Is a Complex Neural Network Based Air Quality Prediction Model Better Than a Simple One? A Bayesian Point of View

K. I. Hoi, K. V. Yuen, and K. M. Mokkmmok@umac.mo

Citation: AIP Conference Proceedings **1233**, 764 (2010); doi: 10.1063/1.3452273

View online: http://dx.doi.org/10.1063/1.3452273

View Table of Contents: http://aip.scitation.org/toc/apc/1233/1

Published by the American Institute of Physics

Is a Complex Neural Network Based Air Quality Prediction Model Better Than a Simple One? A Bayesian Point of View

K.I. Hoi[a], K.V. Yuen[a], and K.M. Mok[a, *]

[a]*Department of Civil and Environmental Engineering, University of Macau (*KMMOK@umac.mo)*

Abstract. In this study the neural network based air quality prediction model was tested in a typical coastal city, Macau, with Latitude 22°10'N and Longitude 113°34'E. By using five years of air quality and meteorological data recorded at an ambient air quality monitoring station between 2001 and 2005, it was found that the performance of the ANN model was generally improved by increasing the number of hidden neurons in the training phase. However, the performance of the ANN model was not sensitive to the change in the number of hidden neurons during the prediction phase. Therefore, the improvement in the error statistics for a complex ANN model in the training phase may be only caused by the overfitting of the data. In addition, the posterior PDF of the parameter vector conditional on the training dataset was investigated for different number of hidden neurons. It was found that the parametric space for a simple ANN model was globally identifiable and the Levenberg-Marquardt backpropagation algorithm was able to locate the optimal parameter vector. However, the parameter vector might contain redundant parameters and the parametric space was not globally identifiable when the model class became complex. In addition, the Levenberg-Marquardt backpropagation algorithm was unable to locate the most optimal parameter vector in this situation. Finally, it was concluded that the a more complex MLP model, that fits the data better, is not necessarily better than a simple one.

Keywords: Artificial neural network, Air quality prediction, Bayesian approach, Macau, PM_{10}.
PACS: 92.60.Sz

INTRODUCTION

The artificial neural network (ANN) is widely used in the disciplines of environmental science and engineering in the recent decades. Among different architectures of artificial neural network, the multilayer perceptrons (MLP) architecture is used as a tool for function approximation in this area. Examples of application include air quality prediction, estimation of nonlinear relationships between atmospheric aerosol and its gaseous precursors, modeling of river water quality, tidal forecasting, etc [1-5]. The MLP is popular since it could approximate any smooth, measurable function between the input and output vectors by a suitable combination of network parameters [6]. However, the choice in the number of hidden neurons, which represents the complexity of the model class, is still an open question. Therefore, it is aimed to investigate the performance of the ANN model during model training and validation by altering the number of hidden neurons. In addition, the trained network weights and biases are evaluated by the Bayesian approach. The Bayesian approach is a probabilistic approach which allows one to obtain the most probable estimates of some unknown parameters of a system and to quantify the associated uncertainties based on given data [7].

In order to accomplish the aforementioned objectives, the MLP was applied in this study as a statistical air quality prediction model. The model performs the one-step-ahead prediction of the daily averaged PM_{10} concentration by using the concentration of yesterday and the selected meteorological parameters on the day of prediction as the inputs to the model. A case study was also provided as a reference. In the following section, the formulation of the neural network based air quality prediction model is briefly described.

