Practical Guidelines for Developing BP Neural Network Models of Measurement U...
Chang-Xue Jack Feng; Abhirami C Gowrisankar; Alice E Smith; Zhi-Guang Samuel Yu
Journal of Manufacturing Systems; 2006; 25, 4; ABI/INFORM Global
pg. 239

Journal of Manufacturing Systems
Vol. 25/No. 4
2006

Practical Guidelines for Developing BP Neural Network Models of Measurement Uncertainty Data

Chang-Xue Jack Feng (*cfeng@bradley.edu*) and **Abhirami C. Gowrisankar,** Dept. of Industrial and Manufacturing Engineering, Bradley University, Peoria, Illinois, USA
Alice E. Smith, Dept. of Industrial and Systems Engineering, Auburn University, Auburn, Alabama, USA
Zhi-Guang Samuel Yu, Dept. of Industrial and Manufacturing Engineering, Bradley University, Peoria, Illinois, USA

Abstract

The predictability of measurement uncertainty is a critical issue in quality assurance. The performance of such a process cannot be predicted if no proper mathematical model is available. For manufacturing processes where no satisfactory analytical model exists, or where a low-order empirical polynomial model is inappropriate, neural networks offer a good alternative predictive modeling approach. This paper considers the primary decisions and activities that arise during back-propagation (BP) neural network model construction, selection, and validation for this novel application. Computational experiments were designed to cross-examine the two types of hidden layers of networks with different hidden neurons, training tolerances, and testing tolerances based on the *v*-fold cross-validation technique. Hypothesis testing is used to evaluate different kinds of prediction errors from the models developed. The best models were very accurate in generalizing the measurement uncertainty prediction, and thus are suitable for use in a manufacturing environment. This study reveals no statistical advantages of using a two-hidden-layer net over a one-hidden-layer net in modeling the measurement uncertainty data and offers some general and simple rules for BP model selection and validation.

Keywords: *Measurement Uncertainty Study, Neural Networks, Predictive Modeling, Cross-Validation, Data Mining, Design of Experiments*

Introduction

The predictability of the measurement uncertainty is a critical issue in quality assurance. The performance of such a process cannot be predicted if no such proper mathematical model is available. A model may originate from introspection or observation or both (Gershenfeld 1999). Although developing an analytical model is feasible in some simplified situations, most manufacturing processes, such as ones described herein, are complex, and therefore, models that are less general, more practical, and less expensive than the analytical models are of interest. Box and Draper (1987) and Gershenfeld (1999) classified mathematical models into analytical (or mechanistic) and empirical (or observational).

Both regression analysis (RA) and neural networks (NN) have been used for years in empirical modeling and have only recently been termed *data mining* (Witten and Frank 2001; Groth 1998). For a comparison of regression and NN predictive modeling methods for manufacturing processes, refer to Feng and Wang (2003, 2004) and Smith and Mason (1997). An important advantage of using NN over regression in process modeling is its capacity in dealing with multiple outputs or responses while each regression model is able to deal with only one response. This advantage is shown in the second case study presented later in the paper, where the location of a hole in the X and Y directions is of interest. Another major advantage for developing NN process models is that they do not depend on simplified assumptions such as linear behavior or production heuristics. If a trained NN model represents the process with fidelity, the NN can be used to optimize the controllable process parameters with regard to some cost metric that measures final product quality, rework and scrapping, labor cost during production, or use of raw materials. Optimization takes place by searching over the range of controllable variables for a combination of settings that yields the best performance on the cost function (Coit, Jackson, and Smith 1998). A primary advantage of using regression over

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

NN is its explicitness whereas a NN model is a black box. Moreover, a regression model is more straightforward to build, validate, and interpret. As a result, a regression model is preferred to a NN model whenever they are both applicable and perform statistically the same. This is illustrated in Feng and Wang (2004, 2003, 2002b) and Williams and Feng (2005).

Neural networks possess a number of attractive properties for modeling a complex manufacturing process or a system: universal function approximation capability, resistance to noisy or missing data, accommodation of multiple nonlinear variables for unknown interactions, and good generalization capability (Coit, Jackson, and Smith 1998; Twomey and Smith 1998). In many applications, it is not possible to collect large data sets relative to the number of inputs and outputs due to cost and time constraints. Validation of NN models resulting from small datasets benefits from the use of various resampling techniques such as the bootstrap and jackknife methods. Twomey and Smith (1998) and Ingrassia and Morlini (2005) reported studies of predictive NN modeling using small datasets.

Among the various neural networks models, back propagation is the best general-purpose model and probably the best at generalization (Mitchell 1997; Lawrence 1994). It is a supervised learning scheme by which a layered feed-forward network is trained to become a pattern-matching engine. For a good introduction on this and other kinds of neural networks, refer to Bishop (1995), Lawrence (1994), Swingler (1996), and Wasserman (1989).

Configuring the Neural Nets

In BP neural networks modeling, some problems may not need any hidden layers. Linear and generalized linear models are useful in a wide variety of applications (McCullagh and Nelder 1989). And even if the function to learn is mildly nonlinear, better generalization may be obtained with a simple linear model than with a complicated nonlinear model if too little data are available or too much noise is present to accurately estimate the nonlinearities. *Figure 1* shows a typical one-hidden-layer (OHL) neural net with 10 hidden neurons, multiple inputs, and one output.

Determining the Number of Hidden Layers

Determining the number of hidden layers and the number of neurons in each hidden layer is a considerable task. *The number of hidden layers is usually determined first* and is a critical step. The number of hidden layers required depends on the complexity of the relationship between the inputs and the outputs. Most problems only require one hidden layer, and if the input/output relationship is linear the network does not need a hidden layer at all. It is unlikely that any practical problem will require more than two hidden layers (THL). Cybenko (1989) stated that one hidden layer is enough to classify input patterns into different groups. Experiments conducted by Bounds et al. (1988) revealed no obvious evidence that a network with two hidden layers performed better than a network with one hidden layer only.

