Applied Soft Computing 11 (2011) 3357–3367

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

An inverse parameter identification procedure assessing the quality of the estimates using Bayesian neural networks

Jörg F. Unger*, Carsten Könke

Institute of Structural Mechanics, Bauhaus-University Weimar, Marienstr. 15, D-99423 Weimar, Germany

ARTICLE INFO

Article history:
Received 6 March 2009
Received in revised form 21 June 2010
Accepted 3 January 2011
Available online 14 January 2011

Keywords:
Parameter identification
Model updating
Bayesian neural network
Meta model
Regularization
Design of experiments

ABSTRACT

The paper proposes a general procedure based on Bayesian neural networks for parameter identification of numerical models. In this context, the Bayesian neural networks are extended to multiple outputs with a full covariance matrix to describe the correlation between the noise of output parameters. This extension is especially useful for inverse problems such as a parameter identification procedure, since it allows for the quantification of correlations between output parameters. Based on numerically obtained forward calculations, the Bayesian neural network is trained to solve the inverse parameter identification problem. The main advantage of the method is the ability to verify the accuracy of the identified parameters and their correlation. The methodology further allows to detect, whether a certain set of experiments is sufficient to determine an individual model parameter. As a result, a general scheme for the design of experiments to identify model parameters is developed and illustrated for two examples.

1. Introduction

Complex models are often used to accurately simulate real life problems. This requires the identification of the model parameters which are sometimes not physically interpretable and, consequently, a direct evaluation from experimental data is not straightforward. Furthermore, if parameters are correlated or have a similar influence, they are difficult to identify simultaneously. In standard parameter identification procedures, a certain error measure between the numerical and experimental data set is defined, which might be, e.g. load–displacement curves, eigenvalues or eigenfrequencies. This error is then minimized by modification of the material parameters in the numerical simulation using different optimization procedures such as gradient based methods [1–5] or genetic algorithms [6,7]. Finally, a set of parameters for the numerical model is obtained that seems to best simulate the experimental data. A general overview of different methods for parameter identification is given in [8]. Monitoring problems are another field of application [9–13]. In this context, the parameter identification procedure is used to identify the actual state of the structure with respect to a reference state, which is often based on modal data.

The idea to apply neural networks to parameter identification problems date back to [14], where the nonlinear outputs of dynamic systems are approximated by a black box model. Ref. [15] used neural networks to detect damage within a structure. In [16,17], the idea to create a metamodel based on a set of forward calculations (from model parameters to the system response) that models the inverse relation was introduced. In this context, radial basis function neural networks were used. Ref. [18] highlighted the problem of the bias/variance dilemma and proposed strategies to build up an optimal architecture and perform an efficient training procedure in order to obtain good generalization properties. A discussion on the proper choice for the generation of the training data is given in [19] and the advantages of orthogonal arrays compared to standard procedures, like full factorial or stochastic design of experiments, are highlighted. A model updating procedure using neural networks is also used in [20], where based on modal parameters the system parameters are determined. In order to increase the accuracy, a two stage procedure decoupling the determination of the damping coefficients from the remaining parameters is introduced. An advantage of parameter identification based on neural networks is their ability to solve the problem even in the presence of noisy data. Ref. [21] tries to increase the accuracy of the parameter estimates using the parameters from the neural network as the starting point of a subsequent gradient iteration.

In general, most parameter identification procedures including all the procedures based on standard neural networks have

* Corresponding author at: Department of Mechanical Engineering, Northwestern University, 2145 Sheridan Road, Evanston, IL, United States.
E-mail addresses: j-unger@northwestern.edu (J.F. Unger), carsten.koenke@uni-weimar.de (C. Könke).

1568-4946/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.asoc.2011.01.007

the severe disadvantage that no information about the accuracy of the estimate is given and the complexity of the metamodel has to be determined a priori. If a certain parameter has no influence on the given test data, or two parameters influence the response in the same way (redundant parameters for the given experimental data), the experiment should be modified in order to determine these parameters separately. A simple example is a material model described by a tensile and a compressive strength. Obviously, an experimental tensile test will only allow the identification of the tensile strength, and an additional compression test is required. For complex material models, the interconnection between material parameter and corresponding test required to identify that parameter is not straightforward, and often a trial and error procedure enhanced by engineering knowledge is performed. In this context, the proposed methodology offers the possibility to estimate the parameter and its precision, which allows for a set of experiments to be designed such that they are sufficient to estimate all parameters of the model.

Another disadvantage of most parameter identification procedures is the required number of forward calculations, which is especially difficult for complex and computational expensive numerical models. Genetic algorithms and their variations need, in general, a population of several hundred individuals with a couple of recombination steps which finally results in more than 10,000 forward calculations. For gradient based methods, the gradient has to be calculated either analytically, which is often difficult to realize, or via finite differences. For general nonlinear problems, a line search with e.g. 20 steps or a Hessian matrix is required. As a result, gradient based algorithms also require a non negligible number of simulations with the severe disadvantage that the solution may strongly depend on the starting point of the iteration.

