ESL-PA-96/07-03-3

Great Energy Predictor Shootout II— A Bayesian Nonlinear Regression with Multiple Hyperparameters

Yohimasa Chonan

Katsuyuki Nishida

Takashi Matsumoto, Dr.Eng.

ABSTRACT

When nonlinearity is present, time series prediction becomes a difficult task. The ASHRAE Energy Predictor Shootout II competition problem is no exception; the difficulty is amplified because analytical equations for describing the dynamics are formidable, if not impossible. The problem belongs to a rather interesting class of problems that can arise in many practical situations. A Bayesian approach is taken in performing nonlinear regression on the ASHRAE Predictor Shootout II time series data.

The Bayesian framework enables one to perform the regression in a hierarchical manner: (i) level 1: estimation of the parameters; (ii) level 2: estimation of hyperparameters; and (iii) level 3: model comparison. The prediction results appear to be reasonable.

INTRODUCTION

When nonlinearity is present, time series prediction becomes a difficult task. The ASHRAE Energy Predictor Shootout II competition problem is no exception, and the difficulty is amplified because analytical equations for describing the dynamics are formidable, if not impossible. The problem belongs to a rather interesting class of problems that can arise in many practical situations.

Let t_m (target) be the values of interest, e.g., electric power consumption, hot water consumption, etc., at time m, and let x_m be an n-dimensional vector consisting of the environmental variables, e.g., temperature, humidity, etc. Let A and B be two disjointed subsets of integers.

Problem: Given the "training" data $D: = \{t_m, x_m\}_{m \in A}$, predict t_m, $m \in B$, when x_m, $m \in B$ is provided.

One possible means of handling this class of problems is to use feed-forward neural networks (perceptrons) with appropriate "training." The purpose of this paper is to predict the ASHRAE Energy Predictor Shootout II time series using a Bayesian approach for training (MacKay 1992a, 1992b; Thodberg 1993; MacKay 1994).

METHOD

Bayesian Nonlinear Regression

To facilitate the reader with a little knowledge about the framework given in the above references, we will briefly describe it.

Given the training data set

$$D: = \{t_m, x_m\}_{m=1}^{N}, (t_m \in R, x_m \in R^n)$$

assume that

$$t_m = f(x_m) + \nu_m \tag{1}$$

where $f(x_m)$ is the true relationship between input and output variables, and ν_m is an independent Gaussian noise with zero mean and variance of $1/\beta$, with β unknown. (Whether this assumption is appropriate depends on the problem.) The goal is to find a function $f(w; x_m)$ that fits $f(x_m)$ by adjusting the parameter vector w. One of the possible choices of $f(w; x_m)$ is a feed-forward neural network, the perceptron. Let H be the "architecture" of the perceptron including the number of layers, the number of hidden units, etc. Then the likelihood is given by

$$\begin{aligned} P(D|w, \beta, H): &= P(\{t_m\}|\{x_m\}, w, \beta, H) \\ &= \prod_{m=1}^{N} P(t_m|x_m, w, \beta, H) \\ &= \left(\frac{\beta}{2\pi}\right)^{\frac{N}{2}} \exp(-\beta E_D) \end{aligned} \tag{2}$$

where

Yoshimasa Chonan is a graduate student and Takashi Matsumoto is a professor in the Department of Electrical, Electronics and Computer Engineering at Waseda University, Tokyo, Japan. Katsuyuki Nishida is with the Tokyo Communication Network, Tokyo, Japan.

THIS PREPRINT IS FOR DISCUSSION PURPOSES ONLY, FOR INCLUSION IN ASHRAE TRANSACTIONS 1996, V. 102, Pt. 2. Not to be reprinted in whole or in part without written permission of the American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc., 1791 Tullie Circle, NE, Atlanta, GA 30329. Opinions, findings, conclusions, or recommendations expressed in this paper are those of the author(s) and do not necessarily reflect the views of ASHRAE. Written questions and comments regarding this paper should be received at ASHRAE no later than July 12, 1996.

$$E_D := \frac{1}{2}\sum_{m=1}^{N}(f(w; x_m) - t_m)^2.$$

Equation 2 gives a measure of how well $f(w; x_m)$ fits the given data set $D: = \{t_m, x_m\}_{m=1}^{N}$. Formally, Equation 2 is the likelihood of (w, β, H) for the data set D. One then needs to specify a prior for w. There are several issues to be considered.