In multi-layer perceptrons (MLPs) with step/threshold/Heaviside activation functions, two hidden layers are needed for full generality (Sontag 1992). For further discussion, see Bishop (1995, pp121-126). In MLPs with any of a wide variety of continuous nonlinear hidden-layer activation functions, one hidden layer with an arbitrarily large number of units suffices for the "universal approximation" property (see, for example, Hornik, Stinchcombe, and White 1989 and Bishop 1995, p130). But no theory is available to tell how many hidden units are needed to

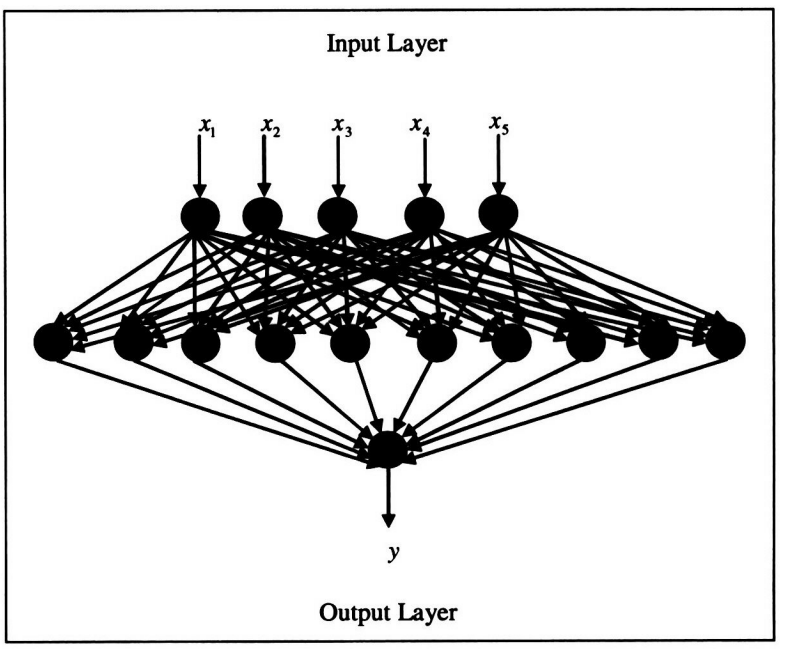

Figure 1
A Neural Network Architecture

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

approximate any given function. If only one input is available, one sees no advantages in using more than one hidden layer. But things get much more complicated when two or more inputs are present.

However, Chester (1990) argued that a THL should perform better than an OHL network. More than one hidden layer can be useful in certain architectures, such as cascade correlation (Fahlman and Lebiere 1990), and in special applications, such as the two-spirals problem (Lang and Witbrock 1988) and ZIP code recognition (Le Cun et al. 1989). A simple explanation for why larger networks can sometimes provide improved training and lower generalization error is that the extra degrees of freedom can aid convergence; that is, the addition of extra parameters can decrease the chance of becoming stuck in local minima or on "plateaus." The most commonly used training methods for back-propagation networks (and many other neural network paradigms) are based on gradient descent; that is, error is reduced until a minimum is reached, whether it be a global or local minimum.

The rule of thumb in deciding the number of hidden layers is normally to start with OHL (Lawrence 1994, p201). If OHL does not train well, then try to change the number of neurons or the training/testing tolerances (weights) or both. Adding more hidden layers should be the last option. In this study, predictive NN models were developed based on experimental data from two measurement uncertainty studies. Each of the two case studies was examined with OHL and THL using the same methodology documented in the next main section of this paper (Methodology in Comparing Neural Nets). It is hoped that these results will assist those who are using neural networks to model manufacturing processes by giving some well-founded, general guidelines on network architecture.

Deciding the Number of Hidden Neurons

The choice of hidden neuron size is problem-dependent. For example, any network that requires data compression must have a hidden layer smaller than the input layer (Swingler 1996, p52). A conservative approach is to select a number between the number of input neurons and the number of output neurons (Kusiak 2000, p364). Hecht-Nelson (1987) used Kolmogorov's theorem—which states that any function of I variables may be represented by the superposition of a set of $2i + 1$ univariate functions—to derive the upper bound for the required number of hidden neurons, where I is the number of inputs. This can be expressed as:

$$h = 2i + 1 \tag{1}$$

Therefore, Hecht-Nelson determined that $2i + 1$ should be used as the upper bound of the number of hidden neurons required for a one-hidden-layer back-propagation network.

Girossi and Poggio (1989) pointed out that Kolmogorov's theorem requires that the component functions are chosen to fit each particular case and not, as is the case with neural networks, that the functions are fixed and parameterized. They also pointed out that a neural network requires a smooth function in order to generalize and that Kolmogorov's theorem does not guarantee this. Kůrková (1991), however, stated that the fact that a neural network is only an approximation eliminates both of these difficulties.

Lawrence and Fredrickson (1998, p. II-12) suggested that a best estimation for the number of hidden neurons is to half the sum of inputs and outputs, that is,

$$h = (i + o)/2 \tag{2}$$

where o is the number of output neurons. An alternative proposed in Lawrence and Fredrickson (1998, p. II-13) is to relate the training data to the number of hidden neurons, as follows:

$$\frac{N}{10} - i - o \le h \le \frac{N}{2} - i - o \tag{3}$$

where N is the number of training observations. Lawrence and Fredrickson (1998, p. II-13) further suggested that, to choose between the above wide ranges of numbers, the training tolerance must be considered. *In general, a tighter tolerance requires that the network be trained with fewer hidden neurons.*

Marchandani and Cao (1989) derived the following relationship:

$$h = i \log_2 P \tag{4}$$

where P is the number of training observations. Baum and Haussler (1989) presented some expres-

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

sions related to training observations, neurons, and weights under the consideration of various statistical confidence levels. Lippmann (1987) indicated that the maximum number of hidden neurons for a OHL network is $o(i + 1)$. Swingler (1996, p55) and Lawrence (1994, p200) also suggested:

$$h \geq \frac{N}{i} \times e \tag{5}$$

where e is the training tolerance.

It can be seen that the general wisdom concerning selection of initial number of hidden neurons is somewhat conflicting. *A good rule of thumb is to start with the number of hidden neurons equal to ½ the number of input neurons, then either add neurons if the training error remains above the training error tolerance, or reduce neurons if the training error quickly drops to the training error tolerance.* The objective is to find the most compact model (fewest trainable weights and, therefore, fewest hidden neurons) that can model the relationship to the degree of accuracy desired. Extra model parameters (trainable weights) add computational burden (both storage and operational time) while increasing the likelihood of overfitting, causing poor generalization.

Determining the Number of Training Data

The next issue is to decide the number of data that should be used to train the neural network well. An overfitted model could approximate the training data well but generalize poorly to the validation data set. On the other hand, an underfitted model would generalize to the validation data set well but approximate the training data poorly. To avoid overfitting and underfitting is to determine the best number of training observations. No general guidelines are available to achieve this. However, Lawrence and Fredrickson (1998, p. II-12) suggested the following *rule of thumb*:

$$2(i+h+o) \leq N \leq 10(i+h+o) \tag{6}$$

Another rule of thumb is to relate the required number of training observations to the training tolerance using Eq. (7):

$$N \leq \frac{hi}{e} \tag{7}$$

Changing the Weights

A back-propagation network (BPN) is sensitive to initial values of weights (Kolen and Pollack 1990). Initial weights that are too small increase the learning time, which may cause difficulties in converging to an optimal solution (Kusiak 2000, p365). If initial weights are too large, the network may get unstable weights (Wasserman 1989).

