Input Noise Immunity of Multilayer Perceptrons

Youngjik Lee and Sang-Hoon Oh

ETRI Journal, volume 16, number 1, pp. 35-43, April 1994

http://www.etri.re.kr

Abstract

In this paper, The robustness of the artificial neural networks to noise is demonstrated with a multi-layer perceptron, and the reason of robustness is due to the statistical orthogonality among hidden nodes and its hierarchical information extraction capability. Also, the misclassification probability of a well-trained multilayer perceptron is derived without any linear approximations when the inputs are contaminated with random noises. The misclassification probability for a noisy pattern is shown to be a function of the input pattern, noise variances, the weight matrices, and the nonlinear transformations. The result is verified with a handwritten digit recognition problem, which shows better result than that using linear approximations.

I.Introduction

Neural networks are employed in most pattern classification applications since they have learning ability and provide a greater degree of robustness than von Neumann sequential computers[1]. It is believed that these properties are achieved via dense interconnection of many nonlinear computational elements. In order to investigate the robustness, Stevenson et al. derived the misclassification probability of Madaline due to weight or input perturbation, assuming input patterns are distributed uniformly over the input space and the weights have arbitrary values[2]. However, the input patterns are sampled from the original population which is rarely uniform distribution, and the misclassification probability is surely a function of the trained weights. Also, Xie and Jabri derived the effects of quantization under the unrealistic assumption that the inputs, the trained weights, and the weighted sums are distributed uniformly in certain ranges[3]. Choi et al. used the first order Taylor approximation method[4], however this has limited range of application.

In this paper, we explain the robustness of MLPs(Multi-Layer Perceptrons) to noise and derive the misclassification probability when the input patterns are contaminated with random noises[5]. In section II, the reasons for the robustness to noise, namely, "noise immunity", is logically analyzed. In section III, we derive the misclassification probability of a single hidden-layer perceptron without any linear approximation. Since the derived result needs intensive computation, we suggest simpler method in section IV, which considers a subset of the output nodes consisting of the target node and the most active non-target node and utilizes the function approximation capability of MLPs. This result is verified with a handwritten digit recognition problem, which is described in section V, and section VI concludes this paper.

II. Noise Immunity of Multilayer Perceptrons

It is well known that MLPs are robust to noise contamination in inputs and/or weights, including the case of quantization. In this section, we explain how MLPs have these properties in two ways.

First, the orthogonal property among the output values of the hidden nodes reduces the noise effect. It is well known that the hidden weight vectors tend to be near orthogonal through learning procedure for efficient feature extraction of input patterns[6]. Thus, after successful learning, the weighted sums to hidden nodes are much less correlated even when a pattern with correlated noise is presented to the input layer. Also, the magnitude of correlation coefficient between the weighted sums decreases under the sigmoidal transformations[7]. Therefore, the correlations among hidden nodes should be very small. As a result, the noise effects are averaged out when the hidden output values are summed through output weights.

Second, noise immunity of MLPs can be explained in the information-theoretic point of view[8]. It is reported that MLPs have hierarchical information extraction capabilities acquired through learning[9]. It is argued there that the input pattern set has the inter-class information as well as the intra-class variation. The inter-class information is the information content that an input pattern belongs to a specific class, and the intra-class variation is a measure of the average variations within the classes including noise contaminations. After learning, each layer of MLPs tries to keep the inter-class information and to minimize the intra-class variation as much as possible. When a noisy pattern is presented to the input, MLPs extract the inter-class information and suppress the noise components, yielding noise immunity of MLPs.

In the next section, we derive the misclassification probability of single hidden-layer perceptrons.