CP1233 *The 2nd International ISCM Symposium and The 12th International EPMESC Conference*
J. W. Z. Lu, A.Y.T. Leung, V. P. Iu, and K. M. Mok

NEURAL NETWORK BASED AIR QUALITY PREDICTION MODEL

In this study the air quality prediction model is the multilayer perceptrons (MLP). Figure 1 shows the architecture of the multilayer perceptrons. It consists of the input layer, the hidden layer with the hyperbolic tangent sigmoid transfer function, the output layer with the linear transfer function, and the output variable. The input variables are selected based on the nature of a typical coastal city, Macau. However, it is believed that only slight modification of input variables is necessary to make the MLP model become applicable to other coastal cities since they are also influenced by similar physical processes. The input layer consists of five input variables denoted by x_{k-1}, x'_{k-1}, u_k, $|\theta_k|$ and r_k. In order to distinguish the normalized variables from the original variables, a bar is put on top of a variable to signify that it is normalized. The symbols x_{k-1} and x'_{k-1} represent the daily averaged PM_{10} concentration of the previous day and the hourly averaged PM_{10} concentration before midnight, respectively. These two variables are used to reflect the initial condition of PM_{10} concentration on the next day. The symbols u_k and $|\theta_k|$ denote the magnitude and the absolute angle of the resultant wind velocity vector. The resultant wind velocity vector is obtained by the vector sum of the hourly wind velocities on the day of prediction.

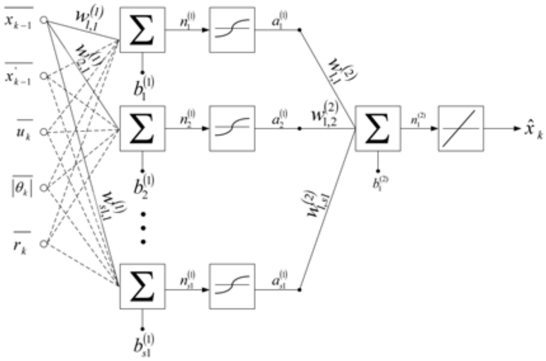


FIGURE 1. Architecture of Multilayer Perceptrons

These two variables are used to reflect the dispersion condition and the type of replenishing air masses being transported to Macau on the day of prediction. The symbol r_k denotes the rainfall index, which is defined as the product of the daily rainfall amount and the duration of rainfall on the k^{th} day. It is used as a discounting factor on the PM_{10} concentrations whenever there is rainfall on the day of prediction. The symbol $\hat{x}_k$ denotes the output variable, which is the one-step-ahead prediction of the daily averaged PM_{10} concentration. The MLP is trained by the Levenberg-Marquardt backpropagation algorithm. Further explanation of the neural network based air quality prediction model can be referred to [8].

BAYESIAN APPROACH

As mentioned above, the trained weights and biases of the prediction model will be evaluated by the Bayesian approach. The Bayesian approach is a probabilistic approach which allows one to obtain the most probable estimates of the uncertain parameters of a system and to quantify the associated uncertainties based on given data [7]. The idea of applying the Bayesian approach is simple. First we define the parameter vector $\boldsymbol{\theta}$ which contains the network weights and biases of the MLP model:

$$\boldsymbol{\theta} = \left[w^{(1)}_{1,1}, \cdots, w^{(1)}_{1,5}, \cdots, w^{(1)}_{s1,1}, \cdots, w^{(1)}_{s1,5}, b^{(1)}_1, \cdots, b^{(1)}_{s1}, w^{(2)}_{1,1}, \cdots, w^{(2)}_{1,s1}, b^{(2)}_1\right]^T . \tag{1}$$

The superscript represents the number of layer. The hidden layer is denoted as layer 1 and the output layer is denoted as layer 2. In addition, the symbols $s1$, $w_{i,j}$, and b_i denote the number of neurons in the hidden layer, the weight assigned to the i^{th} neuron from the j^{th} input, and the bias assigned to the i^{th} neuron, respectively. For a given dataset $\mathbf{D}=\{z_1,..,z_N\}$ which contains N days of measured PM_{10} concentrations, the posterior probability density function $p(\boldsymbol{\theta}|\mathbf{D})$ of the parameter vector conditional on the dataset $\mathbf{D}$ can be estimated by using the Bayes theorem, which reverses the conditioning of the events as shown below:

$$p(\boldsymbol{\theta} \mid \mathbf{D}) = p(\mathbf{D} \mid \boldsymbol{\theta}) \frac{p(\boldsymbol{\theta})}{p(\mathbf{D})}. \tag{2}$$