The neural net shown in *Figure 1* can be considered a statistical model represented by the following function:

$$y = f\left(\sum_{i=1}^{n} w_i x_i\right) = f\left(\mathbf{w}^t \mathbf{x}\right)$$

where $\mathbf{w}$ is the weight vector:

$$\mathbf{w} = \left[w_1\ w_2 \ldots w_n\right]^t$$

and $\mathbf{x}$ is the input vector:

$$\mathbf{x} = \left[x_1\ x_2 \ldots x_n\right]^t$$

The function $f(\mathbf{w}^t\mathbf{x})$ is the transfer function, and its domain is the set of activation values, or *net*, of the neural network paradigm, where *net* is the scalar product computation (Twomey and Smith 1998):

$$net = \mathbf{w}^t \mathbf{x}$$

The most typical continuous activation function is sigmoidal:

$$f(net) = 1/\left(1 + e^{(-net)}\right)$$

This produces output values between 0 and 1 for all input values.

The weight matrices used in above discussion are trained before the use of the NN model, that is, $\mathbf{w}$ transits from a random state to an equilibrium state, storing values of the model parameters. In the most well-known supervised technique, the back-propagation NN adjusts the initial randomized weights during training according to the steepest gradient along the error surface. Weights are adjusted in proportion to their contribution to the output by recycling the squared error signal back through the layers of weights until the error reaches the training and testing tolerances and becomes stable. For this paper, the starting default training tolerance was 0.10, and a default testing tolerance of 0.40 in the software.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Methodology in Comparing Neural Nets

In designing the neural networks, the first step was to identify the type of network and development software to be used, identify the input and output parameters, gather data and choose a validation method. This research uses the data from two measurement uncertainty studies. Two different plans of the parameters were used in building the neural networks. The first plan used OHL nets for network design and two levels of each of the following parameters: hidden neurons, training tolerances, testing tolerances, and threefold cross-validation, leading to a total of 24 neural networks, that is, $2 \times 2 \times 2 \times 3 = 24$ networks. The other plan used THL nets for network design and two levels of the same three parameters in the OHL nets, with a threefold cross-validation, leading to a total of 48 experiments, that is, $2 \times 2 \times 2 \times 3 \times 2 = 48$ networks. This follows the train-test-validation procedure outlined in Witten and Frank (2001).

BrainMaker (Lawrence and Fredrickson 1998) from California Scientific was available and was used to create the neural networks. It is a back-propagation-based feed-forward multi-layer perceptron network with the sigmoid function. It has been used for similar predictive data mining projects (see, for example, Feng and Wang 2004; Feng and Wang 2003; Feng and Yu 2003; Feng, Wang, and Yu 2002; Feng and Wang 2002b; and Liao et al. 2001) and was considered an appropriate software package for this project.

Cross-Validation

Cross-validation (CV) is a method where the limited amount of data is divided into groups and each group is 'held out' once a time for validation. CV is considered a robust method of creating and checking empirical models and, when used with stratification, helps to eliminate bias that may be found within the data set. Tenfold CV is often considered a standard error evaluation method in data mining (Witten and Frank 2001). Many considered fivefold to be as good as tenfold (for example, see Breiman and Spector 1992 and Zhang 1993). However, in engineering practice threefold CV is often used and considered sufficient (Witten and Frank 2001). In addition, Feng, Wang, and Yu (2002) and Feng, Yu, and Kusiak (2006) demonstrated that a threefold CV method for a system with multiple inputs and multiple outputs is appropriate. Twofold CV is considered to be the worst option and leads to the largest prediction errors (Zhang 1993). Yu and Feng (2004) did not reveal any statistical advantages in using fivefold CV over threefold CV. This study used a threefold CV. See Twomey and Smith (1998) for a discussion on cross-validation and other methods of neural network validation for sparse data.

Hypothesis Testing of Equal Means and Variances of Predictions and Observations

To validate that the predictions from any learning scheme and the corresponding observations or the prediction errors have statistically equal means and variances, hypothesis testing is used (Witten and Frank 2001; Mitchell 1997). Otherwise, any conclusions drawn by simply looking at the error magnitude would often be incorrect and misleading (Feng and Wang 2003; Feng, Wang, and Yu 2002). For data following a normal distribution, the t-test can be employed to compare the means of predictions and the corresponding observations or their absolute/relative errors. Because the two-sample t-test relies on the equal variance assumption, an F-test is conducted before this t-test. Of particular interest is the paired-t test because one is always interested in comparing the predictions and observations in pairs given the same set of input conditions. The paired t-test does not require equal variance assumption; therefore, the F-test is not required prior to the paired t-test. For data with non-normal or unknown distributions, the nonparametric Mann-Whitney test was used instead of the t-test to compare the means or medians, and the Levene's test was used instead of the F-test to compare the variances.

The studies of Feng and Wang (2004, 2003, 2002a) and Feng, Wang, and Yu (2002) have shown that using hypothesis testing to qualify predictive models is more stringent than using the usual α value, e.g., 0.05, in selecting regression models. The latter is normally adequate in selecting descriptive models and tends to select a larger model than the former. Sometimes more than one model candidate is qualified by the mean and variance tests and other model selection techniques, such as the prediction error rates based on cross-validation methods briefly discussed in the next section, and the necessary domain knowledge have to be applied to choose the best model(s).

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Prediction Error Evaluation

The error between the predicted values and the observations is a good measure in evaluating the goodness of a predictive model. Some authors suggest that at least 15 data be available to obtain a reasonable evaluation of prediction errors (for example, see Montgomery, Peck, and Vining 2001). This research will follow this principle in deciding the number of validation data. The following three traditional estimates from forecasting techniques are often used: (1) mean absolute deviation (MAD), (2) mean absolute percent error (MAPE), and (3) mean root squared error (MRSE). They are mathematically defined next, respectively:

$$\text{MAD} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{8}$$

$$\text{MAPE} = \left(\frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right|\right)\times 100 \tag{9}$$

$$\text{MRSE} = \frac{1}{n}\sqrt{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{10}$$

where y_i is the observed response and $\hat{y}_i$ is the predicted response. Additional measures of prediction error rates are summarized in Witten and Frank (2001, p148) and Burnham and Anderson (2002).

Example 1: Predictive Modeling of Experimental Data from Gage R&R Study

One of the procedures in a gage repeatability and reproducibility (R&R) study is to measure a certain number of parts of the same dimension with the gage of interest, using different operators. Variability in measurement comes from the measurement device itself and the operator. An R&R study estimates the contribution of these two sources of variability. For more details on gage R&R study, refer to Burdick, Borror, and Montgomery (2003); AIAG (2002); Montgomery (2005); and Barrentine (1991).

Factors considered in this case study are the operators, the runs or replicates, and the parts. The dataset consists of measurements of 40 shafts using a digital caliper. These shafts are supposed to be one inch in diameter and were produced by a production type of CNC turning center in the Bradley University CIM Lab. Sixteen operators measured each shaft diameter twice. This design of experiments gave a total of 1,280 measurements. One important point about the dataset is that the factors are categorical and the response is continuous. More details about the design of experiments and data were reported in Narayanan (2003).