(1) Those weights between inputs and hidden units and those between hidden and output units should have different prior probability distributions. Similarly, the bias parameters may have different prior probability distributions. Furthermore, weights corresponding to different inputs may have different prior probability distributions. This leads to a decomposition $w = (w_1, w_2, \ldots, w_G)$, $w_g \in R^{k_g}$, $g = 1, \ldots, G$, of the weights, each having a different prior probability distribution.

(2) The specific form of priors is another important issue. One possible prior is a Gaussian distribution:

$$\begin{aligned} P(w|a, H) &:= \prod_{g=1}^{G} P(w_g|\alpha_g, H) \\ &= \prod_{g=1}^{G}\left(\frac{\alpha_g}{2\pi}\right)^{\frac{k_g}{2}} \exp\left\{-\sum_{g=1}^{G}\alpha_g E_{w_g}\right\} \end{aligned} \quad (3)$$

where

$$a = (\alpha_1, \ldots, \alpha_G) \in R^G,$$

$$E_{w_g} := \frac{1}{2}\sum_{w \in group\ G} w^2, g = 1, \ldots, G.$$

This says that smaller w-values are more probable than larger ones, which is sometimes valid.

If one assumes Equations 2 and 3, one obtains the posterior distribution of w:

$$\begin{aligned} &P(w|D, a, \beta, H) \\ &:= P(w|\{t_m\}, \{x_m\}, a, \beta, H) \\ &= \frac{\exp(-M)}{\int \exp(-M)dw} \end{aligned} \quad (4)$$

where

$$M = \beta E_D + \sum_{g=1}^{G}\alpha_g E_{w_g}.$$

Thus, the determination of the optimal weights reduces to the minimization of $M(w, a, \beta)$. Note that $M(w, a, \beta)$ is linear in (a, β) so there is no way of determining optimal (a, β) without examining other quantities. The optimal (multiple) hyperparameters are given by maximizing the marginal likelihood:

$$P(D|a, \beta, H) = \int P(D|w, \beta, H)P(w|\alpha, H)dw \quad (5)$$

or the normalization constant in Equation 4. Sometimes, Equation 5 is called "evidence" for hyperparameters.

Model architecture comparison can be done by computing

$$P(H|D) \propto P(D|H)P(H)$$

where

$$P(D|H) = \int P(D|a, \beta, H)P(a, \beta|H)da\, d\beta, \quad (6)$$

which is sometimes called "evidence" for models. Exact analytical formulas for Equations 5 and 6 are not possible due to the nonlinearity of perceptron. Various approximations must be utilized.

Nomenclature

w	= weight vector for perceptron decomposed into G groups: $w = (w_1, \ldots, w_G)$
β	= unknown "noise level" of v_m (see Equation 1)
D	= data set $\{t_m, x_m\}_{m=1}^{N}$
H	= model architecture
a	= $(\alpha_1, \ldots, \alpha_G)$ = unknown multiple hyperparameters that determine the sharpness of the prior probability distributions of $w_1, \ldots, w_G$
$P(D|w, \beta, H)$	= likelihood for D
$P(w|a, H)$	= prior probability distribution of w
$P(w|D, a, \beta, H)$	= posterior probability distribution of w
$P(D|a, \beta, H)$	= marginal likelihood for D, marginalized with respect to w
$P(D|H)$	= marginal likelihood for D, marginalized with respect to (a, β)
$P(a, \beta|H)$	= prior probability distribution of (a, β)
$P(H)$	= prior probability distribution of H

IMPLEMENTATION

Model

Due to time limitations, we used the simplest model (Equation 1) although much more general models are possible. For instance, it would be interesting to see how the dynamic system model works:

$$t_{m+1} = f(t_m, t_{m-1}, \ldots, t_{m-\tau_1}; x_t, x_{t-1}, \ldots, x_{t-\tau_2}) + v_m.$$

This project is presently in progress. Preliminary results are encouraging. Details will appear elsewhere.

Preprocessing

The preprocessing method used here is basically the same as the one used in Iijima et al. (1994), which will be briefly described below.