The term $p(\mathbf{D}|\theta)$ denotes the likelihood factor, which represents how well the model output approximates the data for a given parameter vector $\boldsymbol{\theta}$. The term $p(\boldsymbol{\theta})$ denotes the prior PDF of the parameter vector. It is an arbitrary PDF specified by the user and it reflects the prior knowledge of the user on the uncertain parameters. In this study, a uniform prior is chosen. The term $p(\mathbf{D})$ is a normalizing constant such that the integration on the right hand side over the parametric space is equal to 1. Since a uniform prior is chosen, the prior PDF can be directly absorbed by the normalizing constant and the posterior PDF can be directly estimated from the likelihood factor which can be expressed as the product of the PDF of the measurement z_k conditional on the parameter vector θ:

$$p(\mathbf{D} \mid \boldsymbol{\theta}). = \prod_{k=1}^{N} p(z_k \mid \boldsymbol{\theta}). \tag{3}$$

The measured PM_{10} concentration of the k^{th} day z_k is related to the model output $\hat{x}_k$ according to the following relationship:

$$z_k = \hat{x}_k + n_k. \tag{4}$$

where n_k is the prediction error. The prediction error is modeled as Gaussian i.i.d. with zero mean and variance σ_n^2. The measurement z_k conditional on the parameter vector θ is normally distributed as follows:

$$z_k \sim N\left(\hat{x}_k, \sigma_n^2\right). \tag{5}$$

and the corresponding PDF has the following form:

$$p(z_k \mid \boldsymbol{\theta}) = \frac{1}{\sqrt{2\pi}\sigma_n} . \exp\left(-\frac{(z_k - \hat{x}_k)^2}{2\sigma_n^2}\right). \tag{6}$$

Therefore, the likelihood factor $p(\mathbf{D}|\theta)$ is given by

$$p(\mathbf{D} \mid \boldsymbol{\theta}) = \frac{1}{(2\pi)^{N/2} \sigma_n^N} . \exp\left(-\frac{N}{2\sigma_n^2} J(\boldsymbol{\theta})\right). \tag{7}$$

where $J(\theta)$ is the mean squared prediction error of the MLP model and the variance of the prediction error is equal to the value of $J(\theta)$ evaluated at the optimal parameter vector. The posterior PDF $p(\theta|\mathbf{D})$ of the parameter vector θ conditional on the dataset $\mathbf{D}$ can be displayed by plotting some projections of its cross-section cutting along a specified principal axis. This is done by (i) generating samples around the optimal parameter vector along the specified principal axis, (ii) evaluating the corresponding values of $p(\theta|\mathbf{D})$ from Eqn. (7), and (iii) finally plotting the posterior PDF. Samples of parameter vector are generated along the principal axis in order to reduce the generation of samples over the region with little probability density. As the posterior PDF $p(\theta|\mathbf{D})$ is multivariate Gaussian, its covariance matrix is equal to the inverse of $\mathbf{H}(\theta)$, which is the Hessian matrix of $-\ln[p(\theta|\mathbf{D})]$ with respect to the parameter vector $\boldsymbol{\theta}$. Therefore, the principal axes of the posterior PDF are found by finding the eigenvectors of $\mathbf{H}(\theta)$. For a uniform prior PDF, the Hessian matrix $\mathbf{H}(\theta)$ is simplified as follows:

$$\mathbf{H}(\boldsymbol{\theta}) \equiv \frac{\partial^2 \left[-\ln p(\boldsymbol{\theta} \mid \mathbf{D})\right]}{\partial \boldsymbol{\theta}^2} = \frac{N}{2\sigma_n^2} \left[\frac{\partial^2 J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^2}\right]. \tag{8}$$

