# Energy Aggregation using Product of Experts[*]

## Abstract

The *IJCAI–15 Proceedings* will be printed from electronic manuscripts submitted by the authors. The electronic manuscript will also be included in the online version of the proceedings. This paper provides the style instructions.

## 1 Introduction

Aggregation is a process in which data is gathered and used for statistical analysis. Energy aggregation is to collect data from multiple meters. The purpose of energy aggregation is to get some valuable information about single or multi-site units. In this paper, we are aggregating energy consumption information using product of HMMs such that the contrastive divergence between the probability distributions is minimized. This concept is applied on two datasets, REDD dataset and the faculty housing dataset generated by smart meters. The proof of concept of REDD dataset and faculty housing dataset is given in section 4 and section 5 respectively.

## 2 Related Work

### 2.1 Automata and their products

Distributed networks can be modelled using interacting automata. Benveniste [Fabre *et al.*, 2000] defines automaton as a quadruple, $Á = (X, X_0, A, T)$ where X is a finite state of sets, X0 is the subset of initial states, A is a finite set of messages, T is a set of transitions of the form $t = \{x_-, a, x\}$ where $x_-$ is the previous state, a is the message label on which the state transitions to the next state x. The figure 1 below explains the automata with an example.

For automaton R, $X_R = \{2; R1, R2\}$, $X_{0R} = \{R1\}$, $A_R = \{a, b\}$, $T_R = \{R1, a, R1; R1, b, R2; R2, a, R2; R2, b, R1\}$
For automaton S, $X_S = \{3; S1, S2, S3\}$, $X_{0S} = \{S1\}$, $A_S = \{a, b\}$, $T_S = \{S1, a, S1; S1, b, S2; S2, a, S2; S2, b, S3; S3, a, S3; S3, b, S1\}$
The product of two automata $Á = R \times S$ is defined as follows:
$X = X_R \times X_S$

$X_0 = X_{0R} \times X_{0S}$
$A = A_R \cup A_S$
Benveniste uses a notion of stuttering transition which helps to distinguish between local and global time by inserting dummy transitions between two transitions of a local automaton attached to a node. This stuttering transition waits for others to progress.

| A | R1 | R2 |
|---|---|---|
| R1 | 0.6 | 0.4 |
| R2 | 0.3 | 0.7 |

Table 1: Transition probability, A

| B | a | b |
|---|---|---|
| R1 | 0.2 | 0.8 |
| R2 | 0.5 | 0.5 |

Table 2: Observed probability, B

| | R1 | R2 |
|---|---|---|
| $\pi$ | 0.4 | 0.6 |

Table 3: Initial state probability, $\pi$

Talking in terms of HMM, requires us to equip products of automata with probabilities. Benveniste defines HMM as a triple $(Á, \mu, \pi)$ where $Á = (X, X_0, A, T)$ is an automaton, $\mu$ is the initial state probability, $\pi$ is factored as state transition probability $\pi_x$ and message transition probability $\pi_A$. He uses a random arbiter $\alpha$, with values first, second, third to choose automaton to initiate transition. If $\alpha$ = first then first automaton chooses any transition having a private message whereas second automaton performs a stuttering transition, and vice versa for $\alpha$ = second. If $\alpha$ = both, then both automata agree on some shared message and move accordingly.

Using the traditional HMM notation of the parameters $\lambda = \{A, B, \pi\}$ where A is the transition probability, B is the observed probability, $\pi$ is the initial state probability. For automata R, we have the values of A, B, $\pi$ as shown in table 1, 2, 3 respectively.
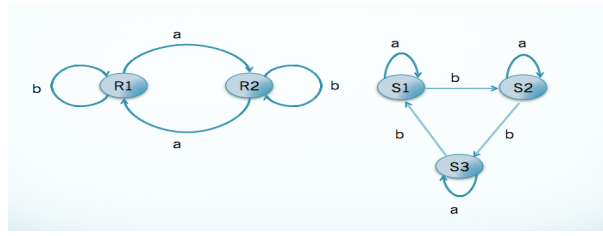
Figure 1: Automata R and S