Design of Computational Experiments

While developing neural network models with OHL, training tolerances of 10% and 15% and testing tolerances of 20% and 40% were used after some trial runs. Two levels of hidden neurons were used. The first hidden layer neuron size of three equals the number of factors or inputs in the R&R study. The second level with seven neurons was obtained from Eq. (1). As the threefold CV was applied, this resulted in a total of 24 neural networks models with eight models for each division of data. When applying THL nets, the same training and testing tolerances and hidden neurons used in the OHL were employed to facilitate the comparison. For threefold cross-validation, this resulted in a total of 48 models with 16 models for each division of data.

Equation (6) resulted in a range between 22 and 70 as the smallest number of data for training and testing of the networks that would still yield acceptable performance. Because the original data size was much larger, a random sample of 80 experiments consisting of measurements of eight parts, five operators, and two replicates was used. Thus, input factor "part #" ranged between 1 and 8, "operator #" between 1 and 5, and "replicate #" between 1 and 2. This resulted in a total of 54 or 53 data in each training/testing set, which is more than the minimum 15 data that are required to give a reasonable estimate of prediction errors.

Computational Results with One Hidden Layer

A summary of the prediction error statistics MAD, MAPE, and MRSE based on Eqs. (8) to (10) from the 24 neural network models is provided in *Table 1*. In addition, the *p*-values of the hypothesis tests are also provided in the same table. In this table, model I-X uses data folds 1 and 2 for training and testing and 3 for validation, model II-X uses folds 2 and 3 for training and testing and 1 for validation, and model III-X uses folds 1 and 3 for training and testing and 2 for validation.

Due to non-normality of data, the *p*-values from the Levene's test and the Mann-Whitney test were

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table 1
Summary of Prediction Statistics from 24 Neural Net Models with OHL

Model No.	Parameters*	MAD	MAPE	MRSE	Mann-Whitney Test *p*-values	Levene's Test *p*-values
			Data folds 1 and 2 for Training and Testing			
Model I-1	0.1/0.4/3	0.0207	0.0813	0.00470	0.8610	0.0000
Model I-2	0.1/0.4/7	0.0235	0.0924	0.00533	0.4630	0.0000
Model I-3	0.1/0.2/3	0.0210	0.0826	0.00529	0.0253	0.0020
Model I-4	0.1/0.2/7	0.0258	0.1018	0.00613	0.0007	0.0000
Model I-5	0.15/0.4/3	0.0236	0.0929	0.00637	0.0175	0.9160
Model I-6	0.15/0.4/7	0.0224	0.0883	0.00517	0.1207	0.0010
Model I-7	0.15/0.2/3	0.0268	0.1056	0.00647	0.0004	0.0000
Model I-8	0.15/0.2/7	0.0187	0.0737	0.00447	0.9927	0.0000
			Data folds 2 and 3 for Training and Testing			
Model II-1	0.1/0.4/3	0.0383	0.1505	0.00931	0.1338	0.4750
Model II-2	0.1/0.4/7	0.0332	0.1307	0.00728	0.7484	0.0190
Model II-3	0.1/0.2/3	0.0346	0.1361	0.00753	0.1086	0.0160
Model II-4	0.1/0.2/7	0.0359	0.1413	0.00823	0.4663	0.0250
Model II-5	**0.15/0.4/3**	**0.0357**	**0.1403**	**0.00817**	**0.6151**	**0.4500**
Model II-6	0.15/0.4/7	0.0369	0.1452	0.00848	0.5327	0.8190
Model II-7	0.15/0.2/3	0.0377	0.1481	0.00859	0.8556	0.5930
Model II-8	0.15/0.2/7	0.0411	0.1616	0.00902	0.0032	0.2020
			Data folds 1 and 3 for Training and Testing			
Model III-1	0.1/0.4/3	0.0302	0.1188	0.00725	0.2952	0.0000
Model III-2	0.1/0.4/7	0.0290	0.1140	0.00692	0.0001	0.0000
Model III-3	0.1/0.2/3	0.0323	0.1273	0.00800	0.0053	0.0000
Model III-4	0.1/0.2/7	0.0279	0.1099	0.00678	0.0000	0.0000
Model III-5	0.15/0.4/3	0.0293	0.1150	0.00668	0.0080	0.0000
Model III-6	0.15/0.4/7	0.0290	0.1140	0.00673	0.0033	0.0000
Model III-7	0.15/0.2/3	0.0310	0.1219	0.00711	0.0016	0.0000
Model III-8	0.15/0.2/7	0.0330	0.1301	0.00772	0.0002	0.0000

* Parameters: Training tolerance/Testing tolerance/Number of hidden neurons

reported. These *p*-values show that only four of the 24 models passed the hypothesis tests. A comparison of MAD, MAPE, and *p*-values of the Mann-Whitney test and the Levene's test is given in *Figure 2* for the four qualified models in *Table 1*. Model II-5 in *Table 1* was selected as the best model because it provided the least prediction errors and also passed the hypothesis tests with good *p*-values. This best model was developed with a training tolerance of 0.15, a testing tolerance of 0.40, and a hidden neuron size of three.

Computational Results with Two Hidden Layers

The same dataset was now modeled with the two-hidden-layer net. Model I-X used data folds 1 and 2 for training and testing and 3 for validation, model II-X used folds 2 and 3 for training and testing and 1 for validation, and model III-X used folds 1 and 3 for training and testing and 2 for validation. Because the observed data were not normal, the Levene's test and the Mann-Whitney test were performed.

Similarly, only 13 of the 48 models from the THL net passed the hypothesis tests. *Figure 3* gives a comparison of MAD, MAPE, and *p*-values of the mean and variance tests for these 13 qualified models from THL nets. Model II-16 was recommended as the best model because it provided the least prediction error values along with high *p*-values for the Mann-Whitney test and the Levene's test. This best model was developed with a training tolerance of 0.15, a testing tolerance of 0.40, and a hidden neuron size of seven in the two hidden layers.