The Hessian matrix of $J(\theta)$ with respect to the parameter vector θ can be derived by using the idea of backpropagation and its expression is shown below:

$$\frac{\partial^2 J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^2} = \frac{1}{N} \sum_{k=1}^{N} \frac{\partial^2 J_k(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^2}. \tag{9}$$

where $J_k(\theta)$ denotes the squared prediction error of the k^{th} day for the MLP model and its Hessian matrix with respect to the parameter vector θ is given by:

$$\frac{\partial^2 J_k(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^2} = \begin{bmatrix} \mathbf{S}_{11} \otimes \left(\mathbf{p}_k \mathbf{p}_k^T\right) & \mathbf{S}_{11} \otimes \mathbf{p}_k & \mathbf{S}_{12} \otimes \left[\mathbf{p}_k \left(\mathbf{a}_k^{(1)}\right)^T\right] - 2\left(z_k - \hat{x}_k\right)\mathbf{F}_k \otimes \mathbf{p}_k & \mathbf{S}_{12} \otimes \mathbf{p}_k \\ \vdots & \mathbf{S}_{11} & \mathbf{S}_{12}\left(\mathbf{a}_k^{(1)}\right)^T - 2\left(z_k - \hat{x}_k\right)\mathbf{F}_k & \mathbf{S}_{12} \\ \vdots & \ddots & 2\left(\mathbf{a}_k^{(1)}\right)\left(\mathbf{a}_k^{(1)}\right)^T & 2\mathbf{a}_k^{(1)} \\ \vdots & \cdots & \cdots & 2 \end{bmatrix}. \tag{10}$$

The symbol ⊗ denotes the kronecker product of two matrices. The symbol **p** represent the input vector which contains the normalized input variables and the symbol $\mathbf{a}^{(1)}$ denotes the output vector of the hidden layer. The symbol $\mathbf{S}_{11}$ denote the Hessian matrix of the squared prediction error of the k^{th} day with respect to the net input vector to the hidden layer $\mathbf{n}^{(1)}$ evaluated on the k^{th} day and the entries of $\mathbf{S}_{11}$ are computed by:

$$\left[\mathbf{S}_{11}\right]_{i,j} = 2w_{1,i}^{(2)} w_{1,j}^{(2)} \left[1 - \left(a_{i,k}^{(1)}\right)^2\right]\left[1 - \left(a_{j,k}^{(1)}\right)^2\right] + 4w_{1,i}^{(2)}\left(z_k - \hat{x}_k\right) a_{i,k}^{(1)} \left[1 - \left(a_{i,k}^{(1)}\right)^2\right]\delta_{i,j}, \quad i, j = 1, \ldots, s1. \tag{11}$$

The symbol $\delta_{i,j}$ denotes the kronecker delta such that:

$$\delta_{i,j} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}. \tag{12}$$

The symbol $\mathbf{S}_{12}$ denote a $s1 \times 1$ column vector which contains the second order derivative of $J_k(\boldsymbol{\theta})$ with respect to the net input vector to the hidden layer $\mathbf{n}^{(1)}$ and the net input to the output layer $n_1^{(2)}$, evaluated on the k^{th} day:

$$\left[\mathbf{S}_{12}\right]_i = 2w_{1,i}^{(2)} \left[1 - \left(a_{i,k}^{(1)}\right)^2\right], \quad i = 1, \ldots, s1. \tag{13}$$

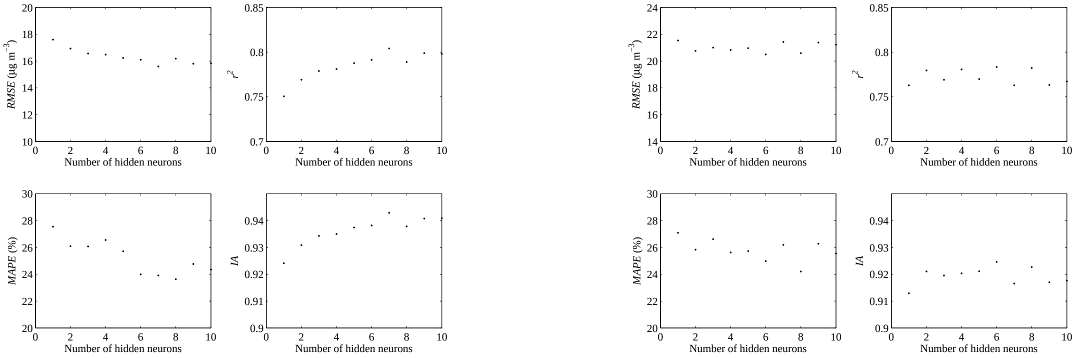
Finally, the matrix $\mathbf{F}_k$ is a diagonal matrix. Its diagonal entries contain the first order derivative of the transfer function in the hidden layer with respect to the net input vector to this layer:

