STATISTICAL ANALYSIS OF NEURAL NETWORKS AS APPLIED TO BUILDING ENERGY PREDICTION

Robert H. Dodier*
Department of Civil Engineering
University of Colorado at Boulder
Boulder, Colorado

Gregor P. Henze†
Department of Civil Engineering
University of Colorado at Boulder
Boulder, Colorado

Abstract

It has been shown that a neural network with sufficient hidden units can approximate any continuous function defined on a closed and bounded set. This has inspired the use of neural networks as general nonlinear regression models. As with other nonlinear regression models, tools of conventional statistical analysis can be applied to neural networks to yield a test for the relevance or irrelevance of a free parameter. The test, a version of Wald's test, can be extended to yield a test for the relevance or irrelevance of an input variable. This test was applied to the building energy use data of the Energy Prediction Shootout II contest. Input variables were selected by initially constructing a neural network model which had many inputs, then cutting out the inputs which were deemed irrelevant on the basis of the Wald test. Time-lagged values were included for some input variables, with the time lag chosen by inspecting the autocovariance function of the candidate variable. The results of the contest entry are summarized, and the benefits of applying Wald's test to this problem are assessed.

Introduction

This paper describes the authors' entry to the Energy Predictor Shootout II. The purpose of the contest was to encourage research in methods of estimating the reductions in energy use in large commercial buildings due to equipment retrofits. The immediate goal of the contest entrants, then, was to construct a model of energy use of an unmodified (pre-retrofit) building. This model was calibrated using data collected in the building before the retrofit. In these pre-retrofit data sets about one week's data per month were withheld from the contestants. The performance of each contestant's model was ranked by the accuracy of prediction of energy use on the withheld pre-retrofit data, which was known only to the contest organizers. Finally, the reduction in energy use was estimated as the difference between the actual post-retrofit energy use in the building and the energy use predicted for the pre-retrofit building under the post-retrofit environmental conditions. The contest opened in June, 1994, and closed November 1, 1994.

There were two buildings under consideration, an Engineering Center (EC) in central Texas, and a Business Building in northern Texas. For each building, there were five energy use variables for which contestants had to make predictions. These variables were whole building electricity (WBE), motor control center electricity (MCC), lights and equipment electricity (LTEQ), chilled water energy (CHW), and hot water energy (HW). Contestants were supplied with pre-retrofit data from the EC covering most of the year 1990, and pre-retrofit data from the Business Building covered a little more than the first six months of 1991. Contestants were asked to submit predictions on certain parts of the pre-retrofit data that were withheld by the organizers, to evaluate the accuracy of the contestants' models. Accuracy was evaluated by the organizers according to the coefficient of variation (CV); a second criterion, mean bias error (MBE), was defined but not used. (As the purpose of the contest was to develop methods for accurate estimation of *total* energy use, it may be that ranking models by MBE would have been

*E-Mail: dodier@colorado.edu
†E-Mail: henze@colorado.edu

more useful.)

$$CV = \frac{\sqrt{1/N \sum_t (y_t - \hat{y}_t)^2}}{1/N \sum_t y_t} \quad (1)$$

$$MBE = \frac{1/N \sum_t (y_t - \hat{y}_t)}{1/N \sum_t y_t} \quad (2)$$

Here y_t is a target value, $\hat{y}_t$ is a prediction, and N is the number of items in the test set. Contestants were also asked to submit predictions for the EC and the Business Building for the entire calendar year 1992, which was after the HVAC equipment retrofits.

A summary of the results of the contest is given by Haberl *et al.* (1996). There were four complete entries. A group composed of the present authors made the most accurate predictions for the withheld data. The energy savings estimated by all groups for the EC after equipment retrofits were remarkably similar, but the energy savings estimated for the Business Building differed wildly, and increased energy use was predicted for some end uses.

The First Energy Prediction Contest

The first energy prediction contest, the Great Energy Predictor Shootout, was held in 1993. As in the second contest, contestants were asked to make hour by hour predictions of energy use in a large building. Contestants were also asked to estimate beam insolation given multipyranometer readings. Several of the contestants used neural networks, with good results. It is instructive to consider the most successful strategies used in the first contest. The contest winner, D. Mackay, used neural networks which featured 'automatic relevance determination' (ARD), a method of automatically detecting relevant input variables (Mackay, 1994) based on Bayesian estimation. While similar in spirit to the Wald test used for pruning in this paper, ARD is 'soft' in the sense that less-relevant inputs are given less influence, instead of the hard, all-or-nothing pruning. Final predictions were made by a 'committee' of networks instead of by a single network; this again represents a Bayesian approach. The second–place contestants, M. Ohlsson et al., employed a statistical test for nonlinear correlation, which they termed the 'delta-test,' to select input variables which were relevant to predicting the output (Ohlsson et al., 1994). This team also used the early stopping procedure (Weigend *et al.*, 1990) to prevent overfitting on the building energy data. As yet another approach to the application of neural networks to building energy prediction, the third–place contestants, B. Feuston and J. Thurtell, assembled large numbers of input variables and then used principal components analysis to reduce the number of input variables (Feuston and Thurtell, 1994). Hence the neural network inputs consisted not of environmental and calendar variables directly, but linear combinations of environmental and calendar variables. A large number of networks were trained by early stopping and the average prediction of a committee was reported, as Mackay also did.

The three contestants mentioned here all identified one essential problem in common, namely the problem of determining which inputs are relevant to prediction. Each had a different way of solving this problem, be it ARD, the delta-test, or principal components. In agreement with these investigators, the present authors' team also identified input relevance as a serious problem in neural networks applications. Toward this end, the Wald test was used to evaluate the relevance of various inputs; the results are described in this paper.