## 2.2 Product of HMM

Product of HMM is a way of combining HMM's to form distributed state time series model. The figure 2 is a product of two HMMs shown in 1. For P = R x S, the quadruple becomes

$X = \{6; R1S1, R1S2, R1S3, R2S1, R2S2, R2S3\}$

$X_0 = \{R1S1\}$

$A = \{a,b\}$

The rules for synchronised product construction are :

1. $< p,q > -a-> < p',q >$ if $a \in A_R \cap A_S$ and $p -a-> p'$ and $q -a-> q'$
2. $< p,q > -a-> < p',q >$ if $a \in A_R, a \notin A_S$ and $p -a-> p'$
3. $< p,q > -a-> < p,q' >$ if $a \notin A_R, a \in A_S$ and $q -a-> q'$

## 3 Methodology

### 3.1 Training product of experts by minimising contrastive divergence

High dimensional distributions are approximated as a product of one dimensional distributions. The product of individual distributions which are uniguassian or multivariate guassian will also be multivariate guassian. If the individual models are more complicated and contain one or more hidden variables, multiplying their distributions together and renormalizing them can be very powerful. These individual models are called "experts". The product of experts produce sharper distribution than the individual distributions[Hinton, 2000].

## 4 Proof of concept on REDD House 2

### 4.1 Aim

To represent streams of energy consumption data from $n^1$ appliances by product of $k$ HMMs.

### 4.2 Method

- **Data** The Reference Energy Disaggregation Data Set (REDD) is used in empirical analysis. The data contains power consumption from real homes, for the whole house as well as for each individual circuit in the house (labeled by the main type of appliance on that circuit). It is intended for use in developing disaggregation methods, which can predict, from only the whole-home signal, which devices are being used. The REDD data set contains two main types of home electricity data:

high-frequency current/voltage waveform data of the two power mains (as well as the voltage signal for a single phase), and lower-frequency power data including the mains and individual, labeled circuits in the house. The main directory consists of several house directories, each of which contain all the power readings for a single house. Each house subdirectory consists of a labels and channels files. The labels file contains channel numbers and a text label indicating the general category of device on this channel. Each channel_i.dat file has two columns containing UTC timestamps (as integers) and power readings (recording the apparent power of the circuit) for the channel. Experiments reported here use the House 2 data from REDD. It has 11 channels where each channel corresponds to the following appliance:

1. mains_1
2. mains_2
3. kitchen_1
4. lighting
5. stove
6. microwave
7. washer_dryer
8. kitchen_2
9. refrigerator
10. dishwaser
11. disposal

The dataset has 318759 records and 2 columns. We randomly sample 300 records for our initial experiment. Time series data from two appliances are represented as product of $k$ HMMs.

- **Time Series :** The time series data of the microwave, dryer, kitchen_2 and refrigerator are plotted below in Figures 3, 4, 5, 6.

- **Code** The implementation of the product of experts model is obtained from Iain Murray's website[2]. It implements the technique described in Geoff Hinton's paper [Hinton, 2000].

- **Additional details** Some additional details regarding experiments:

  1. The product of HMMs model (PoHMM) minimizes "contrastive divergence" as described in the paper [Hinton, 2000].
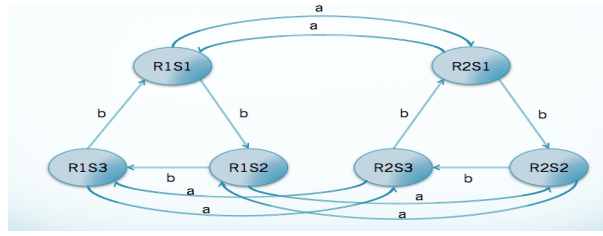
---
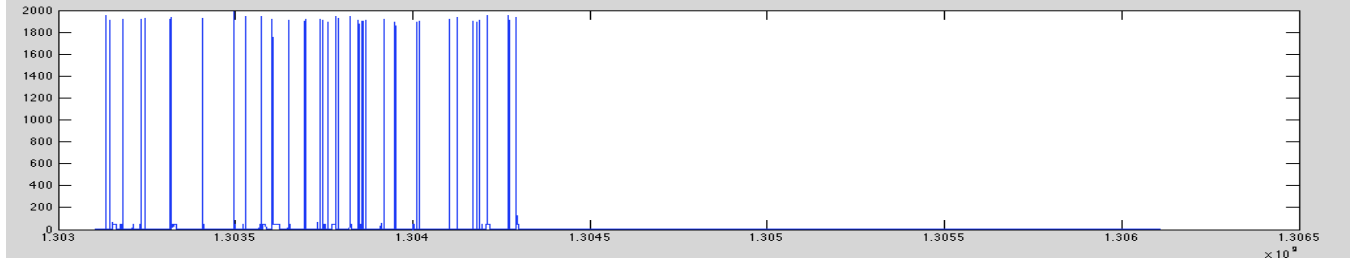
[1]n=2