$$\mathbf{F}_k = diag\left(1 - \left(a_{1,k}^{(1)}\right)^2, \ldots, 1 - \left(a_{s1,k}^{(1)}\right)^2\right). \tag{14}$$

CASE STUDY

In this section the neural network based air quality prediction model is tested in a typical coastal city, Macau, with Latitude 22°10'N and Longitude 113°34'E. The data consists of daily averaged PM_{10} concentrations and the meteorological conditions including the wind speed, the wind direction, and the amount of precipitation recorded at an ambient air quality monitoring station between 2001 and 2005. The station has an altitude of 158.2 m, hence its air quality and meteorological measurements are considered to be representative of the general background conditions for the whole city. The data between 2001 and 2002 are used for model training, while the data of the following three years are used for model validation. In order to investigate the effect of the number of hidden neurons on the model performance, a parametric study was carried out by training the ANN model with different number of hidden neurons from 1 to 10. To evaluate the performance of each model class, the root-mean-square error (*RMSE*), the mean absolute percentage error (*MAPE*), the coefficient of determination (r^2) and the index of agreement (*IA*) are adopted [1,9]. In general, good predictive models associate with small values of *RMSE* and *MAPE*, as well as large values of r^2 and *IA*.

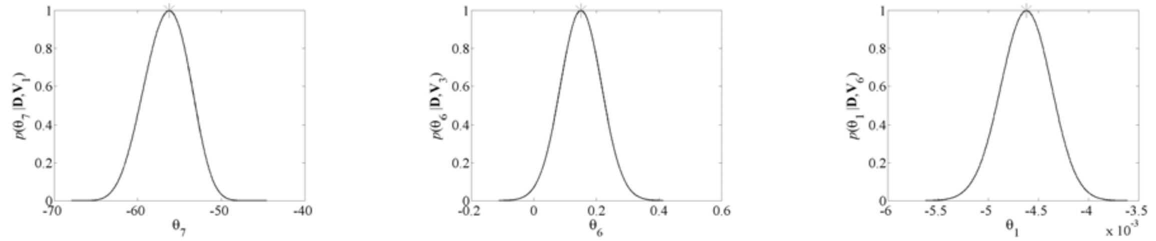
Figure 2(a) shows the variation of the error statistics of the ANN model with respect to the number of hidden neurons in the training phase. It is noted that the performance of the ANN model is generally improved when the number of hidden neurons is increased. Figure 2(b) shows the variation of the error statistics with respect to the number of hidden neurons in the prediction phase. It is surprising to see that the performance of the ANN model in the prediction phase is not sensitive to the change in the number of hidden neurons. In the training phase, the weights and biases of the ANN model are adjusted in order to minimize the error between the prediction and the measurement. Therefore, a complex ANN model tends to have better model performance in the training phase since it has more uncertain parameters than a simple one. In other words, a complex ANN model has more capability to fit the data. However, the measurement is a superposition of the actual signal and the measurement noise. The improvement in the error statistics for a complex ANN model in the training phase may be due to the overfitting of the data. This also explains why the model performance is not sensitive to the change in the number of hidden neurons in the prediction phase.