Trend Removal First, data were divided into two parts: workdays and holidays. Let W (respectively M, L, C, H) denote the WBE (respectively, MCC, LTEQ, CWE, HWE) and let W_w (W_H) be the WBE of workdays (respectively, holidays). Then the "trends" were removed from WBE data in C.TRN, which has a 24-hour component via

$$\hat{W}_W^{m,d,t} = W_W^{m,d,t} - h_{W_w}^{m,t}$$

$$\hat{W}_H^{m,d,t} = W_H^{m,d,t} - h_{W_H}^{m,t}$$

where

$$h_{W_w}^{m,t} = \sum_{d \in \#W^m} W_W^{m,d,t} / \#W^m$$

$$h_{W_H}^{m,t} = \sum_{d \in \#H^m} W_H^{m,d,t} / \#H^m.$$

Similarly, a 24-hour periodic component was removed from WBE and LTEQ in D.TRN via

$$\hat{L}_W^{m,d,t} = L_W^{m,d,t} - h_{L_w}^{m,t}$$

$$\hat{L}_H^{m,d,t} = L_H^{m,d,t} - h_{L_H}^{m,t}$$

where

$$h_{L_w}^{m,t} = \sum_{d \in \#W^m} L_W^{m,d,t} / \#W^m$$

$$h_{L_H}^{m,t} = \sum_{d \in \#H^m} L_H^{m,d,t} / \#H^m$$

$W_W^{m,d,t}$ means the WBE of a workday at time t, day d, and month m. $\#W^m$ is the number of workdays of month m, and $\#H^m$ is the number of holidays of month m. No trends were discernible in the CWE and HWE data.

Outliers Outliers in the data set often deteriorate "training" in a significant manner. Removal of outliers is an important step. The following data were removed by inspection from the training set:

Data Set C:

WBE:

'90 Jan. 2-5, 8-12

'90 March 12-16

'90 Aug. 14-17, 20-24

'90 Sept. 17 13:00, 20:00

CWE:

'90 Jan. 19 22:00-20 7:00

'90 Feb. 27 10:00-17:00

'90 June 21 14:00-16:00

'90 Sept. 17 13:00, 20:00

HWE:

'90 June 20 0:00-1:00

'90 Sept. 17 13:00, 20:00

Data Set D:

WBE:

'91 March 17-24, 29

LTEQ:

'90 Dec. 22-31

'91 Jan. 9, 10, 24-27

'91 Feb. 7, 9

'91 March 4-5, 22-24, 30-31

'91 Apr. 17-19, 22, 25-26, 29-30

'91 May 1-2, 4

'91 July 12

Automatic outlier removal is a rather interesting subject for future works.

Normalization Let x_m be one of the environmental variables, e.g., temperature. Let x_{max} and x_{min} be the maximum and minimum values of x_m among the data set. Normalization

$$\frac{x_m - x_{min}}{x_{max} - x_{min}}$$

was performed before the variable was fed into perceptrons so that the input to perceptrons varies within the unit interval.

Training For data set C, the following were used as the x_m-variables: temperature, solar flux, wind speed, and time where the time variables consist of sin (day of the year), cos (day of the year), sin (hour of the day), and cos (hour of the day). Note that each hidden unit of a perceptron receives a linear combination of the input variables so that the "phase shift" can be realized through

$$w_1 \sin(\text{day of the year}) + w_2 \cos(\text{day of the year}) = \sqrt{w_1^2 + W_2^2} \sin\left(\text{day of the year} + \tan^{-1}\frac{w_1}{w_2}\right).$$

Similarly, the phase shift of sin (hour of the day) and cos (hour of the day) can be realized. The relative humidity (RH) data provided were not used since too many data points were missing. For data set D, the same variables were used for training except for the fact that the RH data were also used. The number of hidden units were chosen in terms of the marginal likelihood $P(D|H)$ (see Equation 6) and are shown in Tables 1 and 2.

Note that six perceptrons were provided for data set C while eight perceptrons were trained for data set D. Here "training" means optimization with respect to w and (α, β) so as to optimize Equations 4 and 5. The conjugate gradient method was used. For a fixed number

TABLE 1 Data Set C

Network	W_W	W_H	C_W	C_H	H_w	H_H
# of hidden units	8	8	11	6	10	10

TABLE 2 Data Set D

Network	W_W	W_H	L_W	L_H	C_W	C_H	H_w	H_H
# of hidden units	7	12	8	6	8	5	7	5

of hidden units, training typically took from five to seven hours on a Sunday (SPARC CY7C601 performing 28.5MIPS).