III. Problem Formulation

In this section, we derive the misclassification probability of an MLP when the inputs are contaminated with additive Gaussian noises. Consider a well-trained single hidden-layer perceptron with N inputs, H hidden nodes, and M output nodes. When a training pattern $\mathbf{x}^{(p)} = [x_1^{(p)}, x_2^{(p)}, \ldots, x_N^{(p)}]^T$ is contaminated with an independent, identically distributed(i.i.d.) Gaussian noise $\mathbf{x}^n = [x_1^n, x_2^n, \ldots, x_N^n]^T$ with zero-mean and standard deviation σ_I, the input is

$$\mathbf{x} = \mathbf{x}^{(p)} + \mathbf{x}^n, \tag{1}$$

and the weighted sum to the jth hidden node can be written as

$$a_j = a_j^{(p)} + \sum_{i=1}^{N} w_{ji} x_i^n, j = 1, 2, \ldots, H \tag{2}$$

where $\mathbf{W} = (w_{ji})$ is the weight matrix from the input to the hidden layer and

$$a_j^{(p)} \triangleq w_{j0} + \sum_{i=1}^{N} w_{ji} x_i^{(p)}. \tag{3}$$

The case when the additive noises are not i.i.d. Gaussian but finite variances is discussed at the end of section V. It is easy to show that $\mathbf{a} = [a_1, a_2, \ldots, a_H]^T$ is a Gaussian random vector[10], and its probability density function(p.d.f.) is given by

$$f_a(\mathbf{a}) = \frac{1}{\sqrt{(2\pi)^H |\mathbf{C}|}} \exp[-\frac{1}{2}(\mathbf{a} - E[\mathbf{a}])^T \mathbf{C}^{-1}(\mathbf{a} - E[\mathbf{a}])] \tag{4}$$

where $\mathbf{C} = (C_{ij})$ is the covariance matrix of $\mathbf{a}$ whoes i, jth element is

$$C_{ij} = \sigma_{a_i}\sigma_{a_j} r_{ij} = \sigma_I^2 \sum_{k=1}^{N} w_{ik} w_{jk}. \tag{5}$$

Here, r_{ij} denotes the correlation coefficient between a_i and a_j. The output random vector $\mathbf{h} = [h_1, h_2, \ldots, h_H]^T$ of the hidden layer can be calculated using

$$h_i = \frac{2}{1 + \exp(-a_i/T)} - 1. \tag{6}$$

Hence, the p.d.f. of $\mathbf{h}$ can be derived[10] as

$$f_h(\mathbf{h}) = \frac{f_a(-T\ln[\frac{2}{h_1+1} - 1], \ldots, -T\ln[\frac{2}{h_H+1} - 1])}{|J_h(a_1, a_2, \ldots, a_H)|} \tag{7}$$

where

$$J_h(a_1, a_2, \ldots, a_H) = (\frac{1}{2T})^H \prod_{k=1}^{H} (1 - h_k^2). \tag{8}$$

It is clear that $\mathbf{h}$ is not Gaussian vector. However, it is easy to see that the elements of $\mathbf{h}$ have finite variances.

The weighted sum to the kth output node is

$$b_k = v_{k0} + \sum_{j=1}^{H} v_{kj} h_j, \;\; k = 1, 2, \ldots, M \tag{9}$$

where $\mathbf{V} = (v_{kj})$ is the weight matrix between the hidden layer and the output layer. To calculate the p.d.f. of b_k, we introduce $H - M$ auxiliary variables[10], i.e.,

$$b_k = h_k, \;\; k = M+1, M+2, \ldots, H. \tag{10}$$

Letting $\boldsymbol{\beta} = [b_1 - v_{10}, \ldots, b_M - v_{M0}, b_{M+1}, \ldots, b_H]^T$, it can be represented as

$$\boldsymbol{\beta} = \mathbf{Zh} \tag{11}$$

where

$$\mathbf{Z} = \begin{bmatrix} v_{11} & v_{12} & \cdots & v_{1,M+1} & \cdots & v_{1H} \\ v_{21} & v_{22} & \cdots & v_{2,M+1} & \cdots & v_{2H} \\ \vdots & \vdots & & \vdots & & \vdots \\ v_{M1} & v_{M2} & \cdots & v_{M,M+1} & \cdots & v_{MH} \\ & & & 1 & & \mathbf{O} \\ & \mathbf{O} & & & \ddots & \\ & & & & & 1 \end{bmatrix}. \tag{12}$$