(a) (b)

FIGURE 2. Error Statistics of Prediction Model: (a) Training Phase, (b) Prediction Phase

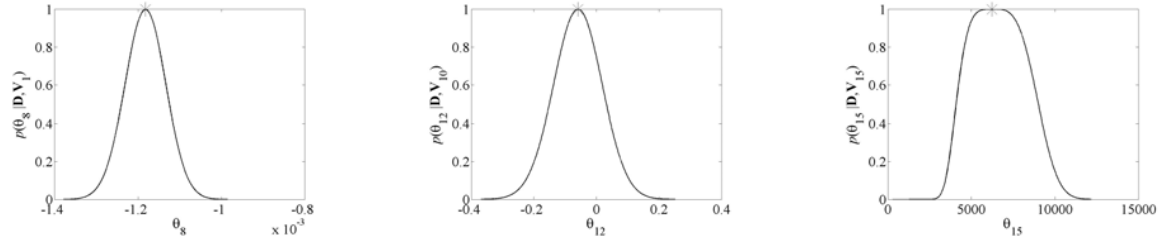
Figure 3(a) shows the projection on the θ_7 direction for the cross-section of the posterior PDF $p(\boldsymbol{\theta}|\mathbf{D})$ of the parameter vector $\boldsymbol{\theta}$ conditional on the training dataset $\mathbf{D}$ (2001-2002). The cross-section is produced by cutting the posterior PDF along the principal axis specified by the eigenvector $\mathbf{V}_1$. The number of hidden neurons in this ANN model is 1. The symbol * shows the location of the trained parameter vector on the projection axis and the corresponding probability density. It is noted that the shape of the projection is a unimodal distribution which resembles the shape of the Gaussian PDF and the trained parameter vector is located at the point with highest probability density. Similar situations are observed for the projections cutting along other principal axes as shown in Figure 3(b) and Figure 3(c). This implies that the parametric space for this ANN model is globally identifiable, i.e., there is a unique optimal parameter vector and the Levenberg- Marquardt backpropagation algorithm is able to locate the optimal parameter vector.



(a) (b) (c)

FIGURE 3. Projections for the cross-sections of $p(\theta|D)$ cutting along different principal axes for $s1=1$

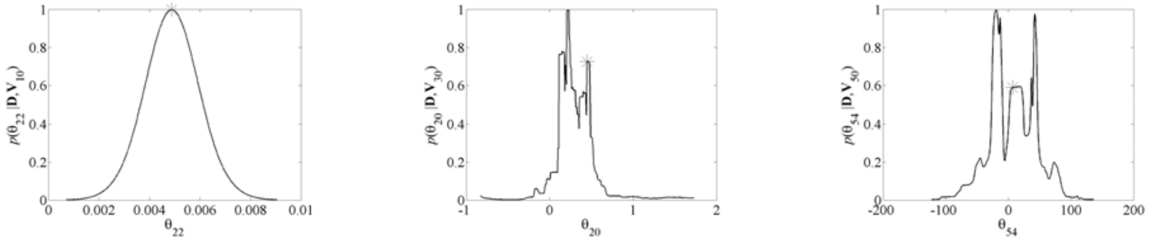
Figures 4(a)-4(c) show the projections for the cross-sections of the posterior PDF $p(\boldsymbol{\theta}|\mathbf{D})$ cutting along the principal axes specified by the eigenvectors $\mathbf{V}_1$, $\mathbf{V}_{10}$ and $\mathbf{V}_{15}$, respectively. The number of hidden neurons in this ANN model is 2. It is interesting to note that the shape of the projection shown in Figure 4(c) is different from those shown in Figure 4(a)-4(b). The most obvious difference is that the projection in Figure 4(c) does not have a single peak. Instead, it has a wide plateau with probability density equal to 1. This means that the parametric space is not globally identifiable. In addition, the actual posterior PDF may resemble a twisted manifold. Therefore, the parameter vector may contain a redundant parameter.