Comparison of Best Performing OHL and THL Networks

Statistical tests are applied now to compare the two best models obtained from using different parameters in the same learning scheme. The Mann-Whitney test is used to compare the means, and the Levene's test is applied to compare the variances of the two best models, one each from the one-hidden-layer and two-hidden-layer networks, respectively. The best models

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

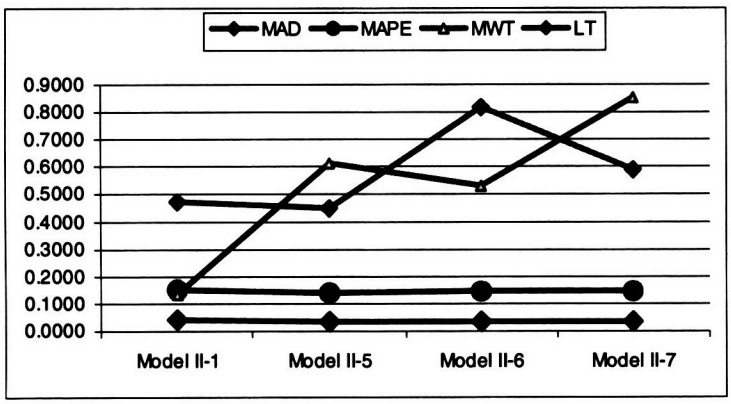

Figure 2
Comparison of Four Statistics of Qualified Models in *Table 1*

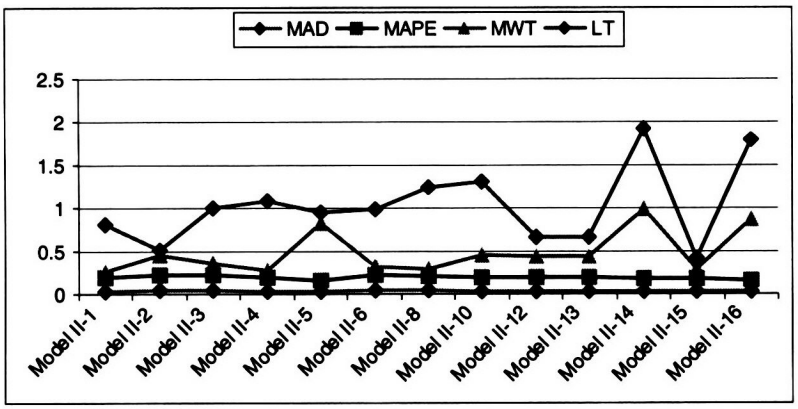

Figure 3
Comparison of Four Statistics of Qualified Models

from the one-hidden-layer network and the two-hidden-layer network do not share the same parameters of NN (tolerances, hidden layers, etc.) in this case. As the observations or predictions from these models do not follow normality, the Mann-Whitney test and Levene's test were conducted to compare the predictions from these two best models.

In addition to comparing the predictions from the two models, the absolute error and relative error to the respective observed value are also compared for better understanding. The statistics from these three sets of Mann-Whitney test and Levene's test are summarized in *Table 2*. The *p*-values resulted from comparing three types of prediction errors between predictions, and the corresponding observations are greater than 0.05, indicating that the best models from the one-hidden-layer and two-hidden-layer networks perform statistically the same in terms of these error evaluators. It is worthwhile to notice the difference of the two best models. The first one has only one hidden layer with three hidden neurons, while the second one has two hidden layers with seven hidden neurons in each layer. As simpler models are preferred for the reasons mentioned earlier, the OHL network is recommended.

Example 2: Modeling of Data from CMM Measurement Uncertainty Study

A coordinate measuring machine (CMM) is an electromechanical system widely used in digitization for reverse engineering and in inspection. It consists of a contact probe that can be positioned in the three-dimensional space relative to the surfaces of a work part; and the *x*, *y*, and *z* coordinates of the probe can be precisely recorded to obtain dimensional and geometrical data. In a CMM, the probe is fastened to a mechanical structure that allows the movement of the probe relative to the part. The part is usually located on a worktable that is connected to the structure. The contact probe is a key component of a CMM. It indicates when contact has been made with the part surface during measurement. The Mitutoyo RV507 CMM used in this case study is the fixed-bridge type of configuration with a touch-trigger probe. See Feng and Pandey (2002) for a schematic view of the CMM, and see Dotson, Harlow, and Thompson (2003) for more details about CMM metrology and configurations.

CMM Measurement Uncertainty

All measurement processes, including digitization and inspections, have some extent of uncertainty. When reporting the measurand, it is necessary to report the uncertainty associated with the quantity of interest, and measuring the uncertainty can be quite complex. Once the relationship between the measurement uncertainty and the measurement parameters were identified, empirical models were developed to aid the selection of these parameters given the complexity and the tolerable measurement uncertainty, which would either help to reduce the measurement cost or improve the measurement productivity.

In applying a CMM with a touch trigger probe to digitizing a mechanical object, the following five parameters tend to impact the uncertainty of digitiz-

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table 2
Hypothesis Testing Statistics of Predictions from Two Competitive Models

95% C.I.	Prediction	*p*-value Error	Relative Error
Levene's test	0.521	0.854	0.847
Mann-Whitney test	0.373	0.539	0.539

ing the most: probe sizes, pitch values (sampling points), travel speeds, stylus length, and feature sizes. A number of studies have been conducted during the past a few years in applying a CMM as a digitizer and reducing the errors induced in the digitization process with a CMM. For example, some reviews and related past studies have been reported in Feng et al. (2007); Feng and Pandey (2002); and Feng and Wang (2002a,b). However, none of the above studies has dealt with multiple outputs or responses as required in this case study. As indicated in the first section, regression modeling is not a viable option in this case. Therefore, NN is used to develop a predictive model of the measurement uncertainty data.

This case uses experimental data from the CMM measurement uncertainty study of the *X* and *Y* locations of a hole in a master gage ring. For details of the design of experiments and data, refer to Saal (2003). This study applied a fractional DOE of 2^{5-1}. It resulted in 16 combinations of the above five inputs and had six duplicates, leading to a total of 96 experiments. Among the five input parameters, measurement speed ranged between 25 and 100 (percent), stylus length (mm) between 31 and 51, probe to diameter ratio (no unit) between 9 and 38, measurement point (count) between 5 and 33, and starting position offset from center between 0 and 80 (percent).

Data Transformation

Data transformation has been performed with the observed and predicted responses of the datasets. The actual experiment consisted of six replicates with a total of 96 data points. To perform a threefold cross-validation, the average of each pair of replicates was computed, which resulted in three sets of averages with a total of 48 data. Also, the observed reponses from the CMM were in the form of difference between the respective mean *X*-location and *Y*-location of the part and the measured location. To avoid ill scales in hypothesis tests, prediction error computation, and neural network training, the mean values of 13.4617 mm for *X*-location and 9.7557 mm for *Y*-location were added to the observed and predicted responses, respectively. As a result, the form of MAPE in Eq. (9) becomes the form in Eq. (11).

$$\text{MAPE} = \left(\frac{1}{n}\sum_{i=1}^{n}\left|\frac{(y_i+\text{mean})-(\hat{y}_i+\text{mean})}{(y_i+\text{mean})}\right|\right)\times 100 = \left(\frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i-\hat{y}_i}{(y_i+\text{mean})}\right|\right) \quad (11)$$

The dataset was first analyzed for the one-hidden-layer net with training tolerances of 10% and 15% and testing tolerances of 20% and 40%, respectively, based on some trial runs. The number of hidden neurons used for this case study was decided based on the number of inputs. The first level of hidden neurons equals the number of inputs, or five neurons, and the second level of hidden neurons was twice of the number of inputs, that is, 10 neurons. For threefold CV, this design of computational experiments resulted in a total of 24 neural network models with eight models for each of the three divisions of data.