If the output weight matrix $\mathbf{V}$ is of full rank, $\mathbf{Z}$ should be non-singular. Thus, the inverse of $\mathbf{Z}$ exists and

$$\mathbf{h} = \mathbf{Z}^{-1}\boldsymbol{\beta}. \tag{13}$$

The p.d.f. of $\boldsymbol{\beta}$ is given by

$$f_\beta(\boldsymbol{\beta}) = \frac{f_h(\mathbf{Z}^{-1}\boldsymbol{\beta})}{|J(h_1, h_2, \ldots, h_H)|} \tag{14}$$

where $J(h_1, \ldots, h_H)$ is the Jacobian of Eq. (11). The p.d.f. of $\mathbf{b} = [b_1, b_2, \ldots, b_M]^T$ can be obtained by integrating Eq. (14) on all the auxiliary variables, i.e.,

$$f_b(\mathbf{b}) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty} f_\beta(\boldsymbol{\beta})dh_{M+1}\ldots dh_H, \tag{15}$$

and the probability of misclassification when $\mathbf{x}^{(p)}$ is in the class corresponding to the first output node can be calculated by

$$Pr(E_p) = 1 - \int_{-\infty}^{\infty}\int_{-\infty}^{b_1}\cdots\int_{-\infty}^{b_1} f_b(\mathbf{b})db_M\ldots db_2 db_1. \tag{16}$$

Calculating $Pr(E_p)$ involves H-dimensional integration, which is computationally intensive.

IV. Two Node Approximation

In the previous section, we have considered all output nodes to calculate the probability of misclassification. Since the misclassification mainly occurs between the target node b_t and the output node b_ϕ with the largest $\sigma_{b_k}/|E[b_k]|$, we try to approximate the probability considering only the joint p.d.f. of the two nodes.

Since the hidden node values have finite variances, the central limit theorem[10] can be applied for large H with the fact that the correlations among them are small[7]. Thus, they can be approximated as asymptotically joint Gaussian random variables and their p.d.f. is

$$f(b_t, b_\phi) = \frac{1}{2\pi\sigma_{b_t}\sigma_{b_\phi}\sqrt{1-{r_{t\phi}}^2}} \exp[-\frac{1}{2(1-{r_{t\phi}}^2)}(\frac{(b_t-\eta_{b_t})^2}{{\sigma_{b_t}}^2} - 2r_{t\phi}\frac{(b_t-\eta_{b_t})(b_\phi-\eta_{b_\phi})}{\sigma_{b_t}\sigma_{b_\phi}} + \frac{(b_\phi-\eta_{b_\phi})^2}{{\sigma_{b_\phi}}^2})] \tag{17}$$

where the means $\eta_{b_t}, \eta_{b_\phi}$, the variances ${\sigma_{b_t}}^2, {\sigma_{b_\phi}}^2$, and the correlation coefficient $r_{t\phi}$ can be calculated by

$$\eta_{b_k} = v_{k0} + \sum_{j=1}^{H} v_{kj} E[h_j], \ k = t, \phi, \tag{18}$$

$$\sigma_{b_k}^2 = \sum_{j=1}^{H} (v_{kj}\sigma_{h_j})^2 + 2\sum_{i=1}^{H-1}\sum_{j=i+1}^{H} v_{ki}v_{kj}C_{ij}^{(h)}, \ k = t, \phi, \tag{19}$$

and

$$r_{t\phi} = \frac{\sum_{j=1}^{H} v_{tj}v_{\phi j}{\sigma_{h_j}}^2 + \sum_{i=1}^{H-1}\sum_{j=i+1}^{H}(v_{ti}v_{\phi j} + v_{\phi i}v_{tj})C_{ij}^{(h)}}{\sigma_{b_t}\sigma_{b_\phi}} \tag{20}$$

Here, ${\sigma_{h_j}}^2$ is the variance of h_j and $C_{ij}^{(h)}$ can be calculated as

$$C_{ij}^{(h)} = \sigma_{h_i}\sigma_{h_j} r_{ij}^{(h)} \tag{21}$$

where $r_{ij}^{(h)}$ is the correlation coefficient between h_i and h_j.

