Predicting the Effect of Input Perturbations on Radial Basis Function Networks

J.L.Bernier, A.F.Díaz, F.J.Fernández, J.González, M.Rodríguez, J.Ortega

Dpto. Arquitectura y Tecnología de Computadores
Universidad de Granada, E18071-Granada (Spain)

Abstract. In previous work we have derived a magnitude termed 'Mean Squared Sensitivity' (MSS) to predict the performance degradation of a MLP affected by perturbations in different parameters. The present paper deals with the same problem in RBF networks to study the implications when they are affected by input noise. We have obtained the corresponding analytical expression for MSS in RBF networks and have validated it experimentally, using two different models for perturbations: an additive and a multiplicative model. Thus, MSS is also proposed as a quantitative measurement for evaluating the noise immunity of a RBFN configuration, giving even more generalization to our approach.

1 Introduction

Radial Basis Function Networks (RBFNs) [1] are neural paradigms that are currently receiving a great deal of interest and which can be considered universal approximators [2]. Nevertheless, the algorithms used to train them provide solutions that correspond to local optima in the space of parameter configurations [2]. This means that different network configurations may present a similar performance to solve a problem (a similar MSE or a similar classification error) but the degradation of this performance is different when perturbations affect its parameters. To understand this fact, we must remember that the training algorithms are based on searching for local minima of the mean squared error with respect to the parameters of the network, but that there exist many possibilities (as many as there are local minima) and that some of these minima are flatter than others [3].

The study of generalization (and also of fault tolerance and noise immunity) has been examined extensively, considering MLPs [4, 5] as well as RBFNs [5–7]. Unlike natural neural networks, the above properties are not inherent to artificial neural networks but need to be quantified [4, 8]. Furthermore, increasing the number of neural units in a network only provides an increment of potential fault tolerance but it is necessary to accomplish this in practice with an adequate distribution of learning among these units to enhance the behaviour [8]. In this way, if the structure of a network is fixed (fixing the number of layers and neurons per layer) it is possible to find different parameter configurations, some of which present better performance against some kinds of perturbations than others. For example, when the parameter configuration is obtained via computer simulation and the network must be translated into a real application where the inputs come from electronic sensors such as termopars, microphones, etc, it is interesting to take into account the effects of deviations in the simulated inputs and to choose the parameter configuration that presents the highest noise immunity among the different candidates.

A first approximation to distinguish between different configurations is to evaluate these properties correctly. Thus, in [5] the fault tolerance of a network is measured considering stuck-at faults and using the worst case hypothesis, nevertheless, as the authors remark, this heuristic method is not totally reliable because the hypothesis may be wrong due to the different combinations of faulty units that are not considered. The same method has been applied to parametric faults [7, 9], which are considered more realistic in practical VLSI implementations than stuck-at ones. Considering parametric faults, other approaches are directed at estimating the error bounds of the performace of a network using a Hessian approximation [6, 8].

In a second step, some authors have proposed new algorithms to enhance the generalization ability, fault tolerance or noise immunity of neural networks. There also exist different approaches, some of which use strategies that are independent of the training algorithm, such as randomly perturbing the network during the training process [5, 8]. Other authors have proposed new learning rules to optimise these properties [4, 10], using a regularizer, for example [11].

In [4] a suitable approximation of the MSE degradation of a Multilayer Perceptron (MLP) subject to perturbations, called Mean Squared Sensitivity (MSS), was obtained. MSS measures the MSE degradation of the MLP in the presence of deviations and its expression can be particularized to consider different types of weight or input perturbations. In this work it is considered that input deviations affect a RBFN. We consider two kinds of parametric input perturbations: additive and multiplicative, both models being frequently used in the literature [4, 6, 8, 9, 11, 12]. Thus, we obtain the expression of MSS for such kinds of deviations and show the validity of our approximations. Therefore, we propose the use of MSS as a measurement of the performance degradation of a RBFN affected by input noise, thus providing a useful criterion to distinguish between different RBFN configurations and select the one that maintains best performance against these perturbations. Although the present work does not present an algorithm to enhance noise immunity, the proposed measurement provides the necessary information to design an explicit regularization scheme, as we did for MLPs in [11].

