTEP and computer science after my extended leave from school while island hopping in the navy. He taught me many things about computer science—and life. Among the many things he showed me was that there really is science in computer science.

And finally, I must thank my dear wife for putting up with me during the development of this work with continuing, loving support and no complaint. I do not have the words to express all my feelings here, only that I love you, Yulia!

NOTE: This thesis was submitted to my Supervising Committee on the May 31, 1996.

Abstract

Solving systems of linear equations is a common computational problem well known to mathematicians, scientists and engineers. Several algorithms exist for solving this problem. However, when the equations contain *interval coefficients* (i.e., intervals in which the desired coefficient values are known to lie), the problem may not be solvable in any reasonable sense. In fact, it has been shown that the general problem of solving systems of linear equations with interval coefficients is NP-*hard*, i.e., extremely difficult and (it is believed) unsolvable; thus, no feasible algorithm can ever be developed that will solve all particular cases of this problem.

It turns out, though, that the widths of the interval coefficients are quite small in a large number of the linear systems having interval coefficients. This becomes readily apparent when we learn that the intervals typically come from measurements.

Any measurement of a physical quantity is limited by the precision and accuracy of the measuring device. To be of practical use, the measuring devices used in science and industry must be reasonably accurate. This implies that, for the most part, the actual values associated with measurements lie within relatively narrow intervals. Indeed, manufacturers often guarantee the error of their instruments to be very small.

Thus, we desire to look only at *narrow-interval* coefficients when considering the development of an algorithm for solving linear systems with interval coefficients. As there already exists an algorithm that solves most such systems, developing such an algorithm seems indeed promising. Therefore, the goal of this thesis is to answer the following question:

> *Can a feasible algorithm be developed for the general problem of solving systems of linear equations with narrow-interval coefficients?*

We show here that this problem, that of solving systems of linear equations with narrow-interval coefficients, is NP-hard; thus, we do not consider it possible to develop a feasible algorithm that will solve all particular cases of this problem.

Table of Contents

Page

Chapter

Appendix

List of Figures

Chapter 1

Smart Grid and PowerTAC Competition

In this chapter, I will describe Smart Grid and PowerTAC competition.

1.1 Traditional Energy Distribution and Consumption System

In traditional electricity generation system there are three subsystems [4]. In electricity generation subsystem, the generator rotates a turbine in magnetic field which generates electricity. The turbine rotates through the power of kinetic energy of water falling from a water fall or a river with strong current, or from the energy of nuclear powerplan or energy received from burning coal or oil. Traditional energy generation system then transmits the electricity through transmission grid and electricity gets distributed in the distribution grid. This generation system is one way meaning a single power generation source serves several consumption source.

1.2 Smart Grid

In contrast to the traditional electricity generation system, Smart Grid (SG) are two way [4]. So, any node in the distribution grid can produce electricity and push it to the distribuiton grid if necessary. The NIST report [4] states that the SG would make the electricity generation and supply robust against generator or distribution node failure, use renewable energy widely and efficiently, reduce green house gas emission, reduce oil consumption by encouraging usage of electric vehicles, it will give customers more freedom to choose among energy sources. Smart grids will encourage usage of electric vehicle as these vehicles have the ability to store power in a battery and transmit the power to the distribution grid if there is a necessity. The major challenge with the usage of renewable energy is it is uncertain. This uncertainity causes the ability to predict how much energy the SG can produce in a future time slot hard. Success of SG will need efficient methods to predict energy production [11].

1.3 Smart Grid and Renewable Energy

One of the major focus of Smart Grid(SG) will be using renewable energy. There are challenges involved with using this abundant source of energy [13]. People are already showing strong motivation to use renewable energy as indicated by the statistics that 20% of total energy is from the renewable sources which is second after coal 24%. People are using renewable energy due to economic reward and environmental concern. Major challenge with Renewable energy is amount of the energy produced is greatly varying. Since the energy produced is volatile there must be a storage mechanism that balances out the surplus energy. The usage of rechargeable electric vehicles might serve the purpose of storage. Accurate prediction of the renewable energy might enable the electric car users to absorb surplus energy and push it back to the grid in peak hours if necessary.

1.4 Importance of accurate load forecasting

. Accurate load forecasting is important to ensure efficient fuel usage, reduce wastage of energy and planning proper operation of power generators [8].

1.5 PowerTAC System

PowerTAC competition which is the abbreviation of Power Trading Agent Competition, is a low risk system that simulates a smart grid based energy system. The powerTAC simulation has several compoents such as wholse sale market, broker, customers and weather service. The system is trained on customers behaviour of several past years and uses real weather data from the past. The following sections give brief explanation of each subsystem. Predicting customer's energy demand is important becasue failure to predict the demand accurately can cause monetary and environmental loss. Acting directly on the real environment can be risky. The powerTAC simulation system gives a low risk platform where the researcher's can build and test their works before deploying to real world.

1.5.1 Broker

Brokers represent the entities that buys energy from the wholesale market and sells to the customers. Contestants implement their own brokers. Each broker's objective is to maximize its profit. A successful broker has to buy and sell energy in a profitable way. Presence of several brokers in the system makes the environment competitive and every broker has to come up with a way to attract the customers.

1.5.2 Wholesale Market

Wholesale market is the bidding place for buying energy. Brokers submit their bids for a future timeslot in the wholesale market. If the bid was successful, the broker receives its desired amount by paying certain amount of money.

1.5.3 Customers

A customer represents an entity that buys energy from the brokers. Customers subscribe to the tariffs that the brokers publish. The customers chooses the most suited and affordable tariff for them by evaluating the existing tariffs in the market. They have to pay certain amount of money to the brokers based on their tariff plans and energy usage.

1.5.4 Balancing Market

Balancing market represents the market from where the broker can buy energy in case of emergency. For example, if a broker has bought less amount of energy for a given timeslot and it finds it needs more energy then it can buy the necessary amount of energy from the balancing market. Usually, the balancing market transactions are costly for brokers than the wholesale market.

1.5.5 Weather Service

The weather service broadcasts weather forecast to the brokers. Many customer's energy usage varies based on the weather. The PowerTAC system uses the real weather data from the past.

Figure 1.1 shows a block diagram of the components of the powerTAC simulation environment.

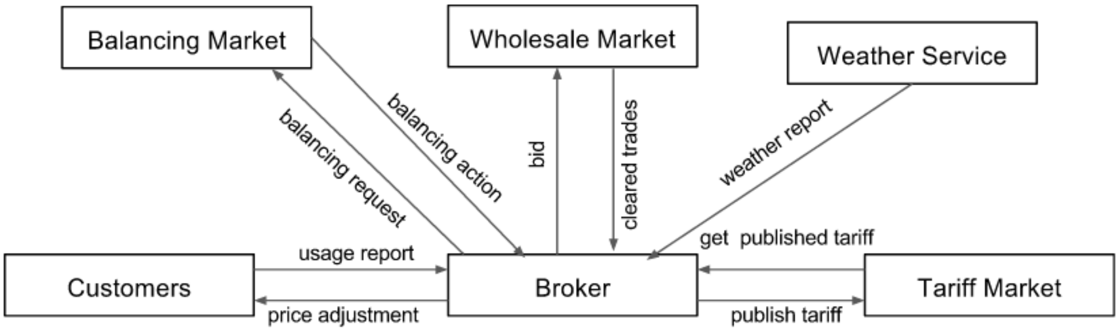


Figure 1.1: PowerTAC simulation Environment.

Chapter 2

Related Works

In this chapter I havel described different methods of energy load forecasting in the literature.

2.1 types of load forecasting

There are mainly two types of load forecasting namely short term load forecasting and long term load forecasting. Short term load forecasting deals with forecasting upto couple of weeks. Long term load forecasting may forecast customer's demand over month or year [3].

2.2 variables in energy demand

Customer's energy demand is correlated with the weather variables such as temperature and the day of the week. Researchers have found that weather effect relies on the time duration the training data has. [2] trained a SVM energy demand predictor that would predict energy demand of customers for the month January. The training data consited of every half hour's electricity demand from 1997 to 1998, average temperature from 1995 to 1998. They trained the predictor with only the portion of data that are related to the month January. They have found that within the month of January the temperature does not vary much and excluding the temperature from the feature set actually gives better prediction. Again, if energy demand is long term which means the window of prediciton is about a year, the temperature seems to have effect on the energy demand of the customers. [6] collected data of 18 months from households of a region of Australia. They collected the weather data from weather office and from self transplanted devices. They observed how the household customers use appliances based on the temperature. They came into conclusion that for that region, equilibrium point for energy usage is at temperature 0.25 degree celcius. If the temperature increases or decreases from this temperature, the electricity usage increases. They explained the behavior by stating as the temperature decreases, houses customers tend to use heaters and if the temperature rises they tend to use coolers more. [3] proposed a model that used a transfer function that relates the daily temperature with energy usage along with the ARIMA model. This scheme resulted better than the univariate ARIMA model. In the survey article [5], the authors reported that the day of week and the month of year is highly correlated with customer's energy demand. They have found that based on the hour of a given the load demand can be higher or lower. They have also found that the weekends usually have different load demand than the usual days. Finally, they found that customers load demand changes based on the season of the year. They have concluded that, weather related variables, seasonal variables should be included in the long term prediction models .