Forward computing the probability, we should know $E[h_i], E[h_i^2]$, and $r_{ij}^{(h)}$. $E[h_i]$ and $E[h_i^2]$ can be calculated through the numerical integration using the p.d.f. of a_i. But, the numerical integration for $r_{ij}^{(h)}$ is two dimensional, which also needs intensive computation. Noting that $r_{ij}^{(h)}$ is a continuous function of $E[a_i], {\sigma_{a_i}}^2, E[a_j], {\sigma_{a_j}}^2$, and r_{ij}. it can be approximated using an MLP[11]. This MLP, called as "correlation MLP" in this paper, consists of five inputs, thirteen hidden nodes, and one output node. The five inputs are $E[a_i], {\sigma_{a_i}}^2, E[a_j], {\sigma_{a_j}}^2$, and r_{ij}, which can be calculated using Eq. (2) and (5) for given σ_I^2, and the output is $r_{ij}^{(h)}$ which is the correlation coefficient after sigmoidal transformation. This MLP is trained with 77,824 training data sets and tested with 10,000 data sets. The mean squared error for the test data is 0.0004. This result is accurate enough to replace the numerical integration for $r_{ij}^{(h)}$ with the correlation MLP.

Using these results, the probability of misclassification with given $\mathbf{x}^{(p)}$ and σ_I can be approximated as

$$Pr(E_p) \approx 1 - \frac{1}{\sigma_{b_t}\sqrt{2\pi}}\int_{-\infty}^{\infty} \exp[-\frac{(b_t-\eta_{b_t})^2}{2{\sigma_{b_t}}^2}]G(-\frac{1}{\sqrt{1-r_{t\phi}^2}}(\frac{b_t-\eta_{b_\phi}}{\sigma_{b_\phi}} - r_{t\phi}\frac{b_t-\eta_{b_t}}{\sigma_{b_t}}))db_t \tag{22}$$

which involves only one dimensional integration. Here, $G(.)$ is the Gaussian distribution function.

We verify the result in Eq. (22) with a handwritten digit recognition problem, which is described in the next section.

V. Simulation result

To verify the derivation, input noise immunity of an MLP is simulated on a handwritten digit recognition problem. 3,000 handwritten digit images of size 16×16 pixels gathered from 40 persons are used for training. We use thinning, feature extraction (4 directional features and 4 branch features), and dimension reduction for preprocessing, yielding 128 dimensional data. The MLP consists of 128 inputs, 19 hidden nodes, and 10 output nodes. After successful training, we add zero mean, independent Gaussian random noises with various σ_I^2 to the training patterns, and estimate the probability of misclassification.

For investigating the input noise immunity of the digit "7", 70 patterns are randomly selected from 300 training patterns for "7". The misclassification probability of a pattern for given σ_I^2 is estimated by a simulation of 1,000 times, and the simulated results for the 70 patterns are averaged to draw the solid line with circles in Fig. 1. Also, the calculated results for them using Eq. (22) are averaged to draw the solid line in Fig. 1. The calculations give a closer bound than that using linear approximations by Choi et al.[4]. They approximated the sigmoidal activation function with first order Taylor series approximation in order to derive the sensitivity of MLPs. Using this approximation, the approximated output values of saturated hidden nodes have smaller variances than the real ones and the misclassification probability has large errors as shown in Fig. 1. However, there are differences between the calculations using Eq. (22) and the simulations. The reasons of the differences are the two node approximation and the assumption that the weighted sums to output nodes are jointly Gaussian. With large variance of input noise, the weighted sums to output nodes tend to have Gamma distributions rather than Gaussian. Fig. 2 is a similar result averaged for all digits.