Figure 2: Product of HMMs, P = R x S



Figure 3: Microwave

2. The number of experts, $k$ used here is 15. This is set somewhat arbitrarily and needs to be experimented on.

3. Learning rate is $\epsilon = \frac{1}{300}$.

### 4.3 Experimental Setup for REDD house 2

Experiments are performed on the REDD which contains $9$ appliances each containing $318759$ rows of energy consumption data. Experiments are done into $4$ phases, in the first phase the number of data samples are varied corresponding to which the values of KL Divergence and convergence time are noted. In the second phase, the number of experts are varied keeping the best value of the sample from the first phase fixed. In the third phase, number of iterations are varied keeping the best values from above first two phases fixed. In the fourth part, the no. of appliances to be aggregated are varied.

| Samples | KLDiv | T(sec) | Iterations |
|---|---|---|---|
| 300 | 2.4864 | 186.212 ±9.087 | 18600 |
| 500 | 0.6761 | 106.564 ±10.046 | 10200 |
| 1000 | 1.1088 | 158.521 ±1.97 | 11200 |
| 1500 | 3.8829 | 92.896 ±8.075 | 5300 |
| 2000 | 1.8686 | 130.98 ±1.932 | 6900 |
| 2500 | 0.4733 | 215.563 ± 2.471 | 9900 |
| 3000 | 2.8204 | 258.213 ±1.918 | 11000 |
| 3500 | 1.2332 | 204.661 ±1.713 | 7900 |
| 4000 | 0.8959 | 292.666 ±0.619 | 10400 |
| 4500 | 1.1118 | 222.558 ±1.967 | 7200 |
| 8000 | 6.392 | 381.635 ±2.952 | 8100 |
| 10000 | 8.276 | 887.932 ±13.824 | 10500 |
| 15000 | 0.7201 | 1368.514 ±13.605 | 9400 |

Table 4: Effect of varying samples on KL div and time

| Experts | KLDiv | T(sec) | Iterations |
|---|---|---|---|
| 5 | 0.774 | 72.968 ±1.177 | 5200 |
| 10 | 1.424 | 117.482 ±1.966 | 6700 |
| 15 | 0.473 | 210.249 ±1.258 | 9900 |
| 20 | 1.56 | 217.739 ±10.452 | 9000 |
| 25 | 7.469 | 347.019 ±8.23 | 12100 |
| 30 | 2.4968 | 413.802 ±7.304 | 12900 |
| 35 | 1.5012 | 348.906 ±14.651 | 11300 |

Table 5: Effect of varying experts on KL div and time

### 4.4 Results

The evaluation of how well the learning has taken place is done by using a Kullback-Leibler divergence. KL divergence of P from Q, $D_{KL}(P\|Q)$ is the measure of information lost when Q is used to approximate P. Here, P is the real data and Q is a fantasy data. The two probability distributions in the REDD example refer to the expert probabilities in real and fantasy data. The learned parameters from the training are fitted to the fantasy data to measure the information lost when fantasy data is used to approximate real data.

| Threshold | KLDiv | T(sec) | Iterations |
|---|---|---|---|
| .1 | 0.473 | 210.6 ±1.493 | 9900 |
| .05 | 0.443 | 240.607±2.436 | 10900 |
| .01 | 0.454 | 431.536 ±14.509 | 18000 |
| .005 | 0.509 | 1167.243 ±43.412 | 49800 |