PREDICTIONS

Since the trends were removed before the data were fed to the neural networks, the prediction phase naturally requires detrending. Namely, the trends are added to the network output after training for predictions.

WBE:

C.TRN, D.TRN: The trends removed in the preprocessing stage were added to the network prediction:

$$W^{m,d,t}_{W,pred} = \hat{W}^{m,d,t}_{W,pred} + h^{m,t}_{W_W}$$

$$W^{m,d,t}_{H,pred} = \hat{W}^{m,d,t}_{H,pred} + h^{m,t}_{W_H}$$

MCC:

C.TRN: The mean over $t = 1, \ldots, N$ was used as the prediction:

$$M_{pred} = \frac{\sum_{t=1}^{N} M^{m,d,t}}{N}.$$

D.TRN: The mean over 24 hours was used as our prediction:

$$M^{m,t}_{i,pred} = \frac{\sum_{d} M^{m,d,t}_{i}}{N_{i,m}}.$$

LTEQ:

C.TRN: The prediction was

$$L^{m,d,t}_{pred} = W^{m,d,t}_{pred} - M_{pred} - A$$

where

$$A = W^{m,d,t}_{W} - \hat{L}^{m,d,t}_{W} - M^{m,d,t}_{W} = 117.79.$$

D.TRN: Similar to WBE prediction:

$$L^{m,d,t}_{W,pred} = \hat{L}^{m,d,t}_{W,pred} + h^{m,t}_{L_W}$$

$$L^{m,d,t}_{H,pred} = \hat{L}^{m,d,t}_{H,pred} + h^{m,t}_{L_{W_H}}$$

CWE, HWE:

C.TRN, D.TRN: The perceptron outputs were quantized and submitted as our prediction.

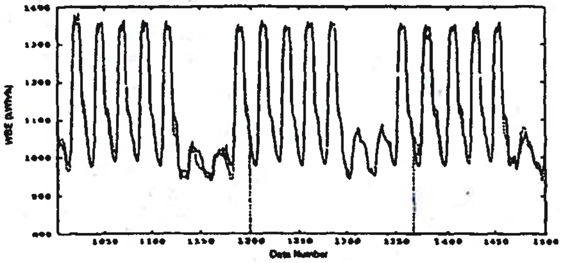

Figure 1 A part of the "predicted" C.TRN WBE (solid line) and the actual (dashed) line.

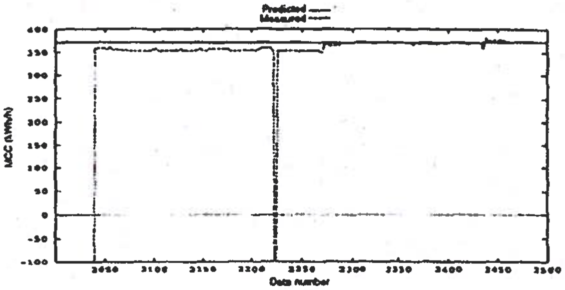

Figure 2 A part of the "predicted" C.TRN MCC and the actual data.

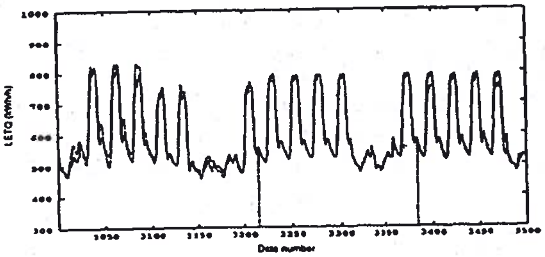

Figure 3 A part of the "predicted" C.TRN LTEQ and the actual data.

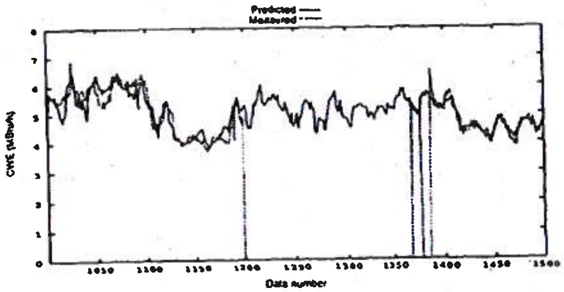

Figure 4 A part of the "predicted" C.TRN CWE and the actual data.