Background on Neural Networks as Regression Models

The present authors' group chose to use neural networks, a class of general, flexible nonlinear regression models, to find the relation between the input variables and the output variables. It has been shown by Cybenko (1989) and Hornik *et al.* (1989) that a neural network with sufficient hidden units can approximate any continuous function defined on a closed and bounded set. Neural networks place few limitations on the input-output relation, and require much data to reliably find the relation. This is in contrast to so-called 'strong' models which are based on physical principles. Neural nets are notorious for finding spurious relations in noisy data. It was assumed that the several thousand data samples supplied by the contest organizers were sufficient data to reliably find the input-output relation. The regression theory and prediction application which are outlined here are described at greater length by Dodier (1995).

A neural network is a function constructed from compositions of weighted sums of bounded monotone functions. A neural network is conventionally visualized as a directed graph. A cycle in the graph is called a *recurrence*. In the work presented in this paper, only nonrecurrent networks are used[1]. As an example of a

[1] Past experience with recurrent networks (Kreider *et al.*, 1995) showed that nonrecurrent networks performed better than recurrent networks with the same inputs. It may be that the greater flexibility of recurrent networks makes them more sensitive to noise in the data. Also, as outputs from one time step become

typical nonrecurrent network, consider

$$f(\mathbf{x},\mathbf{w}) = g_{output}\left(c + \sum_{k=1}^{3} v_k g_{hidden}\left(b_l + \sum_{l=1}^{5} u_{kl}x_l\right)\right). \quad (3)$$

The functions g are the so-called activation functions, which in the work at hand are the identity mapping for the output unit ($g_{output}(x) = x$) and the hyperbolic tangent for hidden units ($g_{hidden}(x) = \tanh x$). Concerning the choice of activation for the output, a squashing activation ($\tanh x$ or $1/(1+\exp(-x))$) is appropriate for classification tasks in which the target is constrained to be ± 1 or $[0,1]$, while a linear activation is appropriate for regression tasks in which targets take on continuous real values without fixed limits. The weights u_{kl} and v_k and the biases c and b_l are free parameters, which are conventionally grouped together into a single weight vector, $\mathbf{w} = (b_1,\ldots,b_5,c,u_{11},u_{12},\ldots,u_{35},v_1,v_2,v_3)$.

Maximum Likelihood Estimation

Given a model of an input-output relation such as a Fourier series or neural network and some amount of data, there is the problem of adjusting the model's free parameters so that it 'fits' the data. This process of picking best-fitting parameters is called *estimation*. A very broadly applicable method of estimation called *maximum likelihood* (ML) estimation will now be introduced, based on the following principle. Given a sample $\mathbf{x}$, we assume it is taken from a known distribution with some parameters θ, although we do not know the correct values of the parameters. Let us write the probability of the data under the distribution as $p(\mathbf{x}|\theta)$. When $\mathbf{x}$ has been observed, $p(\mathbf{x}|\theta)$ is a function of θ called the *likelihood function*. Now the principle is easily formulated.

> **(Principle of Maximum Likelihood)** Given data $\mathbf{x}$, choose free parameters θ to maximize the probability of the data under the distribution. That is, choose θ to maximize the likelihood function $p(\mathbf{x}|\theta)$.

Now let us apply the ML principle to adjusting the free parameters of a regression model $f(\mathbf{x},\mathbf{w})$. The conventional noise model is an additive independent Gaussian noise model,

$$y_k = f(\mathbf{x}_k,\mathbf{w}) + \epsilon_k \quad (4)$$

where the noise term ϵ_k is Gaussian distributed with mean zero and some variance σ^2, and ϵ_j and ϵ_k are independent when $j \neq k$. For each input-output datum $(\mathbf{x}_k, y_k)$ we have $y_k - f(\mathbf{x}_k,\mathbf{w}) = \epsilon_k$, so the likelihood of each datum is

$$p(\mathbf{x}_k, y_k|\mathbf{w},\sigma) = \frac{1}{\sigma\sqrt{2\pi}}\exp(-\frac{1}{2\sigma^2}(y_k - f(\mathbf{x}_k,\mathbf{w}))^2) \quad (5)$$

and the joint negative log likelihood, again assuming independence, is

$$\begin{aligned} -\log L(\mathbf{w},\sigma) &= -\sum_{k=1}^{n}\log p(\mathbf{x}_k, y_k|\mathbf{w},\sigma) \quad (6)\\ &= \frac{1}{2\sigma^2}\sum_{k=1}^{n}(y_k - f(\mathbf{x},\mathbf{w}))^2 \\ &\quad + n\log\sigma + \frac{n}{2}\log 2\pi \\ &= \frac{1}{2\sigma^2}SSE + n\log\sigma + \frac{n}{2}\log 2\pi. \quad (7)\end{aligned}$$

In the last equation the sum of squared errors SSE was introduced for convenience. It can be shown (Seber and Wild, 1989, Sec. 2.2) that the ML estimate for σ is

$$\hat{\sigma}^2 = \frac{1}{n}SSE, \quad (8)$$

and the ML estimate for the model parameters $\hat{\mathbf{w}}$ satisfies

$$SSE(\hat{\mathbf{w}}) \leq SSE(\mathbf{w}) \text{ for all } \mathbf{w}. \quad (9)$$

This, then, is the theoretical justification for the conventional least-squares parameter adjustment. As the name suggests, the least-squares procedure is to minimize the sum of squared errors, SSE. Under the conditions specified here, namely independent additive Gaussian noise and a model of known form but unknown parameter values, the least-squares parameters are just the ML estimates. The usual backpropagation algorithm for training neural networks is an application of least-squares, hence under certain conditions the backpropagation algorithm yields the ML parameters.