In the two-hidden-layer net, the same training and testing tolerances and the same hidden neurons used for the one hidden layer net were employed to facilitate comparison. For threefold CV, this gave a total of 48 models with 16 models for each division of data. Equation (6) gives the number of optimum data range between 34 and 120 data. For 96 experiments with threefold CV, a total of 64 data per training/testing run were obtained, which is between 34 and 120.

Computational Results from the One-Hidden-Layer Net

Model I-X used data folds 1 and 2 for training and testing and 3 for validation, model II-X used folds 2 and 3 for training and testing and 1 for validation, and model III-X used folds 1 and 3 for training and testing and 2 for validation. The *p*-values of all 24 models indicate that all of the models passed the hypothesis tests. Model II-4 was selected as the best model because it provided the least prediction errors and also passed the hypothesis tests with good *p*-values. This best model was developed with a training tolerance of 0.10, a testing tolerance of 0.40, and a hidden neuron size of 10.

Computational Results with Two Hidden Layers

The dataset was now modeled with the two-hidden-layer net. Again, model I-X used data folds 1

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

and 2 for training and testing and 3 for validation, model II-X used folds 2 and 3 for training and testing and 1 for validation, and model III-X used folds 1 and 3 for training and testing and 2 for validation. Because the observed data were not normal, *p*-values from the Levene's and Mann-Whitney tests were reported. These *p*-values indicate that all of the 48 models passed the hypothesis tests. Model II-8 was selected as the best model because it provided the least prediction errors along with good *p*-values. This best model was developed with a training tolerance of 0.10, a testing tolerance of 0.40, and a hidden neuron size of 10 in both of the hidden layers.

Comparison of Best Performing OHL and THL Networks

The Mann-Whitney test was used to compare the means of the two competitive models and the Levene's test was applied to compare the variances of the two models, one each from the OHL network and the THL network. In addition to comparing the predictions from the two models, the absolute error and relative error to the respective observed value are also compared for better understanding. The statistics from these three sets of Mann-Whitney test and Levene's test are summarized in *Table 3*. All of the *p*-values are greater than 0.05, indicating that the best models from the OHL and THL networks perform statistically the same in terms of these error evaluators. The one-hidden-layer network is recommended for the reasons mentioned earlier as it is simpler than the THL net.

Conclusions

This study has proven to be interesting and informative. Back-propagation-based neural networks were applied to predictive modeling of data from two gage measurement uncertainty studies, the first time (to our knowledge) that neural networks have been used in such an application. The first case has multiple inputs and a single output, while the second one has multiple inputs and multiple outputs. The performance of the predictive neural models was satisfactory for use in production. Computational results did not reveal any statistical advantage of using a THL net over an OHL net. The rules of thumb discussed for model construction and size of data set were employed successfully while the computationally expedient threefold cross-validation worked well.

The guideline for developing predictive BP NN models of measurement uncertainty data is summarized in *Figure 4*. A practitioner has to keep in mind that the results reported in this paper are empirical and thus case-dependent, although the procedure may be widely applicable. For example, another case may reveal that using THL nets would lead to statistically different performance from using OHL nets. Another point to keep in mind is that the best model chosen in the case studies is one of many qualified models and is only marginally better than some other ones in terms of prediction error.

Acknowledgments

This research was partially funded by the Bradley University Caterpillar Master's Fellowship to Ms. Abhirami C. Gowrisankar, and by a Caterpillar Fellowship and the Bradley University GRASP grant to Dr. Jack Feng. We are grateful to Ms. Geeta Narayanan and Mr. Tony Saal for providing us with the original data from their respective gage measurement uncertainty studies.

References

AIAG (2002). *Measurement Systems Analysis*, 3rd ed. Detroit: Automotive Industry Action Group.

Barrentine, L.B. (1991). *Concepts for R&R Studies*, 2nd ed. Milwaukee: ASQ Quality Press.

Baum, E.B. and Haussler, D. (1989). "What net size gives valid generalization?" *Neural Computation* (v1, n1), pp151-160.

Bishop, C.M. (1995). *Neural Networks for Pattern Recognition*. Oxford, UK: Clarendon Press.

Bounds, D.G.; Lloyd, P.J.; Mathew, B.; and Waddell, G. (1988). "A multilayer perceptron network for the diagnosis of low back pain." *Proc.* of 2nd IEEE Annual Int'l Conf. on Neural Networks, San Diego, June 21-24. Piscataway, NJ: IEEE Press, pp. II.481-II.489.

Box, G.E.P. and Draper, N.R. (1987). *Empirical Model-Building and Response Surfaces*. New York: John Wiley & Sons.

Breiman, L. and Spector, P. (1992). "Submodel selection and evaluation in regression: the X-random case." *Int'l Statistics Review* (v60, n3), pp291-319.

Table 3
The *p*-Values from Hypothesis Tests of Predictions of the Two Competitive Models

	p-value					
	Prediction		Error		Relative Error	
95% C.I.	*X*-Location	*Y*-Location	*X*-Location	*Y*-Location	*X*-Location	*Y*-Location
Mann-Whitney test	0.853	0.505	0.878	0.536	0.902	0.544
Levene's test	0.702	0.841	0.370	0.175	0.370	0.175