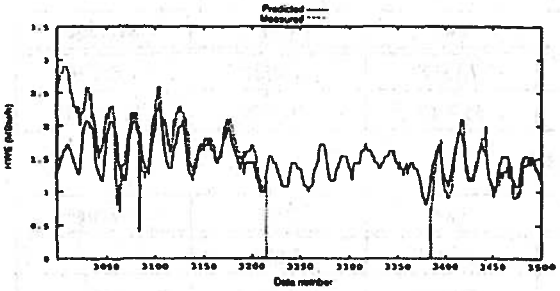

Figure 5 *A part of the "predicted" C.TRN HWE and the actual data.*

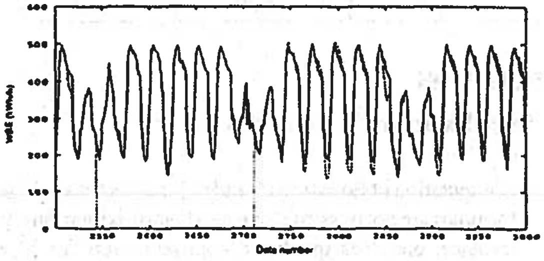

Figure 6 *A part of the "predicted" D.TRN WBE and the actual data.*

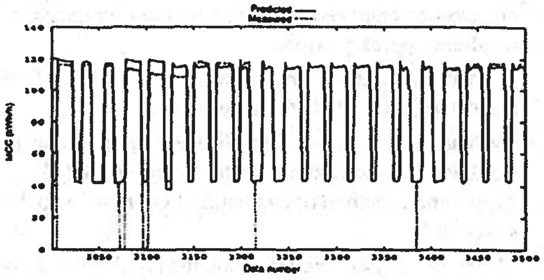

Figure 7 *A part of the "predicted" D.TRN MCC and the actual data.*

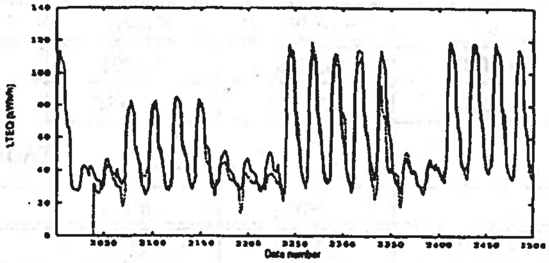

Figure 8 *A part of the "predicted" D.TRN LTEQ and the actual data.*

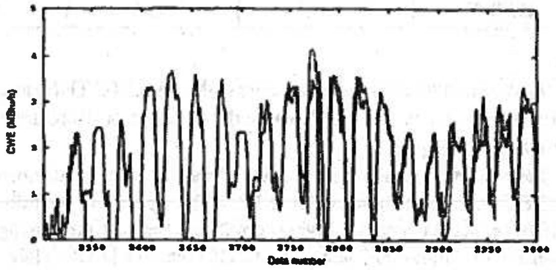

Figure 9 *A part of the "predicted" D.TRN CWE and the actual data.*

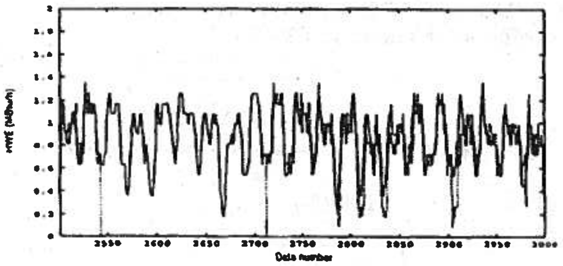

Figure 10 *A part of the "predicted" D.TRN HWE and the actual data.*

TABLE 3 CV-RMSE

	wbe	mcc	lteq	cwe	hwe	average
C.TRN	3.1205	3.2803	4.456	7.1312	21.2758	7.85276
D.TRN	17.0462	17.5371	21.3187	55.9467	46.517	31.58008

TABLE 4 MBE

	wbe	mcc	lteq	cwe	hwe	average
C.TRN	0.27225	0.4817	0.5106	−0.8917	−3.0988	—
D.TRN	2.7898	4.4054	−3.4806	−3.6721	−11.1313	—

TABLE 5 Average Difference for the "Removed Data" vs. the "Training" Data

	wbe	mcc	lteq	cwe	hwe	average
C.TRN	0.75	0.11	1.17	4.65	6.65	2.66
D.TRN	1.96	3.29	5.96	21.45	15.30	9.59

Typical results are given in Figures 1 through 5 (C.TRN) and Figures 6 through 10 (D.TRN) where the "predictions" are done within the training set.