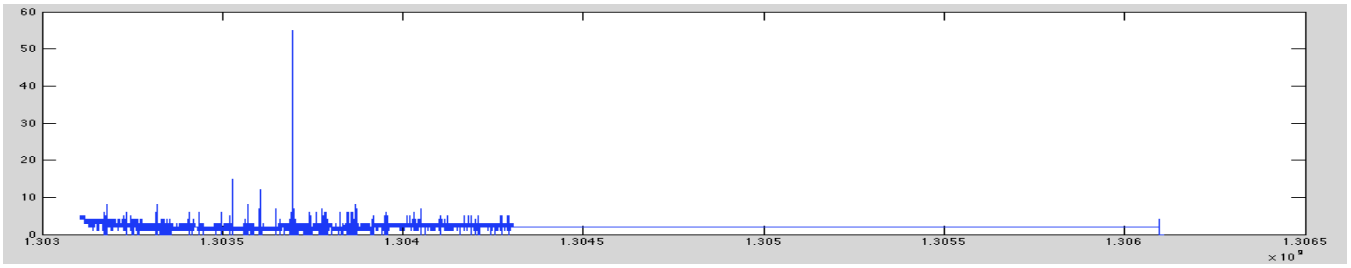Table 6: Effect of varying min threshold on KL div and time
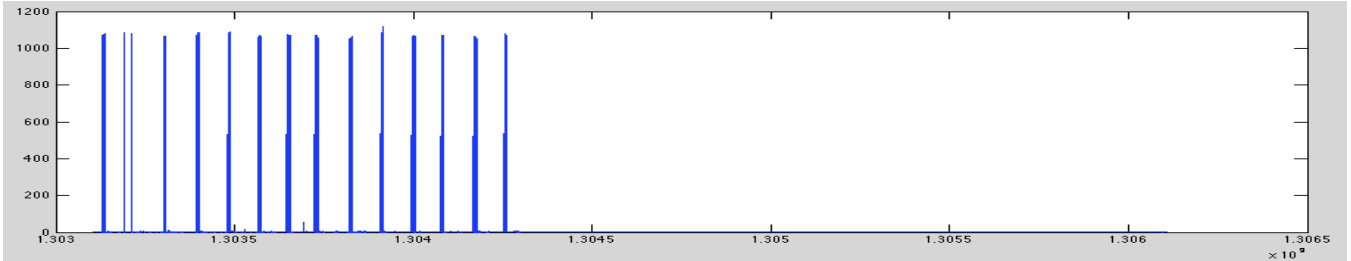
Figure 4: washer_dryer



Figure 5: Kitchen_2

| Appliances | KLDiv | T(sec) | Iterations |
|---|---|---|---|
| 3 | 5.559 | 233.664 ±0.579 | 10700 |
| 4 | 0.188 | 465.634 ±5.275 | 19900 |
| 5 | .432 | 338.416 ±3.988 | 13400 |
| 6 | 8.736 | 606.062 ±7.534 | 28100 |
| 7 | 5.054 | 411.457 ±10.051 | 17300 |
| 8 | 0.436 | 260.544 ±cc27.862 | 10700 |
| 9 | 0.15 | 474.579 ±14.619 | 20600 |

Table 7: Effect of varying appliances on KL div and time

## 5   Proof of concept on Faculty housing data

### 5.1   Aim

To represent streams of energy consumption data from all the floors of faculty housing as a product of k HMMs.

### 5.2   Method

- **Data** This data represents the energy consumed by the IIIT Delhi faculty housing building. As a part of research, a team from IIIT Delhi has installed various temperature, light and motion sensors to perform real world studies and to analyse user preferences for energy conservation. For our analysis, we selected one month's historical data ranging from 01-01-2014, 00:01 hours to 31-01-2014, 23:59 hours. The two smart meters installed captures the data from all the floors. The first meter gives out readings from floors 0 to 5 and the second meter gives out readings from floors 6 to 11. The dataset includes timestamp and power consumed in watts and 84133 records. Time series data from two streams are modelled as a product of $k$ HMMs. We also have the total power consumed by the faculty housing building

which would serve as the ground truth to compare product of $k$ HMMs with. The data is obtained from the website whose screenshot is shown in Figure 7

- **Code** It implements the technique described in Geoff Hinton's paper [Hinton, 2000].