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

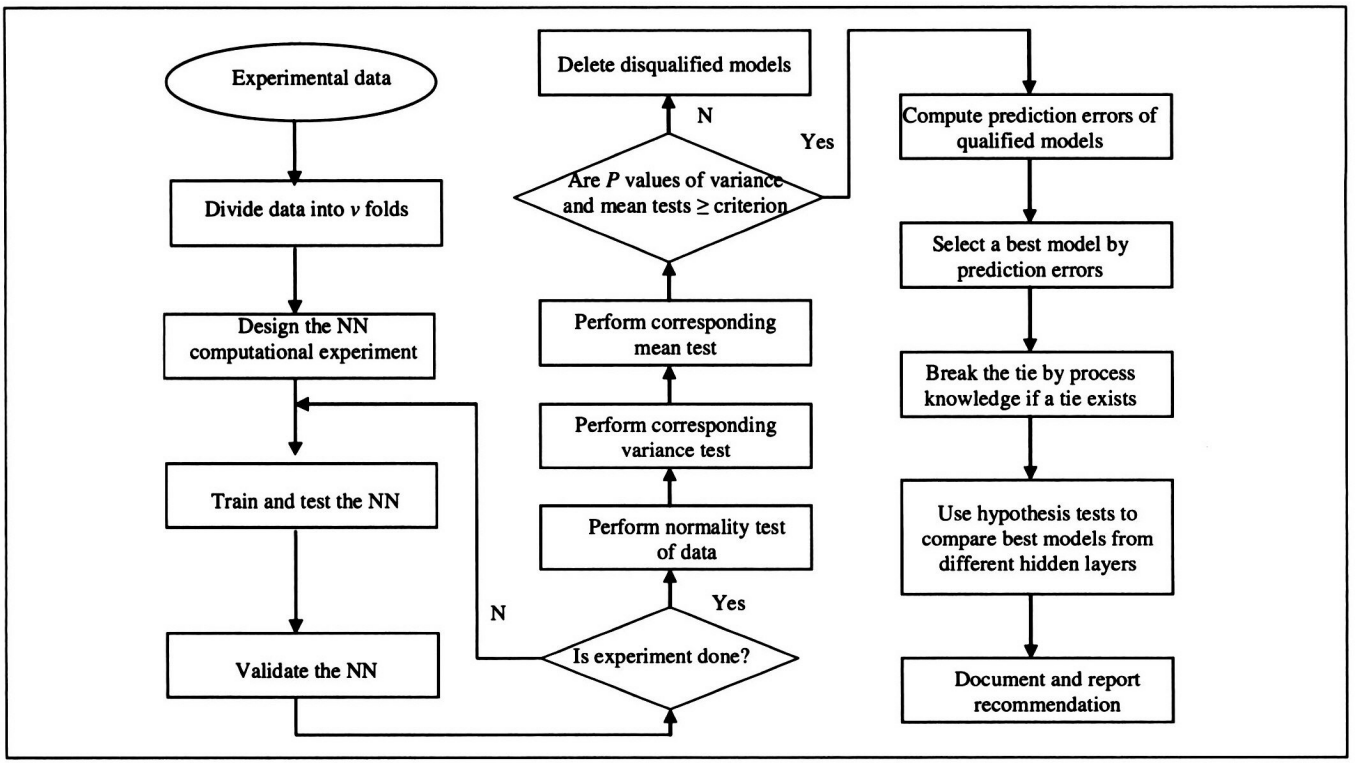

Figure 4
Guideline for Developing BP NN Models of Measurement Uncertainty Data

Burdick, R.K.; Borror, C.M.; and Montgomery, D.C. (2003). "A review of methods for measurement systems capability analysis." *Journal of Quality Technology* (v35, n4), pp342-354.

Burnham, K.P. and Anderson, D.R. (2002). *Model Selection and Inference: A Practical Information-Theoretic Approach*, 2nd ed. New York: Springer.

Chester, D.L. (1990). "Why two hidden layers are better than one?" *Proc.* of 4th IEEE Annual Int'l Conf. on Neural Networks, Washington, DC, Jan. 15-19. Piscataway, NJ: IEEE Press, pp. I.265-I.268.

Coit, D.W.; Jackson, B.T.; and Smith, A.E. (1998). "Static neural network process models: considerations and case studies." *Int'l Journal of Production Research* (v36, n11), pp2953-2967.

Cybenko, G. (1989). "Approximation by superpositions of a sigmoidal function." *Mathematics of Control, Signals and Systems* (v2, n4), pp303-314.

Dotson, C.; Harlow, R.; and Thompson, R.L. (2003). *Fundamentals of Dimensional Metrology*, 4th ed. New York: Thomson.

Fahlman, S.E. and Lebiere, C. (1990). "The cascade correlation learning architecture." *Advances in Neural Information Processing Systems II*, D.S. Tounetzky, ed. San Mateo, CA: Morgan Kaufmann, pp524-532.

Feng, C-X. and Wang, X-F. (2004). "Data mining applied to predictive modeling of the knurling process." *IIE Trans.* (v36, n3), pp253-263.

Feng, C-X. and Wang, X-F. (2003). "Surface roughness predictive modeling: neural networks versus regression." *IIE Trans.* (v35, n1), pp11-27.

Feng, C-X. and Yu, Z. (2003). "Neural networks modeling of turning surface roughness parameters defined by ISO 13565." *Transactions of NAMRI/SME* (v31), pp467-474. Also published as SME Technical Paper MS03-202.

Feng, C-X. and Pandey, V. (2002). "Experimental study of the impact of digitizing parameters on digitizing uncertainty with a CMM." *Int'l Journal of Production Research* (v40, n3), pp683-697.

Feng, C-X. and Wang, X-F. (2002a). "Subset selection in predictive modeling of CMM digitization uncertainty." *Journal of Manufacturing Systems* (v21, n6), pp419-439.

Feng, C-X. and Wang, X-F. (2002b). "Digitizing uncertainty modeling for reverse engineering applications: regression vs. neural networks." *Journal of Intelligent Mfg.* (v13, n3), pp189-199.

Feng, C-X.; Yu, Z.; and Kusiak, A. (2006). "Selection and validation of predictive regression and neural networks models for data from designed experiments." *IIE Trans.* (v38, n1), pp13-23.

Feng, C-X.; Wang, X-F.; and Yu, Z. (2002). "Neural networks modeling of honing surface roughness parameters defined by ISO13565." *Journal of Manufacturing Systems* (v21, n5), pp395-408.

Feng, C-X.; Saal, A.; Sulsbury, J.; Ness, A.; and Lin, G.C.S. (2007). "Design and analysis of experiments in CMM measurement uncertainty study." *Precision Engg.* (v31, n1), pp94-101.

Gershenfeld, N. (1999). *The Nature of Mathematical Modeling*. Cambridge, UK: Cambridge Univ. Press.

Girossi, F. and Poggio, T. (1989). "Representation qualities of neural networks: Kolmogorov's theorem is irrelevant." *Neural Computation* (v1, n4), pp465-469.

Groth, R. (1998). *Data Mining: A Hands on Approach for Business Professionals*. Upper Saddle River, NJ: Prentice-Hall.

Hecht-Nelson, R. (1987). "Kolmogorov's mapping neural network existence theorem." *Proc.* of 1st IEEE Annual Int'l Conf. on Neural Networks, San Diego, June 21-24. Piscataway, NJ: IEEE Press, pp. III.11-III.14.

Hornik, F.M.; Stinchcombe, M.; and White, H. (1989). "Multilayer feed forward networks are universal approximators." *Neural Networks* (v2, n2), pp359-366.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Ingrassia, S. and Morlini, I. (2005). "Neural network modeling of small datasets." *Technometrics* (v47, n3), pp297-311.

Kolen, J.F. and Pollack, J.B. (1990). "Backpropagation is sensitive to initial conditions." *Complex Systems* (v4, n3), pp269-280.

Kůrková, A. (1991). "Kolmogorov's theorem is relevant." *Neural Computation* (v3, n4), pp617-622.

Kusiak, A. (2000). *Computational Intelligence in Design and Manufacturing*. New York: John Wiley & Sons.

Lang, K.J. and Witbrock, M.J. (1988). "Learning to tell two spirals apart." *Proc.* of 1988 Connectionist Models Summer School. San Mateo, CA: Morgan Kaufmann, pp52-59.