2.3 load prediction using statistical method

To make load forecast researchers have used statistical methods such as statistical average and Auto Regressive Integrated Moving Average (ARIMA). Agent TACTEX'13, the winner of the PowerTAC competition in 2013 used statistical average to make prediction for an hour of a day of a week. In a week a customer have 24 * 7 = 168 hours or slots. TACTEX'13 kept track of average usage of 168 weekly slot for each customer. To predict a future time slot, their agent would look at at which weekly slot the future time slot would fall in. Then the agent used that weekly slot's average usage as the prediciton of the future slot. [3] have used ARIMA model for load forecasting. The ARIMA model uses both moving average and auto regression to forecast the demand. To make a forecast about a future time slot, the auto regression model uses some previously observed time slots values based on its degree. Moving average scheme would use the average of all the known time series data points to make a prediciton about a future time slot .Problem with univariate ARIMA model is that they don't take into account other variables that my affect the demand such as temperature.

2.4 load prediction using machine learning

[10] the authors used varios machine learning techniques to make 24 hour ahead load forecast. They found that hour of week, weather related features such as temperature cloud cover were influential to the electricity load. They created one machine learnign forecasting module for each customers by extracting relevant features of the customers. The forecasting modules performed well for the customers that shows regularity in their energy consumption behavior. For the customers with load shifting capabilities to their favored hour, the scheme did no perform well.

2.5 load prediction for specific region

Regional load forecasting will enable us to know which regions need more energy. If we know which regions need more energy, we will know most suitable places to place electricity generator plants. [7] worked on load forecasting based on region. They diivided electricity usage of Taiwan in 4 areas. For each region, they collected GDP, population, highest temperature and aggregated load. After that, they trained Artificial Neural Network model for each region. For baseline, they trained linear regression model for each region. The result showed that, the ANN based load forecasting methods performed better than the linear regression methods.

2.6 load prediction using clustering

[9] have used clustering method to forecast customer's future electricity demand. They collected data from more than 4000 household customers in Ireland for about 6 months. Collected data included electrecticity usage at 30 minutes interval, appliances used in the home and different socio-economic information about the people living in a particular house. They clustered each days usage which they call load profiles. A customer's daily usage then can be assigned to one of those load profiles. The customer is then charactersized by the the mostly used load profile. The authors then trained a linear regression classifier that was built upon the socio-economic information of a the customers, types of appliances used in the house and the description of the house to figure out the common load profile of the given household. The predicted load profile of the customer received from the linear regression model will be used to predict the demand of the customer for a given day.[3] noticed difference of behavior among customers. They manually clustered the population in four categories namely commercial, office, residendial and industrial customers. In their paper [14], the authors proposed a novel demand prediction mechanism. In powertac competition, every broker is provided with past two weeks usage of all the customers or bootstrap usage. Their proposed broker clustered the customers based on the bootstrap data. For each cluster, the broker would make a linear regression model. The input variables included past average usage and weather related information. This approach of prediction clusters is based on the usage pattern of the customers. So this method may not be suitable for customers with irregular usage pattern such as customers with load shifting capabilities and electric vehicle customers.

2.7 engineerign methods to lead forecasting

The authors [1] have used Kalman Filter to forecast short term load demand. Kalman Filters are used widely to approximate current state of a dynamic system. To do this, it computes the next state of the system using the provided algorithm. Also, it observes what the measurements say about the current state of the system. Both of the prediction mechanisms of the current state has high uncertainity. When they are combined toghther, the uncertainity gets reduced.

2.8 expert system based load forecasting

The authors in ref [12] have proposed an expert system based load forecasting method for the region Virgina. The expert system would forecast load of upcoming 24 hours. They observed the variables that are likely to affect the load. They came up with variables such as temperature, load of previous hour, season and day of week have strong correlation with the observed load. They implemented a computer program that mimicked how a human operator makes load forecast based on the independent variables. For a specific region's weather condition, their method worked well and required limited amount of historical data.

From the review of the literature, the importance of weather related variables such as temperature, cloud cover and windspeed is evident. Also, hour of the day and day of week are highly correlated with the load demand. Combination of machine learning classifiers and clustering algorithms appears to be a better idea. For the methodlogy of [10] it will take a large number of predictors for the simulation system. Also, those predictors will not work if the name of the customer is changed or a new customer is introduced as each predictor is hardcoded with a specific customer. It sounds reasonable to cluster the data first and then train machine learning classifier for each cluster. This apporach will hold generality. Instead of training only on bootstrap data as the [14] have done, wealth of data generated from the simulations can be used to train the cluster. Since the clustering is done offline, this apporach wll not suffer from the problem of having a time limit that the broker has to face if the cluster is trained during the competition. After the clustering is done, for each cluster, different machine learning classifiers can trained to figure out which one performs the best. So, the broker will no longer sticked to linear regression. This way, the training module will be able to deal with new customers.

Chapter 3

Customer Description

In this chapter I will describe the customers present in the PowerTAC simulation system, some statistics about them and their attributes.

3.1 Customers

In PowerTAC simulation system the customers are the entities that buys and sells energy. A customer subscribes to one of the tariffs of the brokers and it pays or sells energy according the tariff plan. A customer can represent a population size of one to several thousands. For example, customers that represent a Electric Vehicle represent only one person and the customers that represent a village usually have several thousand population. In PowerTAC environment there are 168 customers.

3.2 PowerTypes

A customer can have among powertype among some possibility. Powertype determines the behaviour of the customers. A customer that has powertype related to production produces energy. A customer that has a power type related to energy consumption usually consumes energy. In the following subsections I describe powertypes of the customer.

3.2.1 consumption

A customer with powertype consumption are the most common customers. They use the energy when they need it. They cannot shift their demand to a future timeslot. Usually they have a regular pattern in their energy usage. Usually they show similar pattern for weekdays. They have similar kind of usage pattern for the weekends.

The figure 3.1 shows 2 days electricity usage of the BrookSideHomes customer. The pattern shows in a day, around at 10 am there is a growing need for electricity. During night after 10 pm the electricity consumption starts decreasing.

The figure 3.3 shows two weeks consumption of the downtown customer. The customer shows similar pattern for all weekdays. It also distinguishable energy usage during the weekends.

3.2.2 Interruptible Consumption

Interruptible customers are smart enough to shift their energy demand in a timeslot where they can buy electricity in a reduced price. Because of this shifting capability, they don't

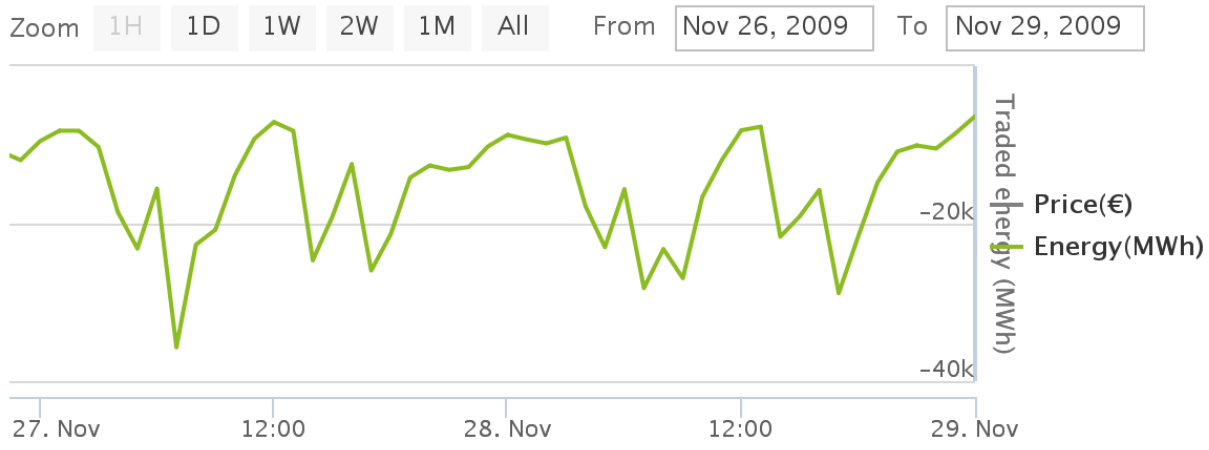


Figure 3.1: Two days energy usage for the customer Brooksidehomes.

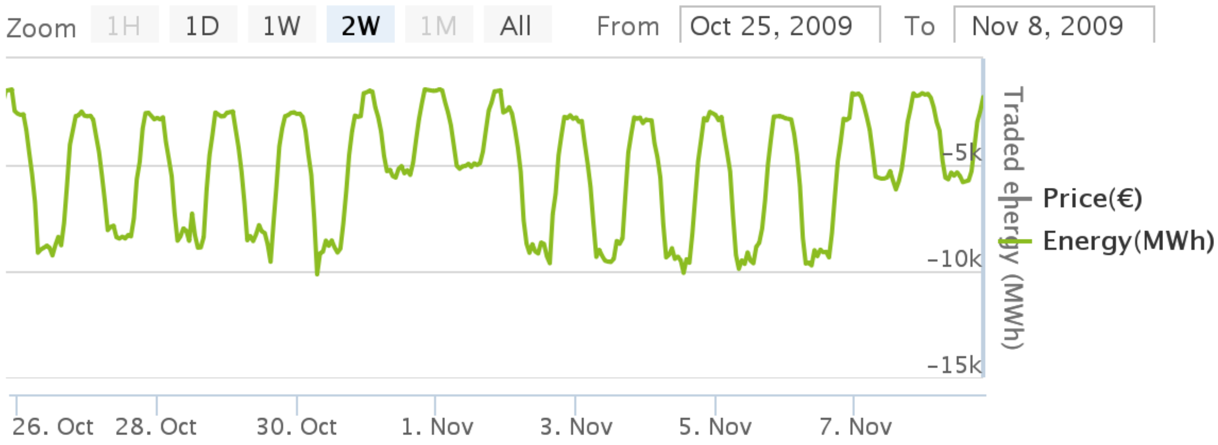


Figure 3.2: Two weeks energys usage of the downtown office customer.