(a) (b) (c)

FIGURE 4. Projections for the cross-sections of $p(\boldsymbol{\theta}|\mathbf{D})$ cutting along different principal axes for $s1=2$

Figures 5(a)-5(c) show the projections for the cross-sections of the posterior PDF $p(\boldsymbol{\theta}|\mathbf{D})$ cutting along the principal axes specified by the eigenvectors $\mathbf{V}_{10}$, $\mathbf{V}_{30}$ and $\mathbf{V}_{50}$, respectively. The number of hidden neurons in this ANN model is 10. It is noted that only the shape of the projection shown in Figure 5(a) is a unimodal distribution which resembles the shape of the Gaussian PDF and the trained parameter vector is located at the point with highest probability density. However, the projections shown in Figure 5(b) and Figure 5(c) are multi-modal and the trained parameter vector is not located at the point with highest probability density. This implies that the parametric space associated with this ANN model is not globally identifiable and the Levenberg-Marquardt backpropagation algorithm is not able to locate the most optimal parameter vector.



(a) (b) (c)

FIGURE 5. Projections for the cross-sections of $p(\boldsymbol{\theta}|\mathbf{D})$ cutting along different principal axes for $s1=10$

CONCLUSION

In this study the neural network based air quality prediction model was tested in a typical coastal city, Macau, with Latitude 22°10'N and Longitude 113°34'E. By using five years of air quality and meteorological data recorded at an ambient air quality monitoring station between 2001 and 2005, it was found that the performance of the ANN model was generally improved by increasing the number of hidden neurons in the training phase. However, the performance of the ANN model was not sensitive to the change in the number of hidden neurons during the prediction phase. Therefore, the improvement in the error statistics for a complex ANN model in the training phase may be only caused by the overfitting of the data.

By using the Bayesian approach, the posterior PDF of the parameter vector conditional on the training dataset was investigated for different number of hidden neurons. It was found that the parametric space for a simple ANN model was globally identifiable and the Levenberg-Marquardt backpropagation algorithm was able to locate the optimal parameter vector. However, the parameter vector might contain redundant parameters and the parametric space was not globally identifiable when the model class became complex. In addition, the Levenberg-Marquardt backpropagation algorithm was unable to locate the most optimal parameter vector in this situation. Finally, it was concluded that the a more complex MLP model, that fits the data better, is not necessarily better than a simple one.

ACKNOWLEDGMENTS

The contents presented in this paper was (partially) supported by the Research Committee of University of Macau under grant RG-UL/07-08S/Y0/MKM/FST. The SMG of Macau is thanked for supplying the data.

REFERENCES

1. K. M. Mok and S. C. Tam, *Energy and Buildings* **28**, 279-286 (1998).
2. K. I. Hoi, K. V. Yuen and K. M. Mok, *Atmospheric Environment* **43**, 2579-2581 (2009).
3. I. B. Konovalov, *Atmospheric Chemistry and Physics* **3**, 607-621 (2003).
4. K. P. Singh, A. Basant, A. Malik and G. Jain, *Ecological Modelling* **220**, 888-895 (2009).
5. D. T. Cox, P. Tissot and P. Michaud, *Journal of Waterway, Port, Coastal and Ocean Engineering* **128**, 21-29 (2002).
6. K. Hornik, M. Stinchcombe and H. White, *Neural Network* **2**, 359-366 (1989).
7. J. L. Beck and L. S. Katafygiotis, *Journal of Engineering Mechanics* **124**, 455-461 (1998).
8. K. I. Hoi, K. V. Yuen and K. M. Mok, "An Artificial Neural Network Model for the Prediction of Daily Averaged PM_{10} Concentrations in Macau" in 6^{th} *National Civil Engineering Forum for Graduate Students*, edited by Y. J. Shi and P. Feng, Conference Proceedings, Tsinghua University Press, Beijing, China, 2008, 6pp.

9 G. Nunnari, S. Dorling, U. Schlink, G. Cawley, R. Foxall and T. Chatterton, *Environmental Modelling & Software* **19**, 887-905 (2004).