The paper is organized as follows: in Section 2, the concept of statistical sensitivity is presented; in Sections 3 and 4 the particular expressions of statistical sensitivity to additive and multiplicative perturbations are derived, respectively. The relationship between statistical sensitivity and MSE degradation is presented in Section 5, where MSS is defined. Section 6 shows the experimental results that enable us to demonstrate the validity of the expressions obtained and, finally, Section 7 draws some conclusions.

2 Concept of statistical sensitivity

Let us consider a neuronal unit that provides an output y whose inputs may be altered from their nominal values. Statistical sensitivity [12] is defined by the following expression:

$$S = \lim_{\sigma \to 0} \frac{\sqrt{var(\Delta y)}}{\sigma} \tag{1}$$

where σ represents the standard deviation of the changes in the inputs, and $var(\Delta y)$ is the variance of the deviation in the output for the present input pattern (with respect to the output in the absence of perturbations) due to these changes, which can be computed as: $var(\Delta y) = E[(\Delta y)^2] - (E[\Delta y])^2$, with $E[\cdot]$ being the expected value of $[\cdot]$.

Note that *statistical sensitivity* is completely different from only *sensitivity.* Sensitivity is a concept frequently used in the literature related to first derivatives of the output y with respect to inputs and thus, it quantifies the dependency of the output with respect to the corresponding inputs. Statistical sensitivity, however, constitutes a statistical measurement of the magnitude of output changes due to input changes within a range. In fact, statistical sensitivity is related to second derivatives, as shown in [4]. For example, a low value of statistical sensitivity indicates that the standard deviation of the output is small when the corresponding inputs vary from their nominal values in a particular interval. A low value of sensitivity, on the other hand, indicates that the actual inputs have little influence on the corresponding output.

Without loss of generality, let us consider a RBF network consisting of n inputs, a single output, and m neurons in the hidden layer. The output of this network is then computed as the averaged sum of the outputs of the m neurons, where each neuron is a radial function of the n inputs to the network:

$$y = \sum_{i=1}^{m} w_i \Phi_i = \sum_{i=1}^{m} w_i \exp\left(-\frac{\sum_{k=1}^{n} (x_k - c_{ik})^2}{r_i^2} \right) \tag{2}$$

where x_k (k=1,...,n) are the inputs to the network, and c_{ik} and r_i are the centres and radius of the RBF associated with neuron i, respectively.

If the inputs presented to the RBF are perturbed by noise, then the output y of the network is changed with respect to its nominal output. As indicated above, the statistical sensitivity, S, enables us to estimate in a quantitative way the degradation of the expected output of a RBF neuron when the values of the inputs change by a given amount.

If the deviations considered are small enough, then the corresponding deviation in the output of the network can be approximated as:

$$\Delta y \approx \sum_{k=1}^{n} \frac{\partial y}{\partial x_k} \Delta x_k \tag{3}$$

To compute expression (2), we need to assume a model for input deviations. We have chosen parametric faults instead of stuck-at ones [7]. The selected model satisfies the following assumptions:

1) Perturbations follow normal distributions with an average equal to zero and a standard deviation equal to σ.

2) Perturbations on different inputs are not statistically correlated.

In Section 3 we consider additive perturbations while in Section 4, we consider a multiplicative nature.

3 Statistical sensitivity against additive perturbations

The additive model of input perturbations is frequently used to study the effects of input quantification [4]. For example, in some practical implementations the inputs are converted to digital, or in other cases the accuracy of the real inputs is different from the precision used via a computer simulation.

The additive model satisfies the following assumptions:

a) $E[\Delta x_k] = 0$

b) $E[(\Delta x_k \Delta x_l)] = \sigma^2 \delta_{kl}$

where δ_{kl} is the Kronecker delta. This perturbation model implies that each input x_k may be modified by a random additive variable with average equal to zero and standard deviation equal to σ. As stated above, perturbations on different inputs are assumed not to be statistically correlated.

Proposition 1: if $E[\Delta x_k] = 0 \ \forall k$ then $E[\Delta y] = 0$.