- **Time Series :** The time series data of the energy consumption of floor 0 to 5, floor 6 to 11 and total power are plotted below in Figures 8, 9, 10.

- **Additional Details**

### 5.3   Experimental Setup

Each of the data stream is modelled as a HMM individually. There are three streams of data, the first stream D1 corresponds to the data from 0-5th floor, D2 corresponds to 6-11th floor and D3 represents the total power from the faculty housing which is represented by a fixed test set, T. Firstly, the stream D1 is used to train the model such that the contrastive divergence is minimized. The parameters (mixing component of each unigauss, means of gaussian bits, log precisions of axis-aligned gaussian bits) that are learnt during the training are provided to the test set T in order to obtain the conditional probability of the gaussians given the data D1 represented as pgauss1. Similarly, the second stream of data, D2 collected from floor 6-11, is used to learn the parameters of the model during the training phase which are then again provided to the test set T to obtain conditional probability of the gaussians given the data D2 as pgauss2. Finally the data D3 is used to learn the model and parameters which are then applied to the test set T to obtain the gaussian probability as pgauss3. Now, as we know that the total power consumption of the building should be approximately equal to the product of HMMs, which is the product of pgauss1 and pgauss2. If we can show that the value of pgauss3 is as close as possible to the product
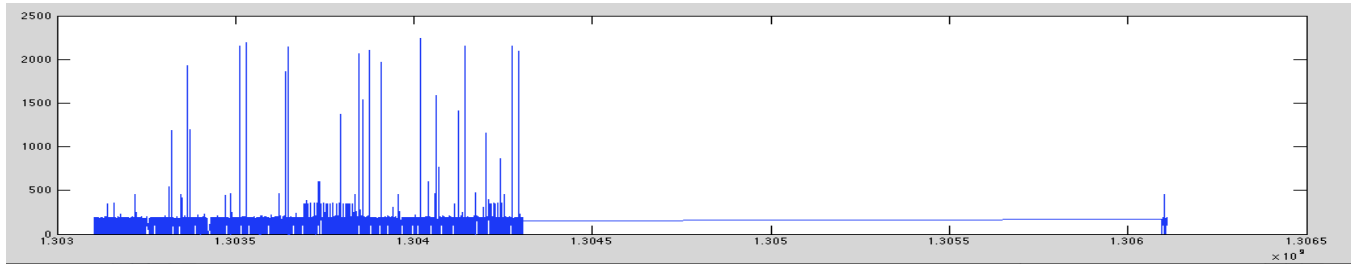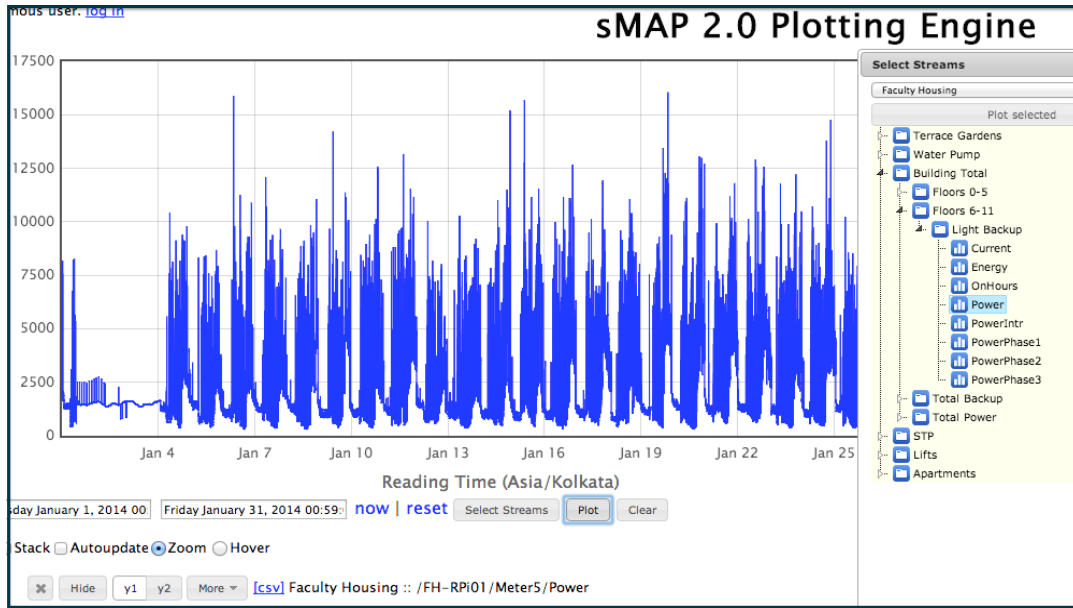
Figure 6: refrigerator