Unfortunately, rather strong assumptions had to be made to derive the ML estimates, Eqs. 8 and 9. When the stated conditions break down, least-squares may give misleading results and some different training algorithm is needed.

Input Selection and Weight Pruning

Because of the noise ϵ in the observations y, two different realizations of a set of input-output pairs $(\mathbf{x}, y)$ may be different even when the inputs $\mathbf{x}$ are the same. Each time backpropagation is applied to a data set, the values found for the weights will be slightly different. Suppose

inputs at the next, errors are propagated through time.

now that we fix the set of inputs and consider all possible realizations of training sets of a given size n. Each data set $(\mathbf{x}, y)$ yields different least-squares weights, but the weights tend to cluster together around the true values. When there are many data, the cloud of post-training weight vectors can be described by a Gaussian distribution. The mean of the post-training weights distribution is the true weight vector, and the dispersion around the true vector is a function of the Hessian of the error. The arguments leading up to the Gaussian post-training distribution are based on an asymptotic linearization. ("Asymptotic" because the number of data is large.) This line of reasoning is not reproduced here; see Chap. 12 of Seber and Wild (1989) for the complete argument.

The main result of the asymptotic theory is the following. As the number of data $n \to \infty$,

$$\mathbf{w} \sim N(\mathbf{w}^*, \sigma^2(\mathbf{F}.'\mathbf{F}.)^{-1}) \tag{10}$$

where $\sigma^2(\mathbf{F}.'\mathbf{F}.)^{-1}$ is the asymptotic covariance matrix. Here $\mathbf{F}.$ is the matrix formed by computing the gradient of the output with respect to the weights for each input $\mathbf{x}_i$ and evaluated at the exact weights,

$$\mathbf{F}. = \left(\frac{\partial f}{\partial w_j}(\mathbf{x}_i, \mathbf{w}^*) \right) \tag{11}$$

This matrix $\mathbf{F}.$ can be estimated as $\hat{\mathbf{F}}.$ in which the required derivatives are evaluated at the least-squares weights $\hat{\mathbf{w}}$.

Suppose that the process generating the data we have is actually a neural network, but we have chosen a network that is bigger than the correct one. That is, the data generating process can be described with fewer weights than we are using. The "irrelevant input" and "irrelevant hidden unit" hypotheses are special cases of this scenario. If an input is irrelevant, then all the weights leading out of that input unit should be zero. If a hidden unit is irrelevant, then all the weights leading out of that hidden unit should be zero. The process of identifying irrelevant weights and units and removing them from the network is known as 'pruning' in the neural network parlance. We should identify the irrelevant units if we can and remove them. Unnecessary weights carry no information and contribute only noise to the output.

If we have many data, we can apply the Gaussian approximation to the post-training weight distribution. (Recall that the Gaussian approximation was derived under the assumption of unlimited data; it is difficult to say, then, whether 50 data, or 50 thousand, or 50 million are 'approximately' infinite.) Let $\mathbf{C} = (\hat{\mathbf{F}}.'\hat{\mathbf{F}}.)^{-1}$ denote the asymptotic covariance matrix. Suppose we wish to prune a set of weights $\{w_{m_k} : k = 1, \ldots, q\}$. For example, if the set is $\{w_7, w_{12}, w_{20}\}$, the sequence of indices is $m_1 = 7, m_2 = 12, m_3 = 20$. Let $\mathbf{B} = (C_{m_i, m_j})$, so $\mathbf{B}$ is $q \times q$. The Wald test[2] statistic is then defined as

$$W = \frac{1}{SSE/n} \sum_{i=1}^{q} \sum_{j=1}^{q} \hat{w}_{m_i} \hat{w}_{m_j} (\mathbf{B}^{-1})_{ij}. \tag{12}$$

This is the equation implemented by the software developed for the Energy Predictor Shootout II.

With the test statistic W in hand, the input selection procedure can be described in detail. Fig. 1 depicts the test. Using the terminology of statistical inference, the random variable of interest is the least-squares weight vector, $\hat{\mathbf{w}}$. A function of this is our test statistic W, given by Eqs. 12. As our null hypothesis we have the true weights leading out of a specified input unit are zero, that is, that a certain set of true weights are all zero, $w^*_{m_1} = \ldots = w^*_{m_q} = 0$. There are q weights under consideration; for an input unit connected to hidden units, q is the number of hidden units. Our alternate hypothesis is that at least one of these true weights are nonzero. We construct our test by specifying an appropriate probability of a type I error, α. This determines a critical value for the test statistic, W_{crit}, which satisfies

$$\int_{W_{crit}}^{\infty} p_{\chi^2}(u, q) du = \alpha. \tag{13}$$

Here $p_{\chi^2}(u, q)$ denotes the density function of the $\chi^2(q)$ distribution; note that the mean value of W is just q, the number of weights under consideration. The null hypothesis is accepted if $W < W_{crit}$. That is, if the test statistic W is small enough, we say the input unit under consideration is irrelevant, and prune it out of the network: the weights leading out of the input unit are frozen at zero. It is of interest to assess the probability of a type II error, but this problem is more difficult and beyond the scope of this paper. See Seber and Wild (1989), Sec. 12.4 for more details.

(It is not clear, however, what is the most appropriate probability of a type I error. In the tests described in this paper, $\alpha = 0.001$ was used. Raftery (1994) suggests that the larger the data sample, the smaller α should be. Whether we should be radical and set α close to 0, thus encouraging pruning, or whether we should be conservative and keep marginally relevant variables, awaits further investigation.)

[2]The test is named after its originator, A. Wald. The development presented here follows Seber and Wild (1989), Sec. 5.9; the original reference is Wald (1943).