The prediction results (Tables 3 and 4) show that our predictions for C.TRN are better than those for D.TRN. This appears to be attributable to the fact that there are higher correlations between the training data set and the "removed" data set for C.TRN than for D.TRN (Table 5). The fact that more data are provided in C.TRN than in D.TRN may have also contributed to the results.

Prediction errors for CWE and HWE are large compared with other quantities (Tables 3 and 4). The withheld data set of CWE and HWE appears to be very different from the training data of CWE and HWE. These errors significantly deteriorated our prediction accuracy. Our CV-RMSE was 19.71% while MBE was -1.3815%.

Coefficient of Validation, CV (%)

$$CV-RMSE = \frac{\sqrt{\frac{\sum_{i=1}^{n} (y_{pred,i} - y_{data,i})^2}{n-p}}}{\bar{y}_{data}} \times 100$$

Mean Bias Error, MBE (%)

$$MBE = \frac{\sqrt{\frac{\sum_{i=1}^{n} (y_{pred,i} - y_{data,i})}{n-p}}}{\bar{y}_{data}} \times 100$$

where

y_{data} = data value of the dependent variable corresponding to a particular set of the independent variables,

$y_{pred,i}$ = dependent variable value for the same set of independent variables above,

$\bar{y}_{data}$ = the mean value of the dependent variable of the data set, n is the number of data points in the data set, and p is the total number of regression parameters in the model, which was arbitrarily assigned as 1 for all models.

CONCLUSION

Several issues need to be studied further.

- Computation of Equations 5 and 6. Since exact analytical formulas are not possible, one needs approximations. For instance, one uses quadratic approximation for $M(w)$. Even though the eigenvalues of the Hessian of $M(w)$ at its minimum is non-negative theoretically, one often encounters negative eigenvalues. In the current algorithm, we are simply ignoring those negative eigenvalues. Care needs to be taken since smaller (positive) eigenvalues contribute larger values for Equation 5 in quadratic approximations.
- A good optimization algorithm of for Equation 5 with respect to (α,β) needs to be developed.
- Evaluation 6 is rather complicated, even if one uses quadratic approximation with respect to (α,β). An approximate method of evaluating Equation 6 should be developed.
- More studies need to be done on the priors of W as well as (α,β).

One of the main purposes of our participation in the ASHRAE competition was to see how our algorithm works on real-world data without being expert on the subject matter. The participation was not easy, however. Most of the time and energy were spent on understanding what to do and by the time we understood that, not much time was left. Despite this, our prediction results appear to be encouraging given the time limitation and the very little knowledge of the subject matter. The prediction results will be significantly improved if we study more about specific details of the problem so that the algorithm can be fine tuned. Significant improvements will also be obtained if we study those methods developed by experts on the particular subject matter and incorporate them into our algorithm. In any practical problem, a great amount of data analysis and fine tuning are necessary for an algorithm to be of practical use. We intend to apply our algorithm to other practical time series prediction problems.

REFERENCES

MacKay, D.J.C. (1992a). Bayesian interpolation. *Neural Computation* 4 3: 415-447.

MacKay, D.J.C. (1992b). A practical Bayesian framework for backprop networks. *Neural Computation* 4 3: 448-472.

Thodberg, H.H. (1993). Ace of Bayes: Application of neural networks with pruning. *Technical report No 1132E.* Danish Meat Research Institute.

MacKay, D.J.C. (1994). Bayesian nonlinear modeling for the prediction competition. *ASHRAE Transactions* 100 (2): 1053-1062.

Iijima, M., K. Takagi, R. Takeuchi, and T. Matsumoto. (1994). A piecewise-linear regression on the ASHRAE time-series Data. *ASHRAE Transactions* 100(2): 1088-1095.