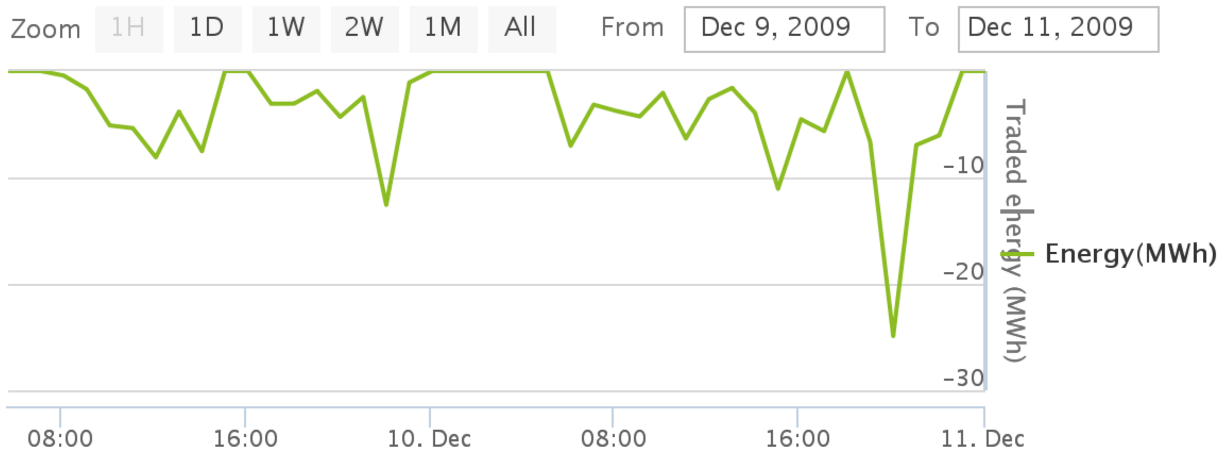
show the regular usage pattern as the consumption customers do. Figure 3.3 shows a controlloable customer's 2 days usage.

3.2.3 Thermal Storage

Thermal storage customers shows weekly pattern in their energy usage. Their energy usage in a day depends very much on the energy they used in the last timeslot. Figure 3.4 and 3.5 shows a day and two week's energy usage of the thermal storage customer sf2.

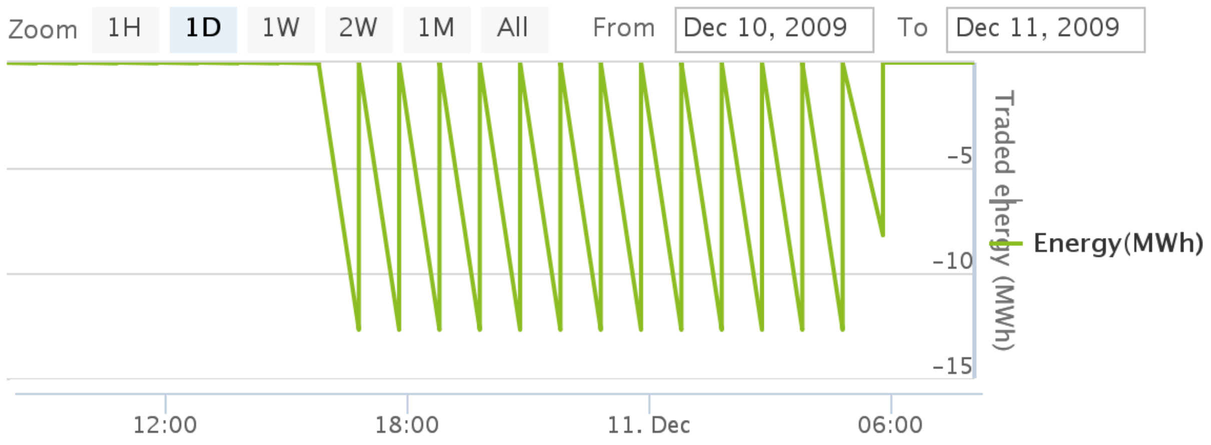
3.2.4 Solar Production

Figure 3.6 shows two day's and figure **??** shows a week's energy produciton of the Solar Production customer of the customer SunnyHill solar production customer.



From Dec 9, 2009 To Dec 11, 2009

Figure 3.3: Two days energys usage of the village 2 ns controllable customer.



From Dec 10, 2009 To Dec 11, 2009

Figure 3.4: A day's energys usage of the sf2 thermal storage customer.

3.2.5 Wind Production

Wind production customers generates energy from the wind.

3.2.6 Electric Vehicle

A electric vehicle customer represnt one electric vehicle. Their usage of energy is quite irregular and hard to predict.

3.3 Statistics

In this section I present some statistics on the customers available in the system.

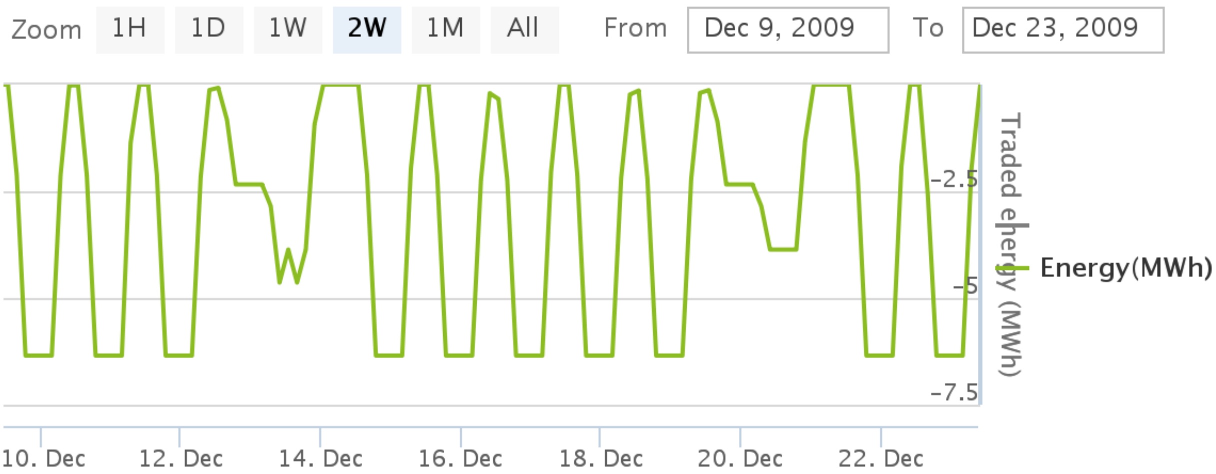


Figure 3.5: Two week's energys usage of the sf2 thermal storage customer.

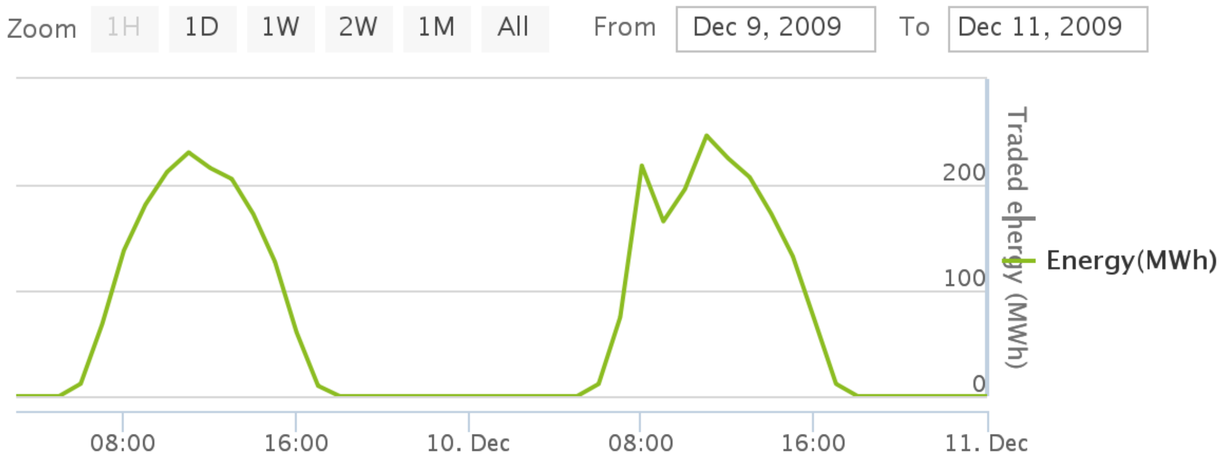


Figure 3.6: Two days energys usage of the SunnyHill solar production.

3.3.1 Customer Vs PowerType

In the figure 3.8 we can see the system has more customer with the power type electric vehicle than any other powertypes. This is because, the electric vehicle represents a population of size 1.