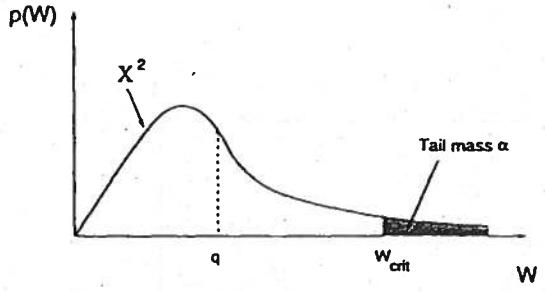

Figure 1: A χ^2-distribution with the mean and critical values of W indicated. The tail mass shown here is $\alpha = 0.001$.

Approach

Making Predictions

The authors' entry used one network per variable to be predicted, so there were 10 networks altogether, there being CHW, HW, LTEQ, MCC, and WBE variables to predict and two buildings, the EC and the Business Building. The networks all had two hidden layers of 25 units each; this gave each net about 1000 free parameters. The number of inputs varied from net to net; see the discussion under Sec. below. The hidden units had tanh activations and the single output unit had a linear activation.

To estimate model parameters the authors used code for backpropagation which one of the authors (Dodier) wrote. An extension of this code was used for computing Wald's statistic, which was used for statistical tests as described in Sec. above. A very useful feature of this code is the use of C++ objects for matrices and networks, which allows a flexible and comprehensible programming environment. Nets were trained on several computers of various types. In all, approximately 40 hours of CPU time on workstation-class machines were used in training and analyzing networks.

Selection of Input Variables

On the basis of visual examinations of the target data, it was decided that there are strong daily, weekly, and yearly cycles. Also, there is reason to believe that the occupancy of the buildings strongly influences energy usage, hence whether school is or is not in session is important also. In addition to the day of the year and hour of the day which were supplied by the contest organizers, day-type flags were generated, as described in Sec. below. Environmental variables (ambient dry-bulb temperature, relative humidity, insolation, and windspeed) were also supplied by the organizers, so there were both calendar and environmental variables available as inputs. Input variables were chosen by Wald's test. Networks with one hidden layer of eight hidden units were trained with all time, occupancy, and instantaneous environmental variables[3], and the effect of removing an input was evaluated by computing the Wald test statistic W. If W is large enough to be statistically significant (*i.e.* larger than W_{crit}), the input is deemed relevant to computing the output and it is kept. If W is not significant, the input variable is removed from the set of inputs. The time and occupancy variables were found relevant for all output variables, but for some outputs some of the environmental variables were not relevant.

The present authors' entry did not have a principled way of dealing with missing inputs; a method called 'mean imputation' was employed, in which the average value of a variable is substituted when the variable is unknown. However, this substitution was only made when computing outputs; data with one or more missing variables were ignored completely in training. Unfortunately, a large part of the pre-retrofit data had some environmental variables missing, so that the actual number of data used for training was substantially less than the number of data (complete and incomplete) which were provided. As it was found that very many of the values for relative humidity (RH) are missing in the training sets (about 40% missing in the EC training data), RH was omitted from all networks. Since Wald's test showed that RH had only low to moderate relevance to any output, it was hoped the gain in reliability of predictions would outweigh the loss of a predictive variable. Missing inputs remain a problem for further investigation.

Choice of Time-Lag Variables

Time-lag environmental variables were also used as inputs. We chose the lag time between successive values as the first zero of the autocovariance function. This method is suggested by Abarbanel (1992). These delay times were chosen as 11 hours for temperature and 8 hours for insolation. It was also observed that the autocovariance of windspeed decreases very rapidly, so given its marginal relevance to any output it seems the relevance of time-lagged values would have to be very low, and time-lagged values of windspeed were not in-

[3]Time-lag variables were included if and only if the instantaneous variable was included. This procedure was based on the notion that time-lag variables would be relevant if and only if the corresponding instantaneous variable were relevant. It is not clear that this is correct.

cluded in any model. Since including more time-lag values increases the size of the net and decreases the amount of data available for training (by time-shifting missing values), only two time-lag values for temperature (11 and 22 hours) and insolation (8 and 16 hours) were included, as well as current-hour values of temperature, insolation, and windspeed.

The motivation behind the choice of time lags is as follows. Past values of environmental variables are relevant to prediction of current energy variables due to the effect of building mass, so including past values of inputs should improve the accuracy of predictions. However, it is clear that there are limits to the useful time lag. If the time lag is very long (a month, say) the time-lagged input will be almost completely uncorrelated with the current energy variable. If, on the other hand, the time lag is very short (a minute, say) the time-lagged input will be almost completely redundant with the current input variable, providing no new information. To choose an appropriate intermediate time lag, we inspected the autocovariance function,

$$C(\tau) = \frac{1}{N}\sum_t (x_t - \bar{x})(x_{t-\tau} - \bar{x}), \qquad (14)$$

of each input variable at multiples of one hour, the shortest possible time lag given the data supplied by the contest organizers. The first zero of the autocovariance represents a time at which the lagged input is uncorrelated with the current value, so it is the shortest time lag which brings in the most new information. Figure 2 shows the autocovariance functions of ambient temperature, insolation, and wind at time lags up to 24 hours. Both temperature and insolation show a strong diurnal cycle as shown by the autocovariance, which is negative around the 12 hour time lag and positive around the 24 hour lag. Wind shows the same cycle, but to a lesser degree. From this graph, it can be seen that the autocovariance of temperature has its first zero near 11 hours, while insolation has its first zero near 6 hours. (Another insolation data sample, which had a zero near 8 hours, was used to choose time lags used to construct networks for the competition.) A longer plot of autocovariance of temperature is shown in Figure 3. The daily cycle is quite evident, as is the gradual decay of the magnitude of the peaks; this is to be expected of a driven, chaotic system.