Lawrence, J. (1994). *Introduction to Neural Networks: Design, Theory, and Applications*, 6th ed. Nevada City, CA: California Scientific Software.

Lawrence, J. and Fredrickson, J. (1998). *BrainMaker User's Guide and Reference Manual*, 7th ed. Nevada City, CA: California Scientific Software.

Le Cun, Y.; Boser, B.; Denker, J.S.; Henderson, D.; Howard, R.E.; Hubbard, W.; and Jackel, L.D. (1989). "Backpropagation applied to handwritten ZIP code recognition." *Neural Computation* (v1, n5), pp541-551.

Liao, W.; Wang, G.; Triantaphyllou, E.; and Chang, P.-C. (2001). "A data mining study of weld quality models constructed with MLP neural networks from stratified sampled data." *Proc.* of 2001 Industrial Engg. Research Conf., IIE.

Lippmann, R.P. (1987). "An introduction to computing with neural nets." *IEEE Acoustics, Speech and Signal Processing Magazine* (v3, n4, April), pp4-22.

Marchandani, G. and Cao, W. (1989). "On hidden nodes for neural nets." *IEEE Trans. on Circuits and Systems* (v36, n5), pp661-664.

McCullagh, P. and Nelder, J.A. (1989). *Generalized Linear Models*, 2nd ed. London: Chapman & Hall.

Mitchell, T.M. (1997). *Machine Learning*. New York: McGraw-Hill.

Montgomery, D.C. (2005). *Introduction to Statistical Quality Control*, 5th ed. New York: John Wiley & Sons.

Montgomery, D.C.; Peck, E.A.; and Vining, C.G. (2001). *Introduction to Linear Regression Analysis*, 3rd ed. New York: John Wiley & Sons.

Narayanan, Geeta (2003). "Gauge repeatability and reproducibility study: a status review and case studies." MS Project Report (Advisor: Dr. Jack Feng). Peoria, IL: Dept. of Industrial & Mfg. Engg. & Technology, Bradley Univ.

Saal, Tony (2003). "Applying regression analysis to a CMM measurement cncertainty study." MS Project Report (Advisor: Dr. Jack Feng). Peoria, IL: Dept. of Industrial & Mfg. Engg. & Technology, Bradley Univ.

Smith, A.E. and Mason, A.K. (1997). "Cost predictive estimation: regression versus neural networks." *The Engineering Economist* (v42, n2), pp137-161.

Swingler, K. (1996). *Applying Neural Networks: A Practical Guide*. San Francisco: Morgan Kaufmann Publishers.

Sontag, E.D. (1992). "Feedback stabilization using two-hidden-layer nets." *IEEE Trans. on Neural Networks* (v3, n6), pp981-990.

Twomey, J.M. and Smith, A.E. (1998). "Bias and variance of validation methods for function approximation neural networks under conditions of sparse data." *IEEE Trans. on Systems, Man, and Cybernetics, Part C* (v28, n3), pp417-430.

Yu, Z-G. and Feng, C-X. (2004). "Threefold vs. fivefold cross-validation in predictive neural networks modeling of experimental data from a turning surface roughness study." *Proc.* of 6th Int'l Conf. on Frontiers in Design and Mfg., Xi'an, China, June 18-20. Jointly published by International Academic Publishers and World Publishing Corp., Beijing, China, pp322-329.

Wasserman, P.D. (1989). *Neural Computing: Theory and Practice*. New York: Van Nostrand Reinhold.

Williams, N.R. and Feng, C-X. (2005). "Neural networks vs. regression in localized scan correction for robotic drilling applications." *Journal of Manufacturing Systems* (v24, n1), pp21-34.

Witten, I.H. and Frank, E. (2001). *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. San Francisco: Morgan Kaufmann Publishers.

Zhang, P. (1993). "Model selection via multifold cross-validation." *Annals of Statistics* (v21, n1), pp299-311.

Authors' Biographies

Chang-Xue Jack Feng is a professor of industrial and manufacturing engineering at Bradley University. He obtained his PhD and MS degrees in industrial and management engineering, MS degree in manufacturing engineering, and BS degree in mechanical engineering. He was a visiting engineering fellow of Caterpillar Production System at Caterpillar Inc. between May 2006 and August 2007. Dr. Feng's many technical publications have appeared in journals such as the *Annals of Operations Research, ASME Transactions, IEEE Transactions, IIE Transactions, International Journal of Production Research, Journal of Manufacturing Systems*, and *Precision Engineering*. He also has coauthored three books and three book chapters. Dr. Feng has conducted funded projects or consulting for Bradley University, Boeing, Caterpillar Inc., Children's Hospital of Illinois, China NSF, Cincinnati Machine, Dana Corporation, John Deere, Penn State, Rockwell International, Rubbermaid, SME, Tyco Health, the University of Illinois, and the University of Iowa. He is a senior member of ASQ, IIE, and SME. He is also a member of the American Statistical Association (ASA) and INFORMS.

Abhirami Gowrisankar is currently assistant production manager at Best Engineers Pumps Pvt. Ltd. in Coimbatore, India. She obtained her MS degree in manufacturing engineering from Bradley University in December 2003 and her BE degree in production engineering from Amrita Institute of Technology and Science affiliated to Bharathiyar University, Coimbatore, India, in April 2002. She is a certified lead auditor for ISO9001: 2000.

Zhi-Guang (Samuel) Yu is a senior project manager at Supply Chain Services International in Peoria, IL. He obtained his MS degree in manufacturing engineering from Bradley University and a BS in material science and engineering from Tsinghua University, Beijing, China. He has coauthored several papers published in *IIE Transactions, Journal of Manufacturing Systems, International Journal of Production Research*, and *Transactions of NAMRI/SME*.

Alice E. Smith is professor and chair of the Industrial and Systems Engineering Dept. at Auburn University. Previously she was on the faculty of the Dept. of Industrial Engineering at the University of Pittsburgh, which she joined in 1991 after 10 years of industrial experience with Southwestern Bell Corporation. Dr. Smith has degrees in engineering and business from Rice University, Saint Louis University, and the University of Missouri-Rolla. Her research in analysis, modeling, and optimization of manufacturing processes and engineering design has been funded by NASA, NIST, Lockheed Martin, Adtranz (now Bombardier Transportation), the Ben Franklin Technology Center of Western Pennsylvania, and NSF. She was awarded a CAREER grant in 1995 and an ADVANCE Leadership grant in 2001 from NSF. Her industrial partners on sponsored research projects have included DaimlerChrysler Electronics, Eljer Plumbingware, Extrude Hone, Ford Motor, PPG Industries, and Crucible Compaction Metals. Dr. Smith holds one U.S. patent and several international patents and has authored more than 60 journal publications. She is a fellow of IIE, a senior member of IEEE and SWE, a member of Tau Beta Pi, INFORMS and ASEE, and a registered professional engineer in industrial engineering in Alabama and Pennsylvania.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.