Proof 1:

$$\begin{aligned} E[\Delta y] &= E[\sum_{k=1}^{n} \frac{\partial y}{\partial x_k} \Delta x_k] = E[\sum_{k=1}^{n} \frac{\partial}{\partial x_k} \left(\sum_{i=1}^{m} w_i \Phi_i \right) \Delta x_k] \\ &= E[\sum_{k=1}^{n} \sum_{i=1}^{m} w_i \frac{\partial \Phi_i}{\partial x_k} \Delta x_k] = 2 \sum_{i=1}^{m} \frac{w_i \Phi_i}{r_i^2} \sum_{k=1}^{n} (c_{ik} - x_k)\, E[\Delta x_k] \\ &= 0 \end{aligned} \tag{4}$$

□

Proposition 2: the statistical sensitivity to additive input perturbations of a RBF network can be expressed as:

$$S = 2 \sqrt{\sum_{k=1}^{n} \left(\sum_{i=1}^{m} \frac{w_i \Phi_i}{r_i^2} (c_{ik} - x_k) \right)^2} \tag{5}$$

Proof 2:

$$E\left[(\Delta y)^2\right] = E\left[\left(2 \sum_{i=1}^{m} \frac{w_i \Phi_i}{r_i^2} \sum_{k=1}^{n} (c_{ik} - x_k)\, \Delta x_k \right) \right.$$

$$\left. \left(2\sum_{j=1}^{m} \frac{w_j \Phi_j}{r_j^2} \sum_{l=1}^{n} (c_{jl} - x_l)\, \Delta x_l \right) \right]$$

$$= 4E\left[\sum_{i=1}^{m} \frac{w_i \Phi_i}{r_i^2} \sum_{j=1}^{m} \frac{w_j \Phi_j}{r_j^2} \sum_{k=1}^{n} (c_{ik} - x_k) \sum_{l=1}^{n} (c_{jl} - x_l) \Delta x_k\, \Delta x_l \right]$$

$$= 4\sum_{i=1}^{m} \frac{w_i \Phi_i}{r_i^2} \sum_{j=1}^{m} \frac{w_j \Phi_j}{r_j^2} \sum_{k=1}^{n} (c_{ik} - x_k)\,(c_{jk} - x_k)\sigma^2$$

$$= 4\sigma^2 \sum_{k=1}^{n} \left(\sum_{i=1}^{m} \frac{w_i \Phi_i}{r_i^2} (c_{ik} - x_k) \right) \left(\sum_{j=1}^{m} \frac{w_j \Phi_j}{r_j^2}\, (c_{jk} - x_k) \right)$$

$$= 4\sigma^2 \sum_{k=1}^{n} \left(\sum_{i=1}^{m} \frac{w_i \Phi_i}{r_i^2} (c_{ik} - x_k) \right)^2 \tag{6}$$

Substituting (6) in (1), Proposition 2 is proved.

□

4 Statistical sensitivity against multiplicative perturbations

When the inputs of a network comes from electronic sensors, it is necessary to take into account the analogue tolerance margin, i.e., the exact value of an analogue electronic magnitude is unknown although we know the nominal value plus the tolerance margin supplied by the manufacturer. The multiplicative model of input perturbations is frequently used to study this situation [4, 7, 9].

The multiplicative model satisfies the following assumptions:

a) $E[\Delta x_k] = 0$

b) $E[(\Delta x_k \Delta x_l)] = \sigma^2 x_k x_l \delta_{kl}$

where δ_{kl} is the Kronecker delta. Unlike the additive model previously considered, each input x_k may be altered by a random variable, this alteration being proportional to the nominal value of x_k.

As assumption a) is the same as in the additive model, then Proposition 1 is also valid.

Proposition 3: the statistical sensitivity to multiplicative input perturbations of a RBF network can be expressed as:

$$S = 2\sqrt{\sum_{k=1}^{n} \left(x_k \sum_{i=1}^{m} \frac{w_i \Phi_i}{r_i^2} (c_{ik} - x_k) \right)^2} \tag{7}$$

Proof 3: The proof of Proposition 3 is similar to that corresponding to Proposition 2, only varying assumption b).