It should be mentioned that the autocovariance detects only linear dependency between different times. For systems with nonlinear dependencies, it is more correct to study the mutual information,

$$I(\tau) = \int p(x_t, x_{t-\tau}) \log \frac{p(x_t|x_{t-\tau})}{p(x_t)} dx_t dx_{t-\tau}, \qquad (15)$$

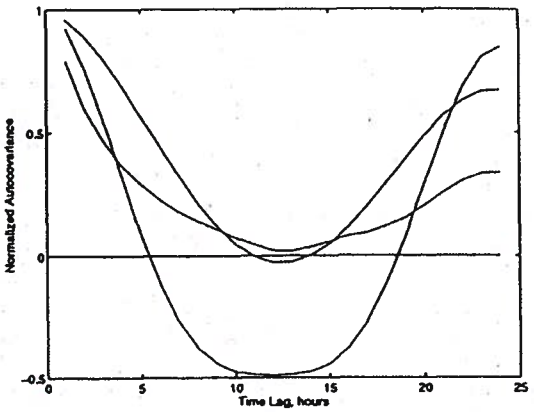

Figure 2: Short-term normalized autocovariance functions, at lags from 1 to 24 hours. At left, from the top downwards, there are temperature, insolation, and wind.

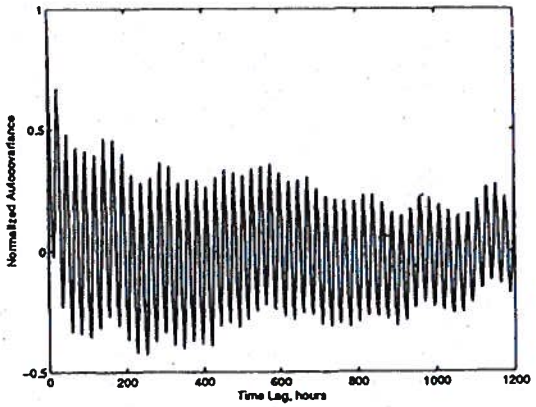

Figure 3: Long-term normalized autocovariance function of ambient temperature, at lags from 1 hour to 50 days.

but this is difficult to compute reliably due to the need to estimate the conditional and marginal densities $p(x_t|x_{t-\tau})$ and $p(x_t)$.

Final Choice of Input Variables

Tables 1 and 2 show the results of using the Wald test for input variable selection as described above. For these tests, networks with 8 hidden units were constructed, using the 10 available input variables without time lags. Input variables written in SMALL CAPS were deemed irrelevant to prediction on the basis of these tests and were removed from the set of inputs. The other input variables, written in lowercase, were kept. Of these, those written in *italics* have a test statistic $W < 100$, and so may be considered less relevant than those written in roman, which have the largest W values. The critical value W_{crit}, given eight degrees of freedom (eight weights removed if input is omitted), is 26.1 for $\alpha = 0.001$. For CHW and HW targets, the original Wald test results were lost and had to be recomputed for this paper. Unfortunately, the computed relevances of some variables differed from the original computation, so that some variables which would be judged irrelevant on the basis of Tables 1 and 2 were kept in the competition networks. This points up a shortcoming of the use of the Wald test statistic, namely that random variations may result in inconsistent relevance assessments.

Data Pre-processing Steps

We converted the day of year N into a 'clock' representation (M. Mozer, personal communication), in which

$$\sin \frac{2\pi N}{365}, \tag{16}$$

$$\cos \frac{2\pi N}{365} \tag{17}$$

together represent the day. This makes December 31 very similar to January 1. Likewise we used a clock representation for the hour-of-day.

We computed the number of days since May 26, 1968 (a Sunday), using an algorithm given by Press *et al.* (1988). If the number of days modulo 7 is 0 or 6, the day is a weekend. We also entered the academic schedule given by the organizers via email in October 1994, and classified days as in-session or out-of-session according to the schedule. The day and session flags were encoded as 0 (weekend or not in-session) and 1 (weekday or in-session).

After we pasted together time, flag, environmental inputs, and a target variable, we struck out all the lines containing missing values (encoded as -99), and the resulting file was used as training data. From 2250 data

Table 1: Summary of Wald Tests on Input Variables, EC Building

Target Variable					
WBE	W	MCC	W	LTEQ	W
sin(hod)	452.0	weekday	322.1	in-session	715.7
cos(hod)	432.0	sin(doy)	180.1	cos(hod)	535.5
in-session	333.9	cos(doy)	176.9	weekday	446.1
cos(doy)	288.2	*in-session*	83.8	sin(hod)	250.2
weekday	194.4	*sin(hod)*	37.7	cos(doy)	215.2
RH	110.0	RH	32.6	sin(doy)	108.6
solar	107.9	*temp*	27.1	RH	95.8
sin(doy)	94.9	WIND	25.3	*solar*	93.7
temp	88.0	SOLAR	21.4	*temp*	62.2
WIND	27.9	COS(HOD)	14.7	WIND	20.1

Target Variable			
CHW	W	HW	W
sin(doy)	149.8	sin(doy)	312.6
in-session	99.5	cos(doy)	266.7
wind	88.7	in-session	173.0
temp	80.3	*weekday*	89.8
RH	69.5	*wind*	85.0
weekday	64.5	*temp*	81.7
cos(doy)	48.8	RH	75.2
cos(hod)	42.7	*sin(hod)*	43.2
solar	36.1	*cos(hod)*	30.2
sin(hod)	33.5	*solar*	13.2