Figure 7: Screen shot of the webpage

of pgauss1 and pgauss2.

The experiments performed in table 8, shows the effect of varying samples on KL Divergence, convergence time and iterations keeping minimum threshold constant at 7.

The other experiment performed in table 9 shows that effect of varying experts on KL Divergence and convergence time.

### 5.4 Results

Table 8 shows that the error was minimum when the sample size was 300. With respect to the number of experts, the error was minimum when there were 5 experts as shown in table 9.

## 6 Conclusion & Future Work

The conclusion goes here. this is more of the conclusion

## References

[Brown and Hinton, 2001] Andrew Brown and Geoffrey Hinton. Proceedings of artificial intelligence and statistics 2001. In *Products of Hidden Markov Models*, number GCNU TR 2000-008, 2001.

| Samples | $KLDiv$ | $T(sec)$ | $Iterations$ |
|---------|---------|----------|--------------|
| 100 | 2.6219e-05 | 257 | 45100 |
| 300 | 1.9753e-05 | 222 | 43200 |
| 500 | 5.5493e-05 | 260 | 44800 |
| 700 | 3.2847e-05 | 249 | 44000 |
| 900 | 3.9486e-04 | 221 | 42600 |
| 1100 | 4.9274e-04 | 317 | 44700 |
| 1300 | 3.0425e-04 | 276 | 43100 |
| 1500 | 3.1128e-04 | 303 | 44400 |
| 2000 | 1.9192e-04 | 306 | 44400 |
| 2500 | 1.7122e-04 | 370 | 44100 |
| 3000 | 1.4686e-04 | 331 | 43300 |
| 3500 | 1.2663e-04 | 370 | 43200 |
| 4000 | 1.0793e-04 | 403 | 43200 |

Table 8: Effect of varying samples on KL div
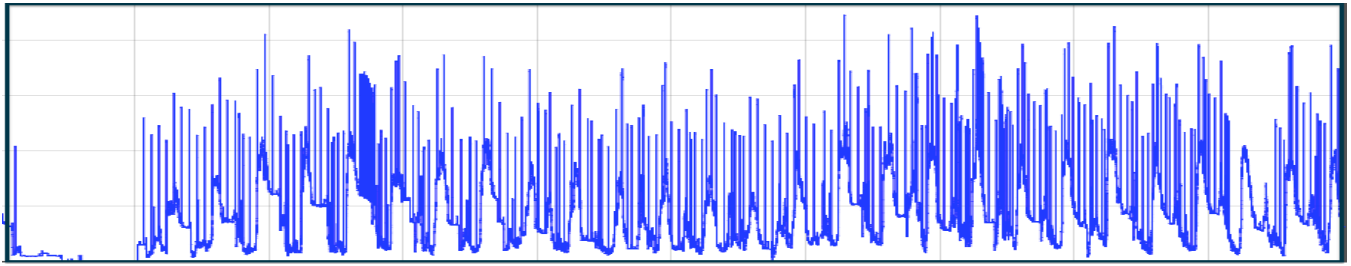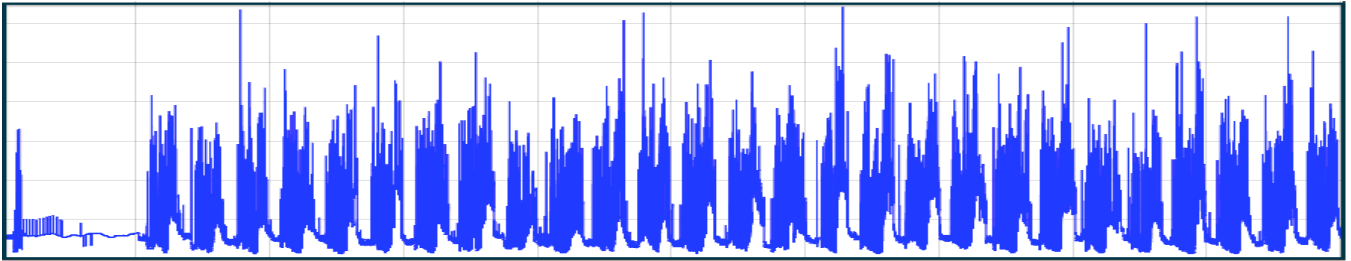
Figure 8: Stream 1: Power consumption of floors 0-5