3.3.2 Population Vs PowerType

From figure 3.9 by far the powertype of consumption has the most number of people in them.

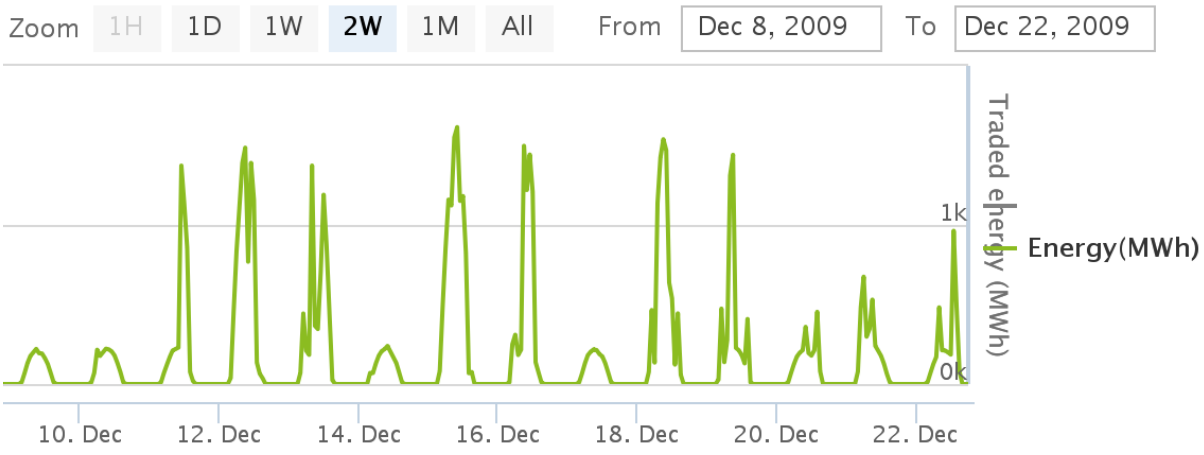


Figure 3.7: One week's energys usage of the SunnyHill solar production.

3.3.3 Total Energy Consumed Vs PowerType

From figure 3.10 we can see that the consumption type customers uses the most amount of the electricity.

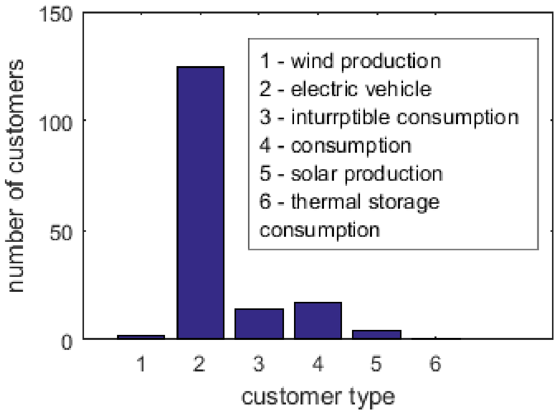


Figure 3.8:
Number of customers vs Powertype.

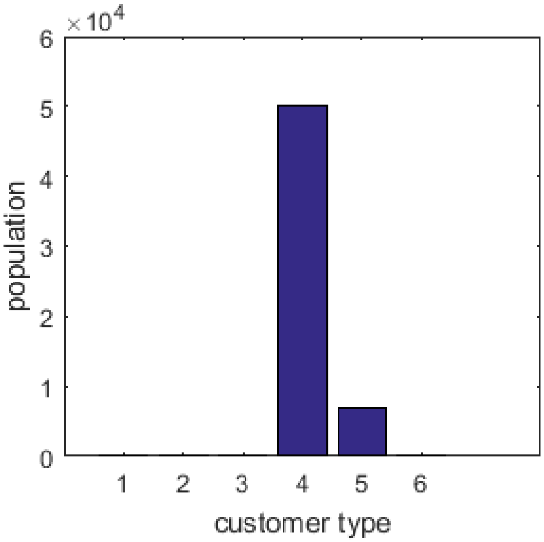


Figure 3.9: Population vs Powertype

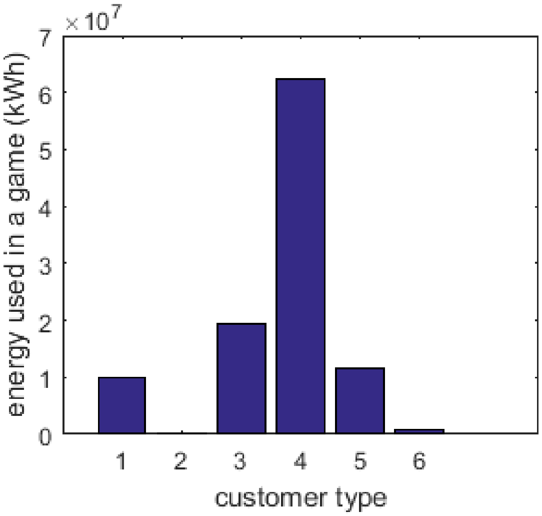


Figure 3.10: Energy vs PowerType.

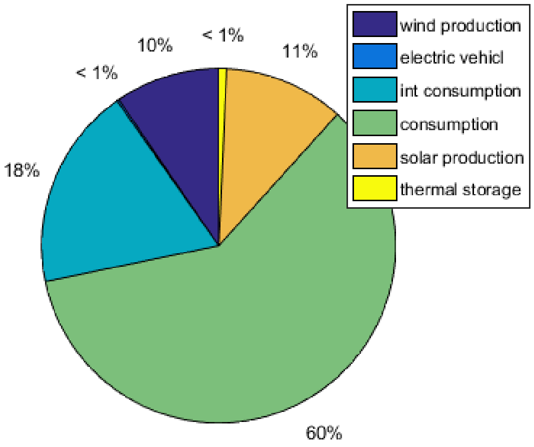


Figure 3.11:
Energy share for each power type.

Chapter 4

Methodology and Result

Traditioanlly, a single type of predictor served to predict the energy demand of all powertype customers. Since each powertype customers acts differently, I have attempted to attack each type of customer separately to make a prediction mechanism that perfroms better than the baseline predictor.

4.1 Baseline Predictor

Baseline predictor is the default prediction mechanism provided by the PowerTAC system. It exploits the fact that usage of a timeslot of a customer in a specific date is highly correlated with the day of the week and the time slot. To make prediction it stores the average energy usage of an hour of a week. So, for each customer it uses $24 * 7 = 168$ memory to remember average usages. As soon as it learns about a new usage information of an hour of a week, it updates old average using the following algorithm.

Algorithm 1 Update average usage for $customer_i$ for day d and timeslot t, $newUsage$

```
avgUsage = get average usage of $customer_i$ at day d and time slot t
$avgUsage = 0.7 * avgUsage + 0.3 * newUsage$
```

Algorithm 2 predict usage for day d and timeslot t for $customer_i$

```
avgUsage = get average usage of $customer_i$ at day d and time slot t
return $avgUsage$
```

There will be another type of predictor that is designed to make prediction for a single customer. In general, if there are n customers in the system, we will need n predictor each one trained on a single customer. I went further by checking different machine learning algorithms such as M5Tree, Linear Regression, M5P rules and REP tree for each customer and picked the best performing one for each customer.

4.2 Prediction Mechanism

In this section I will describe how I attempted to make predictor for each of the powertypes.

4.2.1 Consumption Type Customer

For the consumption type customers, the algorithm 3 describes the proposed method of forecasting energy demand using different methods.

Algorithm 3 Make prediction for consumption type customer

```
Extract features for each time slot for each customer [algorithm 4, 5 and 6 ]
Train kmeans cluster for different sizes of k [algorithm 7]
Train linear regression classifier for each cluster and compute error [algorithm 8]
pick suitable value for k observing on error
For each cluster, find the best performing predictor for that cluster [algorithm 9]
train individual classifer for each customer to make the second baseline [algorithm 10]
evaluate performance using test data [algorithm 11]
```

The algorithm 3 begins with extracting information from game log file. All the activities that occured in a game can be found in a game log. In power TAC the shortest time unit is an hour and it is called time slot. Activities such as buying or selling are occured during a time slot. At the beginning of a time slot the system notifies the broker that a new time slot is about to begin. The system also notifies the broker weather forecast about the future time slots. As a time slot ends, the broker receives information about its customers energy usage which is called tariff transaction report. Algorithm 4 refers how the extraction program retrieves necessary information from tariff transaction report. As the broker gets notification of the beginning of a new time slot, the extraction program has all the information related to energy usage and weather data of the previous time slot available by this time. Algorithm 5 shows the procedure of writing the information of the known time slot's information in training instance file. Once the simulation ends, the extraction program knows the average energy usage of all the customers during a week. In a week there are 24 * 7 = 168 hours or time slots. The extraction program writes all the 168 hourly averages of a week to a file. This is explained in algorithm 6.