Table 2: Summary of Wald Tests on Input Variables, Business Building

Target Variable					
WBE	W	MCC	W	LTEQ	W
in-session	148.1	cos(hod)	618.4	weekday	462.9
cos(hod)	146.1	sin(hod)	521.6	sin(hod)	424.5
sin(hod)	137.1	weekday	421.8	cos(hod)	409.3
weekday	130.2	sin(doy)	419.7	sin(doy)	189.3
temp	117.7	cos(doy)	331.6	in-session	180.9
cos(doy)	84.4	solar	237.3	cos(doy)	145.7
wind	73.0	temp	205.2	*solar*	89.5
RH	51.7	in-session	198.9	*temp*	79.6
sin(doy)	44.8	RH	197.5	*wind*	53.7
SOLAR	22.9	*wind*	68.8	RH	26.8

Target Variable			
CHW	W	HW	W
sin(doy)	109.4	sin(doy)	335.3
cos(doy)	107.1	RH	186.7
in-session	76.9	weekday	151.9
temp	56.1	temp	142.3
weekday	54.6	in-session	133.2
cos(hod)	54.1	cos(hod)	120.7
sin(hod)	44.5	cos(doy)	117.8
RH	42.1	sin(hod)	110.5
wind	26.4	*wind*	80.7
solar	24.1	SOLAR	44.3

(WBE, Business Building) to 4300 data (WBE, EC and CHW, EC) were contained in the training data files. In contrast, the Business Building data file (including incomplete data) supplied by the contest organizers contained 4872 data; likewise the EC data file (including incomplete data) contained 7944 data. To better condition the gradient descent method used for error minimization, we normalized all input and output variables so each variable had mean zero and variance one. The mean and variance used for normalization were those of the training data. That is, for each input or output variable x, the normalized input $\tilde{x}$ is computed as

$$\tilde{x} \leftarrow \frac{x - \bar{x}}{s}. \tag{18}$$

Of course, normalized outputs $\tilde{y}$ must be inverse transformed to recover values in the original units. So for each output variable,

$$y \leftarrow s\tilde{y} + \bar{y}. \tag{19}$$

Results

Estimating Savings due to Retrofits

The stated goal of the energy prediction contest was to encourage the development of models that could be used to estimate energy savings due to heating, cooling, and lighting equipment retrofits. The models developed by contestants were used to predict what the energy use of the building 'would have been' if retrofits were not carried out.

As noted by the contest organizers (Haberl *et al.*, 1996), all contestants gave similar estimates for energy savings due to retrofits for the EC, but wildly differing estimates for the Business Building. This difference between the buildings may be due to two interrelated causes. First, only half a year of Business Building data was available for training, while the post-retrofit test period lasted an entire year. An examination of the predictions of Business Building CHW energy use suggests that inaccurate extrapolations were made during the part of year for which there was no training data. Second, predicting energy use variables for the EC appears to have been easier, on the whole, than for the Business Building. Prediction errors were generally smaller for EC targets than for Business Building targets. This greater accuracy for EC targets may reflect the greater amount of training data available.

Effect of Reducing the Number of Inputs

Having described Wald's test and the use of the autocovariance function for input selection, it is of interest to compare networks having a full complement of input variables with those which have been reduced by input selection techniques. Some limited investigations were carried out. The available evidence suggests that Wald's test can distinguish 'more-relevant' from 'less-relevant' inputs, but the critical value of W which was chosen was much too small. That is, fewer inputs could have been included than were included in networks generated for the Shootout II. As for time lag selection, a test on one input variable (namely CHW, EC) shows that using too many time lag inputs degrades performance, while using a few improves performance slightly. Performance was only slightly better for a network with time lags chosen by the autocovariance, compared to a network with the same number of lags spaced one hour apart. This suggests that the environmental variables are not over-sampled, the sampling interval being one hour.

Results of training reduced networks predicting MCC are shown in Table 3, and corresponding results for CHW are shown in Table 4. In each table, mean square error (MSE) is shown for networks which takes some or all of ten input variables (sine and cosine of day of year, sine and cosine of hour of day, weekday/weekend, in session/not in session, ambient temperature, relative humidity, insolation, and windspeed). No time-lag values are included; only values at the current hour are used. Each network had eight hidden units. First the MSE is shown for a network taking all ten inputs, then MSE is shown for successively reduced networks. Variables are removed according to the results of Wald tests shown in Tables 1 and 2. Variables with lower W are removed first. For MCC (Business) and CHW (EC), the final lines of Tables 3 and 4 show performance of networks trained with the variable of highest W omitted. Reduced networks are trained on the same data as the larger networks; this could lessen the advantage of input removal, since more data become available for training the fewer input variables there are.

Wald's test applied to input variables for the EC (Table 1) shows that for MCC, we should cut cosine of hour of day, temperature, insolation, and windspeed. Performance is indeed slightly better with these inputs omitted. Wald's test applied to the Business Building (Table 2) shows all ten input variables are relevant to predicting MCC. However, as shown in Table 3 performance was improved by omitting some of the variables, namely windspeed, RH, in session/not in session, and temperature. Apparently Wald's test overestimated the relevance of these variables. On the other hand, performance is degraded if the variable with greatest W (cosine of the hour of the day) is omitted. This suggests

Table 3: Full and Reduced Networks, MCC. Instantaneous Inputs Only

Input Variables	MSE, kW^2
EC:	
sd, cd, sh, ch, ww, in, t, r, s, w	193
sd, cd, sh, ww, in, r	189
Business:	
sd, cd, sh, ch, ww, in, t, r, s, w	366
sd, cd, sh, ch, ww, in, t, r, s	361
sd, cd, sh, ch, ww, in, t, s	356
sd, cd, sh, ch, ww, t, s	322
sd, cd, sh, ch, ww, s	322
sd, cd, sh, ch, ww	323
sd, sh, ch, ww	525
sd, cd, sh, ww, in, t, r, s, w	526
Key: sd=sin(doy), cd=cos(doy), sh=sin(hod), ch=cos(hod), ww=weekday/weekend, in=in session/ not in session, t=temp, r=RH, s=solar, w=windspeed.	