$\square$

In sections 3 and 4 we obtained the mathematical expression of statistical sensitivity, and so we can quantify the expected output change of a network when a particular input pattern is randomly perturbed. Nevertheless, we are interested in an overall quantification for such changes, irrespective of particular input patterns. The following section introduces such a measurement.

5 The Mean Squared Sensitivity

The learning performance of a RBF network is usually measured by means of the Mean Squared Error (MSE). This error measurement is computed by the sum of a set of input patterns whose desired output is known, and its expression is the following:

$$MSE = \frac{1}{N_p}\sum_{p=1}^{N_p}\varepsilon(p) = \frac{1}{2N_p}\sum_{p=1}^{N_p}(d(p)-y(p))^2 \tag{8}$$

where N_p is the number of input patterns considered, and $d(p)$ and $y(p)$ are the desired and obtained outputs, respectively, for the input pattern p.

If the inputs of the network suffer a deviation, the nominal output is altered, as is the expected MSE. By developing expression (8) with a Taylor expansion near the nominal MSE found after training, MSE_0, it is obtained that:

$$MSE' = MSE_0 - \frac{1}{N_p}\sum_{p=1}^{N_p}(d(p)-y(p))\Delta y(p) + \frac{1}{2N_p}\sum_{p=1}^{N_p}(\Delta y(p))^2 + 0 \tag{9}$$

Now, if the expected value of MSE' is computed, taking into account the perturbation models adopted that $E[\Delta y] = 0$, and that from (1) we deduce $E[(\Delta y)^2] \simeq \sigma^2 S^2$, the following expression is deduced:

$$E[MSE'] = MSE_0 + \frac{\sigma^2}{2N_p}\sum_{p=1}^{N_p}(S(p))^2 \tag{10}$$

By analogy with the definition of MSE, we define "Mean Squared Sensitivity" (MSS) as:

$$MSS = \frac{1}{2N_p}\sum_{p=1}^{N_p}(S(p))^2 \tag{11}$$

MSS is evaluated from the statistical sensitivities for a set of input patterns, as expression (11) shows, and from the nominal values of the network parameters. By combining expressions (10) and (11), the expected degradation of the MSE, $E[MSE']$ can be expressed in a simplified way as:

$$E[MSE'] = MSE_0 + \sigma^2 MSS \tag{12}$$

Thus, as MSE_0 and MSS can be directly computed after training from the nominal values of the parameters of the network and the same set of input patterns, it is possible to predict the degradation of MSE when the inputs of the network are deviated from their nominal values within a specific range. The only difference between the additive and multiplicative models concerns the particular expression of statistical sensitivity ((5) or (7) depending on the model) but relation (12) is valid in any case. Moreover, as can be deduced from (12), a lower value of MSS implies a lower degradation value of MSE. Note also that the calculation of MSS is not dependent on the training algorithm used. Thus, we propose using MSS as a suitable measure of the noise immunity of RBF networks against deviations.

6 Results

In order to validate the expressions presented, we compared the results experimentally obtained for $E[MSE']$ when the inputs of the network were affected by additive or multiplicative deviations with the predicted value obtained by using (12).

Three problems were considered: a predictor of the Mackey-Glass temporal series[13], an approximator of the f_8 function proposed by Cherkassky et al. [14] and a predictor of the Box-Jenkins gas furnace [15]. The structures of these networks are described in Table 1.

Table 1. Description of the RBFNs

Problem	No.inputs	No.RBFs	No.training patterns	No.test patterns
Mackey-Glass	4	14	500	500
f_8 function	2	16	400	400
Gas-furnace	2	10	291	267

Table 2 shows the values of MSE_0 and MSS obtained after training using a set of test patterns different from those used for training.

Table 2. MSE_0 and MSS obtained after training

Problem	MSE_0	MSS additive	MSS multiplicative
Mackey-Glass	$6.65\ 10^{-5}$	1.074	1.095
f_8 function	$8.41\ 10^{-4}$	9.861	5.126
Gas-furnace	0.164	1.275	583.057

The inputs of the networks were randomly deviated from their nominal values considering different values of σ and the corresponding model of deviation. Each value of $E[MSE']$ was experimentally computed over 50 tests. In each test, all the components of each input pattern p of the network $x_k(p)$ were perturbed taking a value equal to $x_k(p) + \delta_k(p)$ for the additive model, or $x_k(p)(1 + \delta_k(p))$ when the multiplicative model was considered, with $\delta_k(p)$ being a random variable that follows a gaussian distribution with average equal to zero and standard deviation equal to σ.