The parameter identification procedure proposed in this paper requires only few samples and additionally gives information related to the accuracy of the parameter estimates. It is based on a multilayer perceptron with a single hidden layer (e.g. [22,23]). A serious problem is the generalization capacity of a network which is related to the number of free parameters/neurons. Standard multilayer perceptron often show overfitting phenomena, i.e. they accurately represent training data, but new inputs, which are not part of the training set, are poorly approximated. In order to resolve this problem, several regularization procedures have been developed. In general, they can be subdivided into two classes. In the first class of methods, the dimension of the parameter space is adapted to the problem. Examples of these methods are network pruning [24–26], where weights are incrementally removed from the network, or the application of cascade-correlation network architectures [27], where, starting with a simple network, hidden units are incrementally added. In the second class of methods, the size of the parameters is reduced. Examples for this approach are regularization techniques such as weight decay [28] or early stopping [29].

The Bayesian approach to neural network training was introduced by Mackay [30–32]. It naturally comprises a regularization procedure and it furthermore allows a quantification of the approximation accuracy.

The paper is divided into four parts. At first, the Bayesian approach is summarized and the extension to multiple outputs with a covariance matrix describing the correlations between the noise terms of different outputs is introduced. Afterwards, the general parameter identification procedure is presented, which is finally applied to a simple material model with an initial linear elastic part and an exponential softening function to exemplarily demonstrate the methodology and a more complex example of a mesoscale model for concrete.

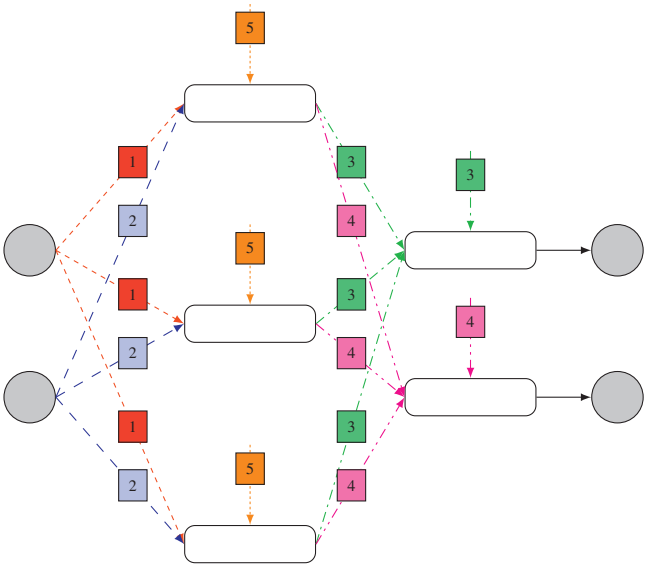

Fig. 1. Definition of groups for the free parameters (weights and biases).

2. Bayesian neural networks

2.1. Architecture

Within the Bayesian framework of neural network interpolation, the output of the neural network $\boldsymbol{y}$ given the input $\boldsymbol{x}$ is assumed to be superposed by Gaussian noise:

$$p(\boldsymbol{t}|\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{\Sigma}_\beta) = \mathcal{N}(\boldsymbol{y}(\boldsymbol{x}, \boldsymbol{w}), \boldsymbol{\Sigma}_\beta), \tag{1}$$

where $\mathcal{N}(\boldsymbol{y}(\boldsymbol{x}, \boldsymbol{w}), \boldsymbol{\Sigma}_\beta)$ is a standard joint Gaussian distribution with mean value $\boldsymbol{y}(\boldsymbol{x}, \boldsymbol{w})$ and covariance $\boldsymbol{\Sigma}_\beta$. It is to be noted that, in contrast to standard Bayesian neural networks, the authors use a full covariance matrix $\boldsymbol{\Sigma}_\beta$, with a separate noise variance for each output of the network to account for a different noise level in each output component and their mutual correlation.

In a similar way, the prior distribution of the free parameters (weights and biases), which corresponds to the distribution of the free parameters without any knowledge of the training data $\mathcal{D} = (\boldsymbol{X}, \boldsymbol{t})$, is assumed to be

$$p(w_j|\alpha_j) = \mathcal{N}(w_j|0, \alpha_j^{-1}). \tag{2}$$

Theoretically, a precision parameter α_j can be defined for all free parameters. However, the network possesses a symmetry with respect to the neurons (e.g. if neuron 1 and 2 in the hidden layer with their associated weights are exchanged, the input–output relation will remain unchanged). In order to account for this symmetry property, the free parameters are placed into groups with a single precision parameter for each group, which is illustrated in Fig. 1 for a network with a single hidden layer and two inputs and two outputs. A group is created for each input neuron with all connections (weights) emanating from that input (groups 1 and 2). Furthermore, an additional group is created for the biases $b_i^{(1)}$ of the first layer (group 5). Finally, a group for each output with its bias and all connections flowing into that output neuron (groups 3 and 4) is created. If an additional second hidden layer should be introduced, an additional group with the weights connecting both layers and the biases of the second layer has to be added.

2.2. Calculation of weights and biases

Assuming furthermore that the hyperparameters α_i and the noise covariance $\boldsymbol{\Sigma}_\beta$ are known, and using the assumption that