Table 4: Full and Reduced Networks, CHW. Instantaneous Inputs Only

Input Variables	MSE, (MBtu/h)^2
EC:	
sd, cd, sh, ch, ww, in, t, r, s, w	0.383
sd, cd, ch, ww, in, t, r, s, w	0.380
sd, cd, ch, ww, in, t, r, w	0.358
sd, cd, ww, in, t, r, w	0.365
sd, ww, in, t, r, w	0.556
cd, sh, ch, ww, in, t, r, s, w	0.483
Key: sd=sin(doy), cd=cos(doy), sh=sin(hod), ch=cos(hod), ww=weekday/weekend, in=in session/ not in session, t=temp, r=RH, s=solar, w=windspeed.	

Wald's test did correctly distinguish cosine of the hour of the day as an important variable. Reduced networks were also trained to predict CHW, for the EC only, as shown in Table 4. Wald's test shows that for this target all ten inputs are relevant. Again, we find that variables low on the relevance list (low W in Table 1) could have been cut out, while cutting out the variable with highest W (sine of the day of the year, in this case) degrades performance.

A test conducted on CHW (EC) prediction shows a slight advantage to selecting time-lag inputs according to the autocovariance. Three networks were trained, each with 20 hidden units. Each network took sine and cosine of day of year, sine and cosine of hour of day, weekday/weekend, in session/not in session, ambient temperature, insolation, and windspeed as inputs. Varying numbers and time-lags for temperature and insolation were tested:

1. Many time lags: 22 one-hour lags for temperature and 16 one-hour lags for insolation.
2. Few time lags: two one-hour lags for temperature and two one-hour lags for insolation.
3. Few time lags: two 11-hour lags for temperature and two 8-hour lags for insolation. These were the time-lag inputs used in the competition networks.

The 22 temperature lags and 16 insolation lags for network (1) were chosen to match the total length of the lags in the competition networks (3). Discouragingly, performance among these three was rather similar. Scheme (3) was best (MSE = 0.297 (MBtu/h)^2), scheme (2) was close (MSE = 0.312 (MBtu/h)^2), and scheme (1) performed worst (MSE = 0.342 (MBtu/h)^2). For comparison, a network taking only instantaneous inputs of temperature and insolation and having 20 hidden units like the networks described here had MSE 0.313, almost the same as the network with 2 one-hour lags. It may be that scheme (1) did most poorly due to the large number of inputs (47 altogether), so that there were many more weights in (1) than in (2) or (3). That scheme (2) performed very nearly as well as (3), used in the competition, suggests that even at one-hour intervals the temperature and insolation are not much over-sampled.

Analysis of Residuals

From a statistical perspective, a study of the residuals

$$e_t = y_t - \hat{y}_t = y_t - f(\mathbf{x}_t, \hat{\mathbf{w}}) \qquad (20)$$

is important because it gives clues about the appropriateness of the model used to fit the data. (The residual is just the difference between a target value from the training set and the corresponding predicted value.) Recall that the regression model given by Eq. 4 assumes that there is additive, independent, constant-variance Gaussian noise. The residuals should have these same characteristics. Three aspects of the residuals for the building energy prediction task were studied: whether the residuals are Gaussian distributed; whether the residuals are functions of the input variables; and whether the residuals are temporally correlated among themselves. Let us consider each of these points in turn.

Distribution of Residuals. — Histograms and scatterplots of residuals were studied to investigate the distribution of residuals. On the whole, it appears that

most residuals pile up in suitably mound-shaped distributions, but for a few of the prediction tasks there were significant numbers of outliers. Outliers are a problem for least-squares regression, because such points have greater influence on the total squared error. The presence of outliers suggests that the Gaussian noise assumption is not quite right.

Dependence of Residuals on Inputs. — The variance of the residuals is significantly correlated with both environmental variables (ambient temperature and insolation) and with day type variables (weekday/weekend and in-session/not in-session). While the dependence on environmental variables appears practically unimportant, in some cases the variance according to one day type is much greater than the variance according to other day types, and this seems practically important.

Temporal Autocorrelation Among Residuals. — All ten series of residuals show temporal autocorrelation. The test statistic for temporal autocorrelation was the correlation coefficient of the residual at time t with the residual at the previous time $t-1$. The correlation coefficients for the residuals series of the ten target variables range from 0.44 to 0.92; most are in the range 0.6 to 0.9. These values are highly statistically significant. It appears that the temporal autocorrelation of residuals is also practically important. Comparing a target time-series (a segment from the training data) with predictions made by a network (Figure 4), it is clear that predictions tend to be too high or too low for several steps in sequence. This may indicate the presence of a relevant variable which affects the output, but which is not included in the set of network input variables. For example, the number of people in the building may be such a variable.

The presence of outliers, dependence on day-type inputs, and temporal autocorrelation suggest that the independent Gaussian noise regression model is incorrect. A more complicated regression model will be needed to deal with these phenomena, including, perhaps, removal of outliers before training, allowing noise magnitude to vary with day type (a form of weighted regression), and an autocorrelated noise model.

Comparison of Neural Network to Multi-Linear Regression

Given the heavy computational burden of training a neural network, one might reasonably ask what benefit is gained over multi-linear models, for which parameters can be very quickly estimated. To investigate the relative merits of multi-linear and nonlinear models, predictions made by multi-linear models were calculated for the same hold-out sets as were withheld from the neural network models. The multi-linear models studied took the same input variables as the neural networks, and the same training sets were used. Thus the inputs variables and the training and testing data were the same, and all that changed was the form of the input-output mapping.