Figures 1 and 2 represent the experimental and predicted values of $E[MSE']$ for the Mackey-Glass problem using an additive or a multiplicative model, respectively, for deviations. The experimental values are plotted with their respective confidence levels at 95%. It can be observed that the values predicted by expression (12) accurately fit those obtained experimentally until the degradation of MSE becomes large. For example, we deduce from Figure 1 that MSE for σ equal to 0.10 is 157 times larger than MSE_0!; this means that in a real implementation the actual parameter configuration should be discarded if noise is high.In any case, the validity of the approximation is demonstrated and, according to (12), this means that the lower the MSS the lower the MSE degradation in the presence of noise. Thus, among RBF network configurations that present a similar MSE_0, the one with the lowest MSS provides the most stable output when its inputs are perturbed.

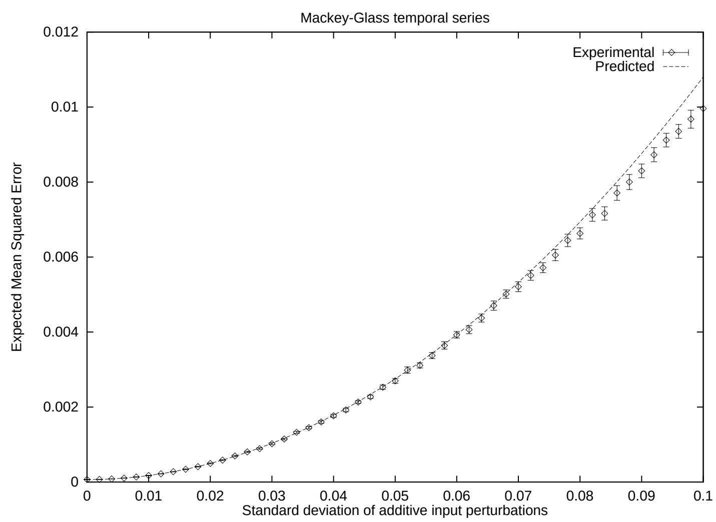

Fig. 1. Experimental and predicted $E[MSE']$ for the Mackey-Glass temporal series in the presence of additive perturbations

It is interesting to note that the value for MSS may be very different when considering additive or multiplicative deviations. For example the values of MSS for the approximator of the f_8 function are 9.861 and 5.126, respectively, while in the case of the gas-furnace problem the corresponding values are 1.275 and

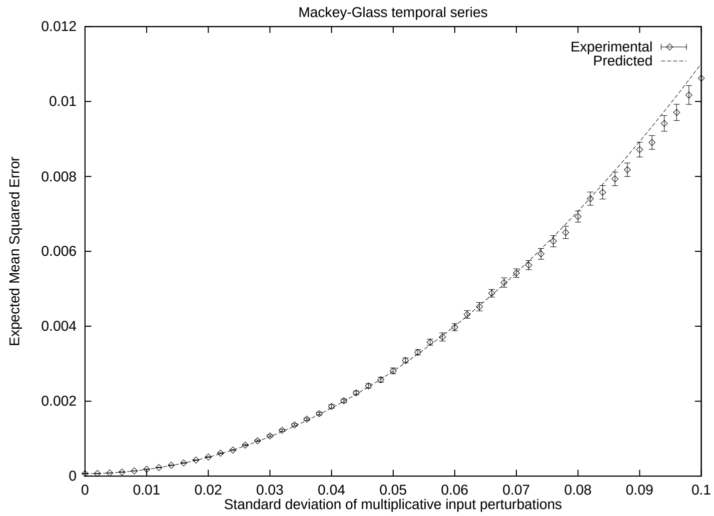

Fig. 2. Experimental and predicted $E[MSE']$ for the Mackey-Glass temporal series in the presence of multiplicative perturbations

583.057. This means that it is possible to have a network configuration with a low value of MSS against additive perturbations but that this does not necessarily imply a low value against multiplicative ones, and viceversa. Thus, good stability against additive perturbations does not guarantee similar performance when the perturbations have a different nature. This result is similar to that discussed in the case of MLPs in [4].