Algorithm 4 extract information from transactionReport sent to broker after each time slot through TariffTransactionHandler call back method

```
timeSlot = get time slot from transactionReport
customerName = get customer name from transactionReport
energyUsed = get energy used from trom transactionReport
addUsage(customerName, timeSlot, energyUsed)
```

Next, all the average weekly usages are combined together to make training set for clustering algorithm. I have used kMeans clustering algorithm to cluster the training set. I have trained clusters of size = 4, 5, 6, 7, 8, 9, 10 and 11. The algorithm 7 describes the procedure of making cluster from the training instances. Once a kMeans of cluster size k is made, a program groups the hourly usages of the customers in the same cluster and combines them to make training instance for a machine learning classifier. This training

Algorithm 5 write extracted data after timeSlot update message received from TimeSlotUpdateHandler call back method

```
knownTimeSlot = timeSlot - 1
for each customer do
    day = get day of knownTimeSlot
    hour = get hour of knownTimeSlot
    statisticalData = get statistics of the customer of day and hour
    weatherData = get weather data of knownTimeSlot
    trueUsage = get true usage of customer in knownTimeSlot
    trainingInstance = create training instance by combining statisticalData, WeaterData and trueUsage
    writeToFile(trainingInstance)
end for
```

Algorithm 6 write average usage of customer of each hour of the week

```
Require: information of all timeslots have been received
for each customer do
    trainingInstance = create empty training instance
    for each day of week do
        for each hour of day do
            averageUsage = get average usage of day and hour of customer
            append averageUsage to the trainingInstance
        end for
    end for
    writeToAvgUsageFile(trainingInstance)
end for
```

set is used to train linear regression classifier. To test the performance of the classifiers, I have separated five game logs and they were not used for training purposes. The algorithm 8 describes how the cluster based predictor's performance was evaluated.

Algorithm 7 create kmeans cluster of size k from weekly usage training instance file

```
data = load weekly average usage file
kmeansCluster = build kmeans cluster of size k
save kmeansCluster
```

Based on the errors observed from different kMeansCluster based predictions, I fixed the number of clusters. Once the number of the clusters was fixed, a program creates creates several machine learning predictors to see which one performs best for a given cluster. The machine learning classifiers that were tried out are linear regression, M5P rules, M5 Tree, REP tree. In the runtime, a customer will be grouped in a cluster based on its weekly usage. Once the program knows the cluster assigned to a customer, the program will load the corresponding predictor to predict about the customer.

At this phase, I had the proposed cluster based customer demand predictor. Next,

Algorithm 8 find error of kmeans clusters of different size

```
for each cluster size k do
    get the kMeansCluster of size k
    for cluster in KMeansCluster do
        combine slot based training instances of that cluster
        train linear regression classifier based on the combined data
        save the classifier for cluster
    end for
end for
for each training instance do
    compute error of the instance using each kMeansCluster
end for
```

Algorithm 9 find best classifiers of each cluster of kmeans cluster of size k

```
for each cluster in kMeansCluster do
    combine slot based data of the all the customers in cluster
    train available classifiers on the combined data using 10 fold cross validation
    choose the classifier with minimum error
    save the classifier for making prediction for cluster
end for
```

the baseline predictor that needs a machine learning classifier for each customer is built. At first the training instances are combined based on the name of the customer. This means for n customers n training set is constructed, each of the training set has only the information of a single customer. A training set related to a customer is used to create machine learning classifiers that customer. Several classifiers had been tried out to figure out which classifier performs the best for a customer. The best performing classifier was choosen to predict about a customer. Algorithm 10 explains the procedure of getting the best classifier.

Algorithm 10 find best classifiers created for each individual customer

```
for each customer do
    combine all slot based training instance of the customer
    train available classifiers on the combined data using 10 fold cross validation
    choose the classifier with minimum error
    save the classifier for making prediction about the customer
end for
```

Next phase is testing the performance of the proposed and baseline methods. For testing, I had used five game logs that were not used for training purposes. For each test instance, all three methods output was observed to figure out the performance. The algorithm 11 shows the mechanism of testing.

Algorithm 11 performance evaluation of each method

```
for each test instance do
    classify the test instance using moving average usage [algorithm 2]
    classify the test instance using individual prediction mechanism
    classify the test instance using cluster based predictor
    find and accumulate errors of each mechanism [algorithm12]
    update moving average baseline predictor based on the information from the test instance [algorithm 1]
end for
find averageError from the accumulated errors
```

Algorithm 12 calculate error from predictedValue and trueValue

```
absoluteError = abs(predictedValue - trueValue)
relativeAbsoluteError = (absoluteError / trueValue ) * 100 %
```

4.3 Result

4.3.1 Finding number of clusters

At first, I have segmented the customer using KMeans clustering algorithm with cluster sizes = 4, 5, 6, 7, 8, 9, 10 and 11. For KMeans with size k, we will have k clusters. For each of the k clusters I had a linear regression predictor. I observed the relative percentage error and absolute average the above cluster sizes. Figure 4.1 shows the result. From, the figure it is clear that the size of the cluster does not have a big impact on the prediction performance.

To keep things simple, I have decided to choose Kmeans cluster of size 4. The figure 4.2 shows the assignments of customers in different clusters. From the figure, cluster-0 holds most of the offices, cluster 2 holds most of the village types, cluster 3 holds the medical center, cluster 1 holds large housing such as brooksidehomes, centerville homes etc.

4.3.2 Finding best predictor for each cluster

I have used the following features for a given timeslot to train prediction models.

- Temperature
- Cloud Cover
- Wind Speed
- Average of the Slot
- Standard Deviation of the Slot

Next, I have tried out M5Tree, Linear Regression, M5P rules and REP tree machine learning algorithms to see which one performs best for each of the 4 clusters. Figure 4.3,

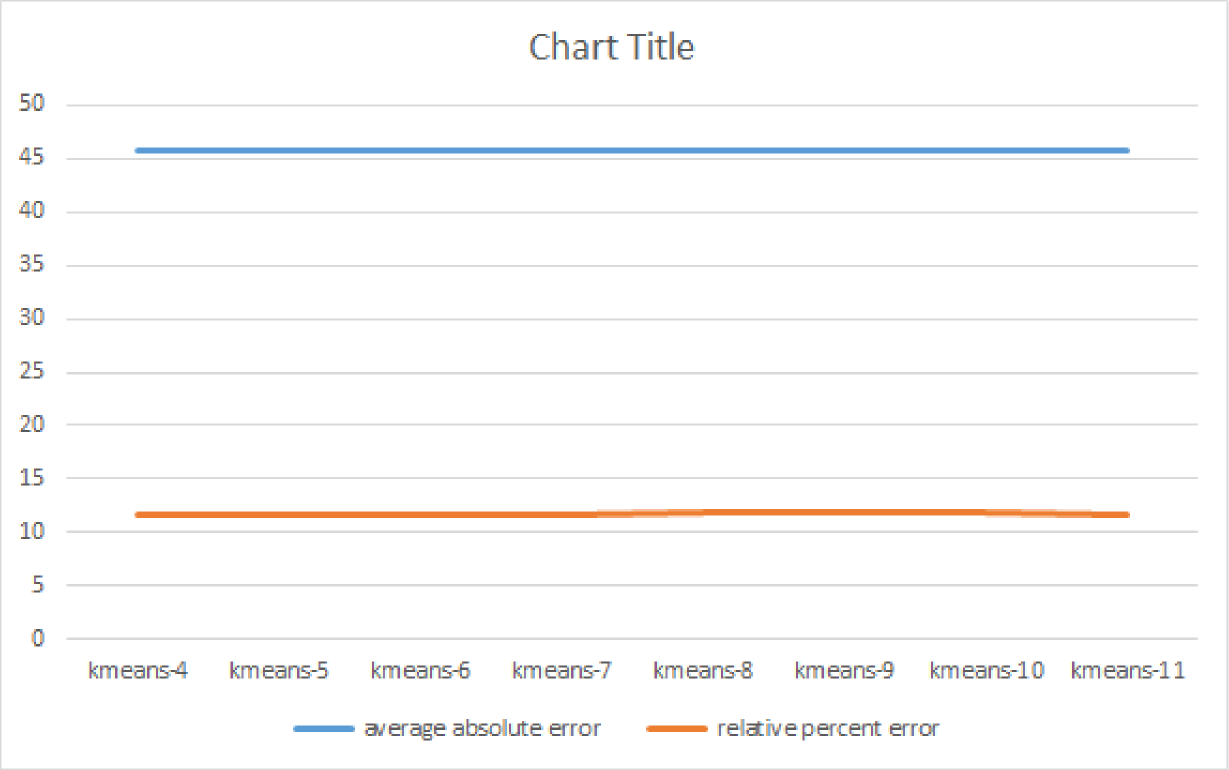


Figure 4.1: Cluster type vs Error.

4.4, 4.5, 4.6 show that M5P, M5P, REPTree and M5RULES are the best predictors for cluster 0, 1, 2 and 3 respectively.

The next step is to find the best predictors for each of the customers. Based on the data from each of the customers, the above four types of predictors were tried out. For each customer, the following predictors performed the best.

The figure 4.7 shows error percentage of each of the predictors type for each of the customer types.

Finally, the cluster based prediction and the two baselines were tested with data extracted from 5 test files that were not used for training. From Figure 4.8 we can see that cluster based prediction mechanism performed almost as well as the mechanism where n predictors are needed for n customers. And it did well than the moving average prediction scheme.