Figure 9: Stream 2: Power consumption of floors 6-11

| Experts | $KLDiv(e-05)$ | $T(sec)$ |
|---|---|---|
| 3 | 1.9780 | 229 |
| 4 | 3.5897 | 217 |
| 5 | 1.9753 | 228 |
| 6 | 4.3488 | 238 |
| 7 | 4.9111 | 245 |
| 8 | 5.6564 | 241 |
| 9 | 5.4290 | 258 |
| 10 | 5.5163 | 267 |
| 12 | 4.4504 | 262 |
| 14 | 6.9006 | 296 |
| 16 | 6.8666 | 300 |
| 18 | 6.2872 | 313 |
| 20 | 5.3842 | 267 |
| 25 | 5.8970 | 326 |
| 30 | 5.9962 | 327 |
| 35 | 5.2716 | 346 |
| 40 | 5.0955 | 320 |

Table 9: Effect of varying experts on KL div and time

[Brown, 2001] Andrew Dennis Brown. Product model for sequences, 2001.

[Fabre *et al.*, 2000] Eric Fabre, Stefan Haar, Albert Benveniste, and Albert Benveniste. Hidden markov models for distributed and concurrent systems, 2000.

[Ghahramani and Jordan, 1997] Zoubin Ghahramani and Michael I. Jordan. Factorial hidden markov models. *Mach. Learn.*, 29(2-3):245–273, November 1997.

[Hinton, 2000] Geoffrey E. Hinton. Training products of experts by minimizing contrastive divergence. Technical Report GCNU TR 2000-004, Gatsby Computational Neuroscience Unit, University College London, 2000.
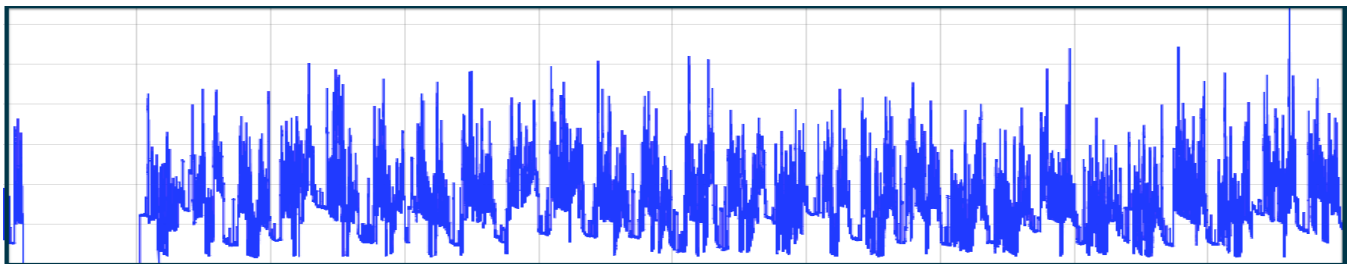
Figure 10: Total Power of the building