7 Conclusions

We have derived and validated a quantitative measure of noise immunity against input perturbations of RBF networks. This measure, which we term Mean Squared Sensitivity (MSS), is explicitly related to MSE degradation in the presence of input deviations. This relationship shows that a lower value of MSS implies a lower degradation of MSE. Thus, we have introduced a useful criterion for selecting between different network configurations that present a similar MSE to solve a particular problem: the one that maintains its performance in the presence of input noise.

We have used two different models for perturbation, an additive and a multiplicative one, obtaining the expression to compute the corresponding value of MSS for such kinds of perturbations.

Moreover, the analytical expression of MSS can be used as a regularizer during the training process in order to improve the final performance of the network with respect to noise immunity in the same way as was done for MLPs in [11].

Acknowledgements

This work has been supported in part by projects CICYT TIC2000-1348 and DIP2001-3219.

References

1. D.S. Broomhead and D. Lowe, "Multivariate Functional Interpolation and Adaptive Networks", *Complex Systems* **2**, 321-355.
2. S. Haykin, *Neural Networks. A Comprehensive Foundation*, (second edition, Prentice Hall, 1999).
3. H. Hochreiter and J. Schmidhuber, "Flat Minima", *Neural Computation* **9**, 1-42 (1997).
4. J.L. Bernier, J. Ortega, E. Ros, I. Rojas and A. Prieto, "A Quantitative Study of Fault Tolerance, Noise Immunity and Generalization Ability of MLPs", *Neural Computation* **12**, 2941-2964 (2000).
5. B.E. Segee and M.J. Carter, "Comparative Fault Tolerance of Generalized Radial Basis Function and Multilayer Perceptron Networks", *IEEE Int. Conference on Neural Networks*, **3**, 1847-1852 (1993).
6. N.W. Townsend and L. Tarassenko, "Estimations of Error Bounds for Neural-Network Function Approximators", *IEEE Trans. on Neural Networks* **10** (2), 217-230 (1999).
7. X. Parra and A. Catala, "Fault Tolerance in the Learning Algorithm of Radial Basis Function Networks", *Proc. IEEE-INNS-ENNS Int. Joint Conference on Neural Networks*, **3**, 527-532 (2000).
8. P.J. Edwards and A.F. Murray, "Towards Optimally Distributed Computation", *Neural Computation*, **10**, 997-1015 (1998).
9. A. Catala and X. Parra, "Fault Tolerance Parameter Model of Radial Basis Function Networks", *IEEE Int. Conference on Neural Networks*,**2**, 1384-1389 (1996).
10. M.V. Hegde, M. Naraghi-Pour and P. Bapat, "Learning Algorithms for Fault Tolerance in Radial Basis Function Networks", *Proc 37th Midwest Symposium on Circuits and Systems*, **1**, 535-538 (1995).
11. J.L. Bernier, J. Ortega, I. Rojas, E. Ros and A. Prieto, "Obtaining Fault Tolerant Multilayer Perceptrons Using an Explicit Regularization".*Neural Processing Letters* **12** (2), 107-113 (2000).
12. J.Y. Choi and C. Choi, "Sensitivity Analysis of Multilayer Perceptron with Differentiable Activation Functions", *IEEE Trans. on Neural Networks* **3** (1), 101-107 (1992).
13. M.C. Mackey and L. Glass, "Oscillation and Chaos in Physiological Control Systems", *Science* **197**, 287-289 (1977).
14. V. Cherkassky, D. Gehring and F. Mulier, "Comparison of Adaptive Methods for Function Estimation from Samples", *IEEE Trans. on Neural Networks* **7** (4), 969-984 (1996).
15. G.E.P. Box and G.M. Jenkins, "Time Series Analysis, Forecasting and Control", CA:Holden-Day 2nd ed., San Francisco (1976).