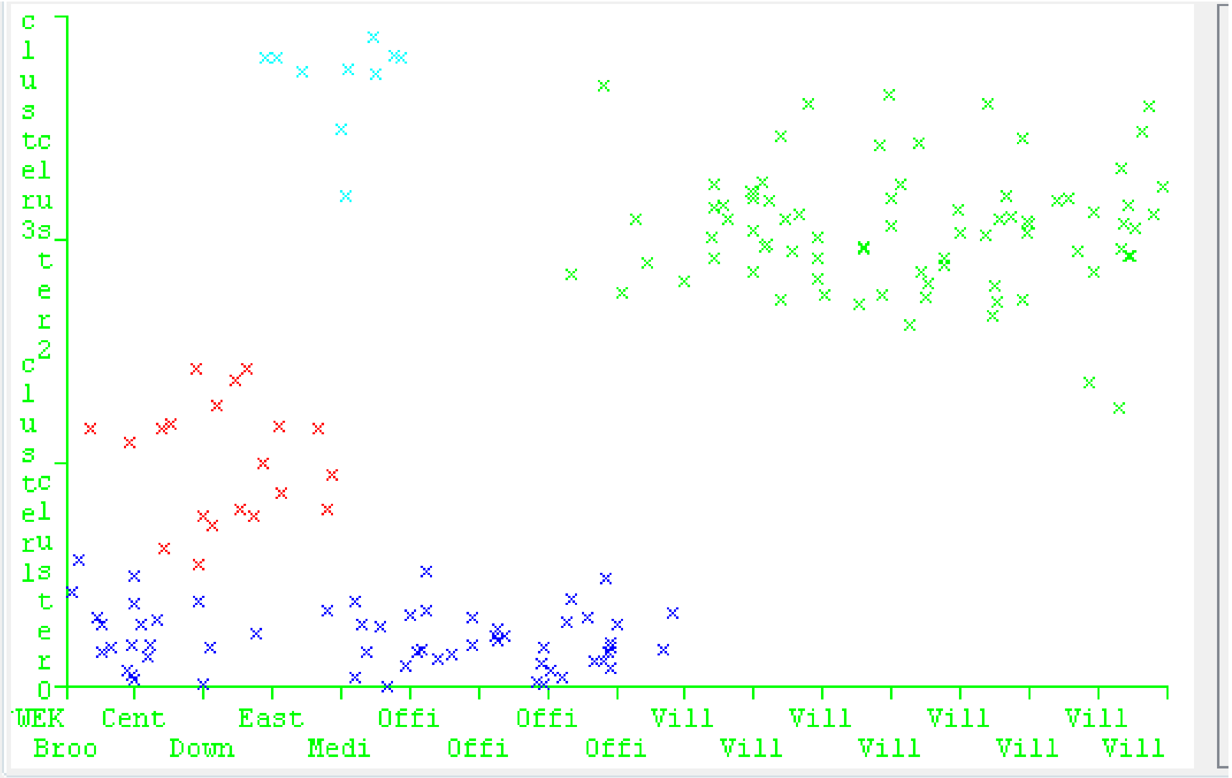


Figure 4.2: Cluster assignments.