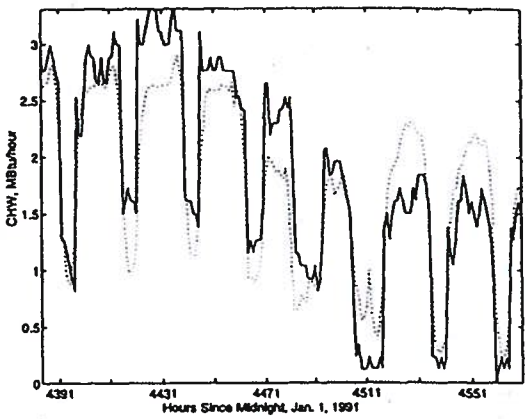

Figure 4: Target values (solid line) of CHW, Business Building, compared to predictions (dotted line) made by a network. When a prediction is too high or too low, the next prediction tends to be too high or too low also.

It was found that nonlinear models are substantially more accurate than linear models, although the improvement varies from target variable to target variable. The mean square errors of the neural networks used in the prediction competition were about 40% to 90% as large as the mean square errors of linear predictors, so modest reductions of sum squared error are possible if one uses a nonlinear model instead of a linear model. This suggests that the true relation between environmental and calendar inputs and building energy use outputs is nonlinear, but only mildly so. These results are described at greater length by Dodier (1995).

Conclusions and Future Work

The results of Wald's test applied to the energy prediction data (Tables 1 and 2) indicate that day and time variables are more relevant to predicting energy use targets than the environmental variables. This suggests that occupancy drives energy use more strongly

than does the environment. As applied to the energy data, the Wald test seems to have been more conservative than necessary; as discussed in Sec. , more inputs could have been removed while improving or maintaining prediction accuracy. This may be due to two causes. First, the significance α could have been made smaller, causing inputs to be pruned more often. Second, the Wald test was derived under assumptions that probably do not hold in the energy prediction task. The utility of selecting time-lag variables according to the autocovariance function is uncertain; as temperature and insolation were not the most important variables for prediction, including time-lag values of these variables had only a weak effect on prediction.

There are four substantial shortcomings in the work presented here, which could be rectified with further research. First, the uncorrelated, constant-variance noise model is apparently incorrect. Second, it may be possible to improve the model selection process by making use of a test statistic which does not assume a particular asymptotic approximation, as the Wald test used here assumes an asymptotic linear approximation. Third, there are many missing values in the data provided for estimating model parameters. As we excluded all incomplete data, we were deprived of a very great amount of data. There are ways to make use of incomplete data, both in training and in computing outputs. Fourth, it is usual in engineering problems that a partial or approximate input-output model is known *a priori*. Yet neural network and other regression models assume no prior knowledge of the problem domain. It is desirable to base the regression model on the already-known aspects of the problem, but it is not clear how to go about this task.

Acknowledgments

The authors thank Profs. Jeff Haberl, Texas A & M University, and Jan Kreider, U. Colorado at Boulder, for organizing the competition. Thanks also to the anonymous reviewers for their insightful comments. This research was supported by NSF Presidential Young Investigator award IRI-9058450 and grant 90-21 from the James S. McDonnell Foundation to Michael C. Mozer.

References

Abarbanel, H. D. I. (1992) Chaotic Signals and Physical Systems. Unpublished preprint.

Cybenko, G. (1989) Approximation by Superpositions of a Sigmoidal Function. *Mathematics of Control, Signals, and Systems*, 2:303–314.

Dodier, R. (1995) Statistical Properties of Neural Networks, with Application to Building Energy Prediction. Master's thesis. U. Colorado, Boulder, Colorado.

Feuston, B., and J. Thurtell. (1994) Generalized Nonlinear Regression with Ensemble of Neural Nets. *ASHRAE Trans.*, 100(2):1075–1080.

Hornik, K., M. Stinchcombe, and H. White (1989) Multilayer Feedforward Networks are Universal Approximators. *Neural Networks*, 2:359–366.

Kreider, J., D. Claridge, P. Curtiss, R. Dodier, J. Haberl, and M. Krarti. (1995) Building Energy Use Prediction and System Identification using Recurrent Neural Networks. *J. Solar Energy Engineering* 117(3):161–166.

Mackay, D. (1994) Bayesian Nonlinear Modeling for the Prediction Competition. *ASHRAE Trans.*, 100(2):1053–1062.

Ohlsson, M., C. Peterson, H. Pi, T. Rognvaldsson, B. Soderberg. (1994) Predicting System Loads with Artificial Neural Networks. *ASHRAE Trans.*, 100(2):1063–1074.

Press, W., B. Flannery, S. Teukolsky, and W. Vetterling. (1988) *Numerical Recipes in C.* Cambridge: Cambridge University Press.

Raftery, A. (1994) Bayesian Model Selection in Social Research. Working Paper 94-12, Ctr. Studies in Demography and Ecology, U. Washington. [ftp.stat.washington.edu: /pub /tech.reports /bic.ps]

Seber, G. A. F., and C. J. Wild (1989) *Nonlinear Regression.* New York: John Wiley and Sons.

Haberl, J., S. Thamilseran, and J. Kreider (1996) Predicting Hourly Building Energy Use: The Great Energy Predictor Shootout II: Measuring Retrofit Savings – Overview and Discussion of Results. To appear in *ASHRAE Trans.* 102(2).

Wald, A. (1943) Test of statistical hypotheses concerning several parameters when the number of observations is large. *Trans. Am. Math. Soc.* 54:426–482.

Weigend, A., B. Huberman, and D. Rumelhart. (1990) Predicting the Future: A Connectionist Approach. *Int'l J. Neural Systems* 1:193–209.