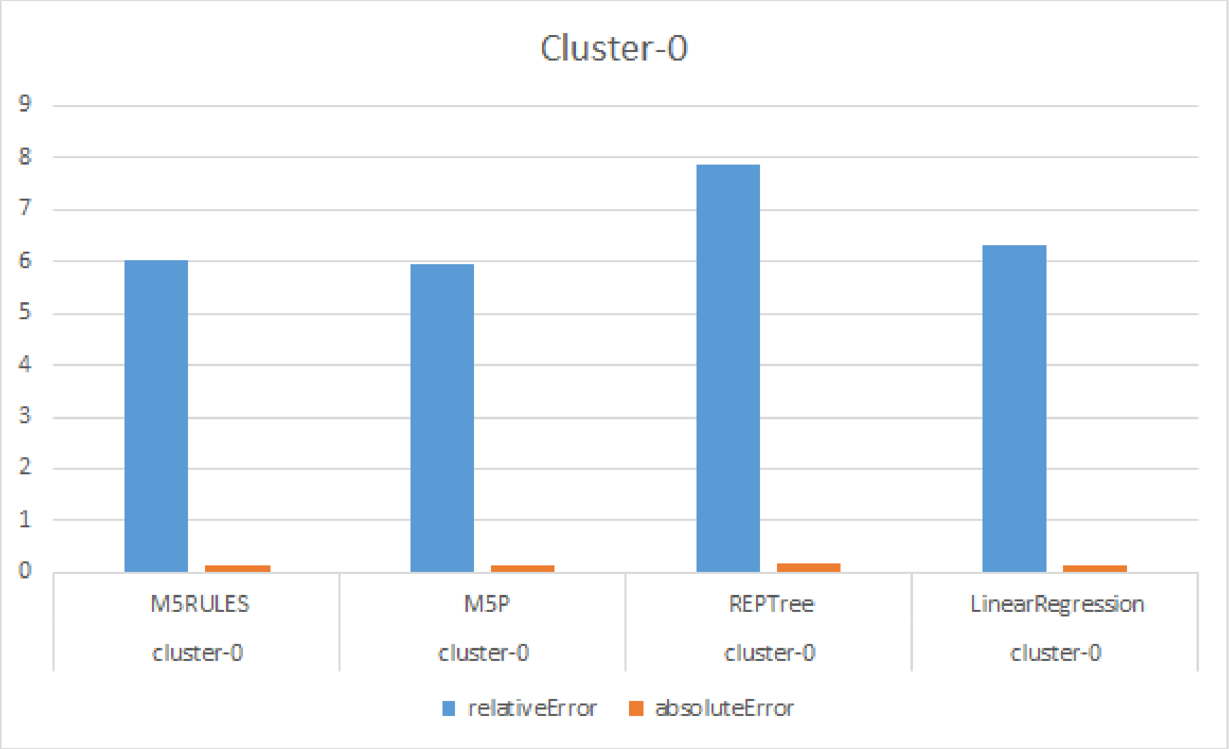


Figure 4.3: Performance of differenc predictors for cluster 0

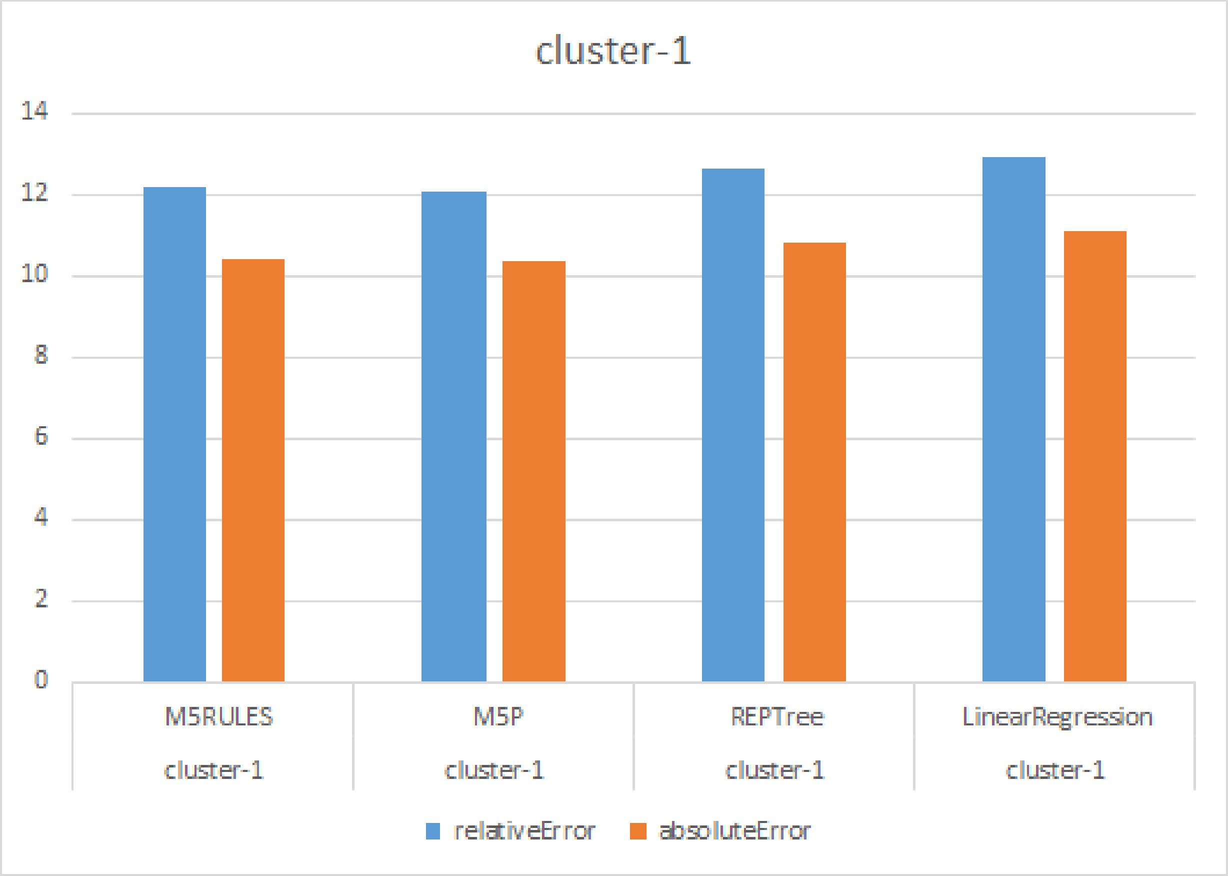


Figure 4.4: Performance of differenc predictors for cluster 1

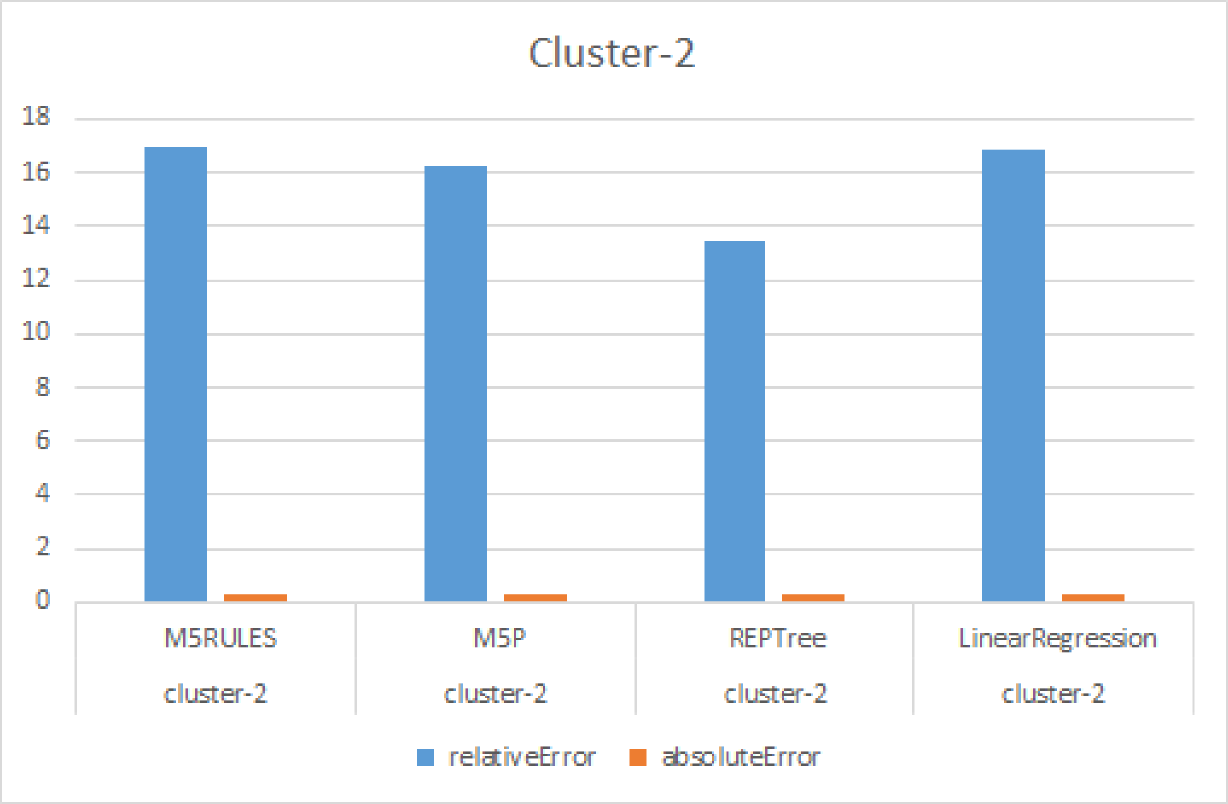


Figure 4.5: Performance of differenc predictors for cluster 2

Customer Name	Best Predictor Type
BrooksideHomes	M5P
CentervilleHomes	M5P
DowntownOffices	M5P
EastsideOffices	M5P
OfficeComplex 1 NS Base	LinearRegression
OfficeComplex 1 SS Base	LinearRegression
OfficeComplex 2 NS Base	LinearRegression
OfficeComplex 2 SS Base	LinearRegression
Village 1 NS Base	M5P
Village 1 RaS Base	LinearRegression
Village 1 ReS Base	M5P
Village 1 SS Base	M5P
Village 2 NS Base	LinearRegression
Village 2 RaS Base	M5P
Village 2 ReS Base	M5P
Village 2 SS Base	M5P
MedicalCenter@1	M5P

Table 4.1: Best individual predictor for each customer

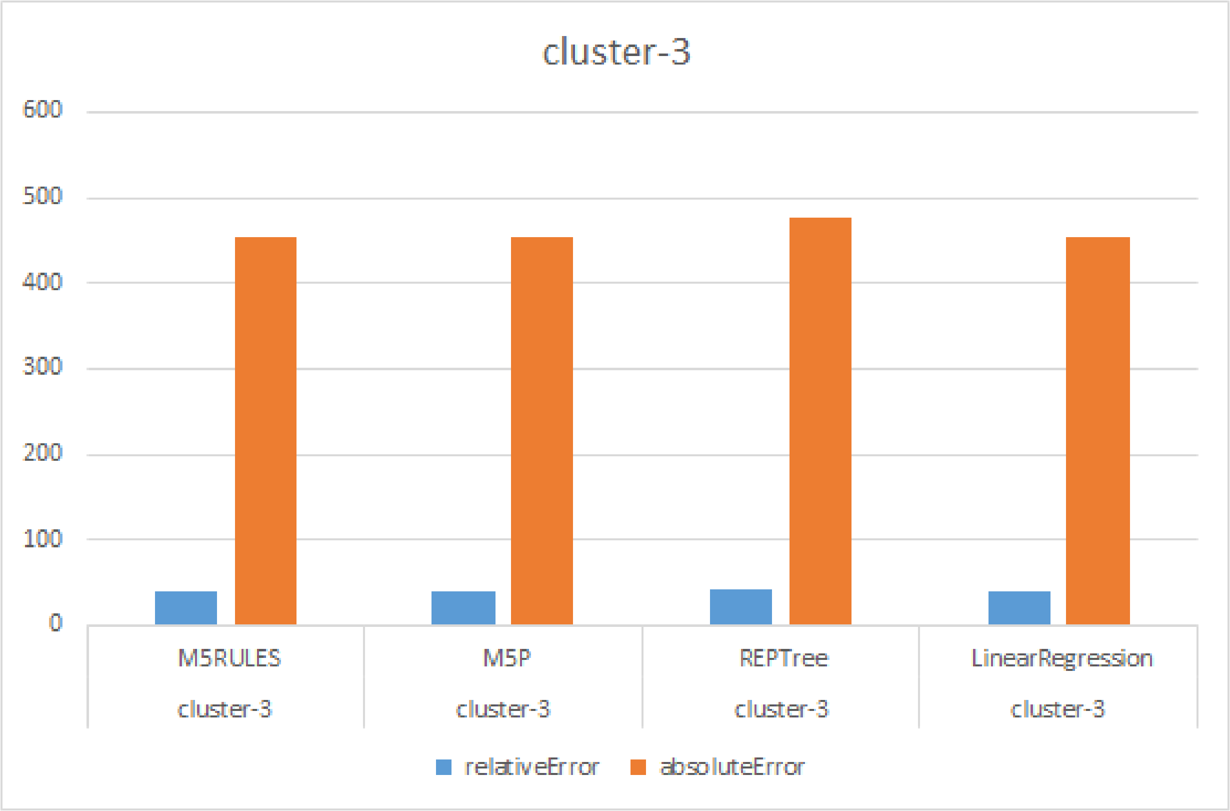


Figure 4.6: Performance of differenc predictors for cluster 3

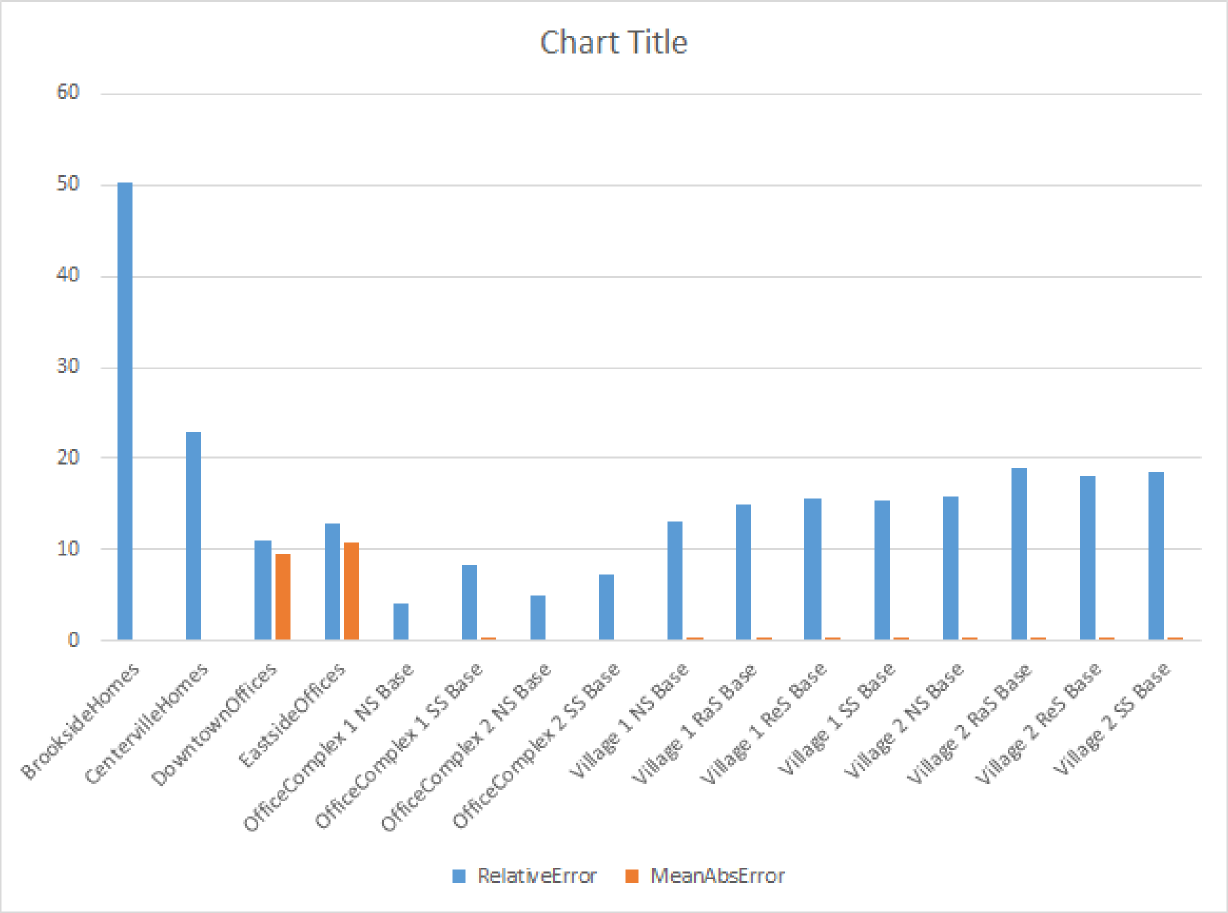


Figure 4.7: Performance of the best predictors for each customer type. Customer Medical center was excluded as it was showing huge error.

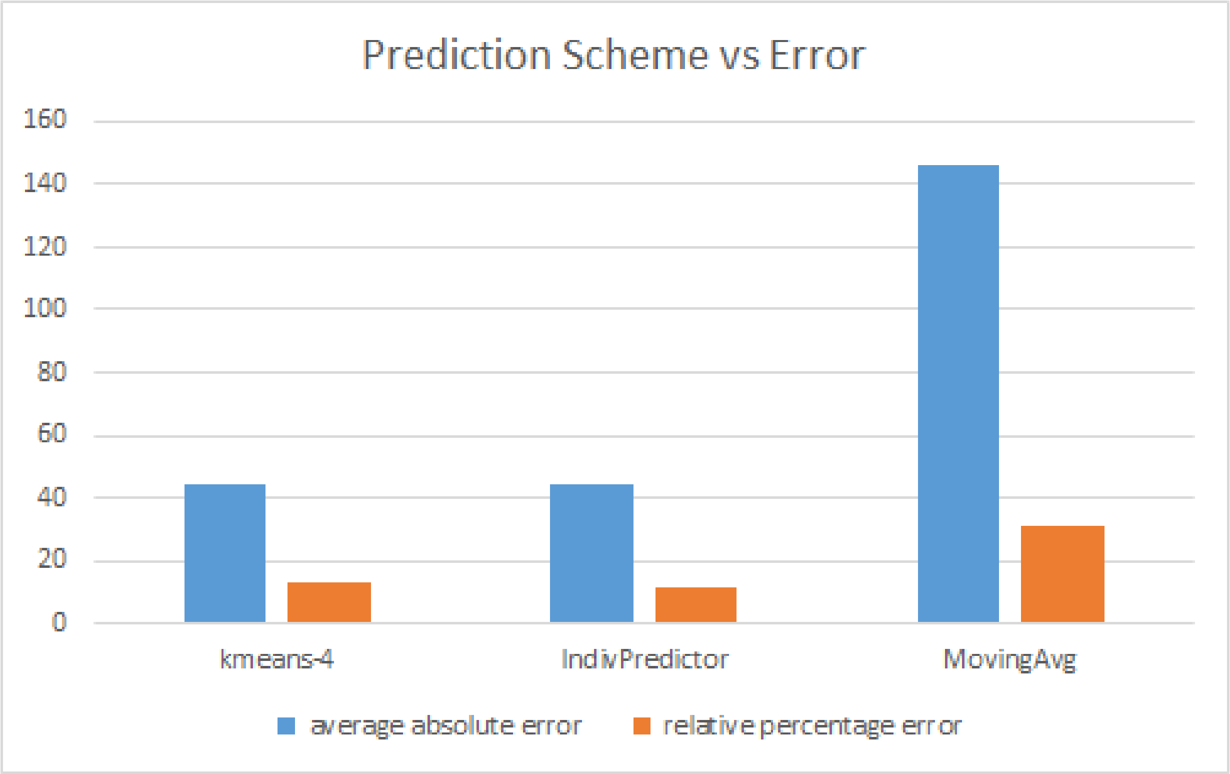


Figure 4.8: Performance of the three prediction mechanisms. Cluster based predictor performs as good as the individual predictor for each customers and performs better than the moving average predictor.

References

[1] HM Al-Hamadi and SA Soliman. Short-term electric load forecasting based on kalman filtering algorithm with moving window weather and load model. *Electric power systems research*, 68(1):47–59, 2004.

[2] Bo-Juen Chen, Ming-Wei Chang, and Chih-Jen Lin. Load forecasting using support vector machines: A study on eunite competition 2001. *Power Systems, IEEE Transactions on*, 19(4):1821–1830, 2004.

[3] MY Cho, JC Hwang, and CS Chen. Customer short term load forecasting by using arima transfer function model. In *Energy Management and Power Delivery, 1995. Proceedings of EMPD'95., 1995 International Conference on*, volume 1, pages 317–322. IEEE, 1995.

[4] Xi Fang, Satyajayant Misra, Guoliang Xue, and Dejun Yang. Smart gridthe new and improved power grid: A survey. *Communications Surveys & Tutorials, IEEE*, 14(4):944–980, 2012.

[5] Heiko Hahn, Silja Meyer-Nieberg, and Stefan Pickl. Electric load forecasting methods: Tools for decision making. *European Journal of Operational Research*, 199(3):902–907, 2009.

[6] Melissa Hart and Richard de Dear. Weather sensitivity in household appliance energy end-use. *Energy and Buildings*, 36(2):161–174, 2004.

[7] Che-Chiang Hsu and Chia-Yon Chen. Regional load forecasting in taiwan—-applications of artificial neural networks. *Energy conversion and Management*, 44(12):1941–1949, 2003.

[8] Fang Liu, Qiang Song, and Raymond D Findlay. Accurate 24-hour-ahead load forecasting using similar hourly loads. In *Computational Intelligence for Modelling, Control and Automation, 2006 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, International Conference on*, pages 249–249. IEEE, 2006.

[9] Fintan McLoughlin, Aidan Duffy, and Michael Conlon. A clustering approach to domestic electricity load profile characterisation using smart metering data. *Applied energy*, 141:190–199, 2015.

[10] Jaime Parra Jr and Christopher Kiekintveld. Initial exploration of machine learning to predict customer demand in an energy market simulation. In *Workshops at the Twenty-Seventh AAAI Conference on Artificial Intelligence*, 2013.

[11] Cameron W Potter, Allison Archambault, and Kenneth Westrick. Building a smarter smart grid through better renewable energy information. In *Power Systems Conference and Exposition, 2009. PSCE'09. IEEE/PES*, pages 1–5. IEEE, 2009.

[12] Saifur Rahman and Rahul Bhatnagar. An expert system based algorithm for short term load forecast. *Power Systems, IEEE Transactions on*, 3(2):392–399, 1988.

[13] Andre Richter, Erwin van der Laan, Wolfgang Ketter, and Konstantina Valogianni. Transitioning from the traditional to the smart grid: Lessons learned from closed-loop supply chains. In *Smart Grid Technology, Economics and Policies (SG-TEP), 2012 International Conference on*, pages 1–7. IEEE, 2012.

[14] Xishun Wang, Minjie Zhang, Fenghui Ren, and Takayuki Ito. Gongbroker: A broker model for power trading in smart grid markets. In *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, volume 2, pages 21–24. IEEE, 2015.

Appendix A

Some more stuff

This is an example of how to add an appendix.

Curriculum Vitae

Patrick Thor Kahl was born on July 12, 1961. The first son of Ulf Thor Gustav Kahl and Carolyn Kahl, he graduated from Coronado High School, El Paso, Texas, in the spring of 1979. He entered Auburn University in the fall of 1979, and, in the spring of 1982, The University of Texas at El Paso. In 1985 he joined the United States Navy where he served for eight years, most of it aboard the submarine USS Narwhal (SSN671). In the fall of 1993, after being honorably discharged from the navy, Patrick resumed his studies at The University of Texas at El Paso. While pursuing his bachelor's degree in Computer Science he worked as a Teaching Assistant, and as a programmer at the National Solar Observatory at Sunspot, New Mexico. He received his bachelor's degree in Computer Science in the summer of 1994.

In the fall of 1994, he entered the Graduate School of The University of Texas at El Paso. While pursuing a master's degree in Computer Science he worked as a Teaching and Research Assistant, and as the Laboratory Instructor for the 1995 Real-Time Programming Seminar at the University of Puerto Rico, Mayagüez Campus. He was a member of the Knowledge Representation Group and the Rio Grande Chapter of the Association for Computing Machinery.

Permanent address: 6216 Sylvania Way
El Paso, Texas 79912-4927