The input noise immunity for test patterns can be calculated with the same method by using test patterns in Eq. (1) instead of training patterns. This result can be applied to select the optimal weight set from many weight sets acquired through a number of

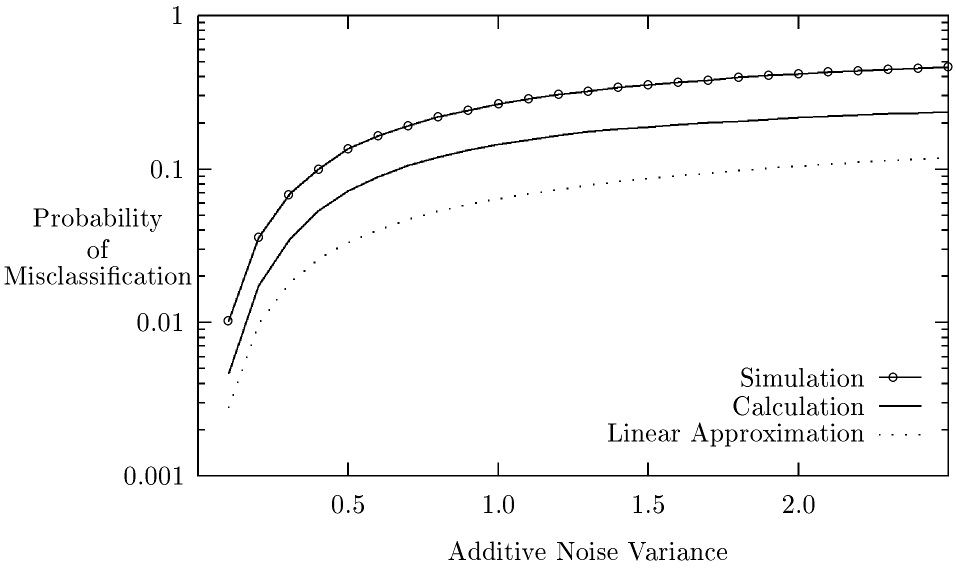

Figure 1: Additive noise variance vs. corresponding probability of misclassification for the digit patterns "7".

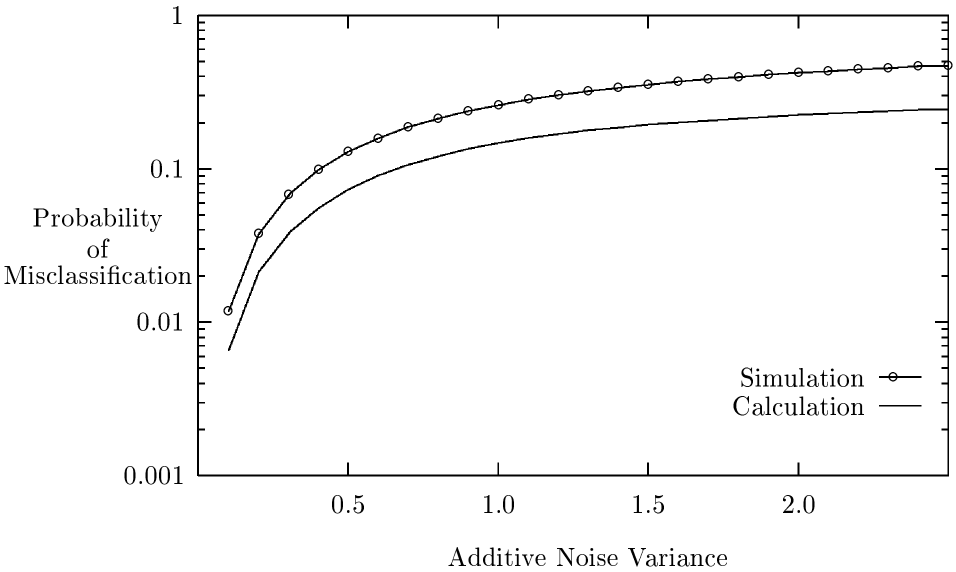

Figure 2: Additive noise variance vs. corresponding probability of misclassification for all the digit patterns.

learning trials. Among these weight sets, an MLP with the optimal set will have the best generalization capability and the best input noise immunity.

In section III and IV, the contaminated noises are assumed to be i.i.d. Gaussian random noises. This assumption is valid if we deal with the measurement noise of sensors such as scanner, CCD camera etc. When we deal with the variations such as thickness, writing style, and the variations of handwritten characters, the additive noises can be neither i.i.d. nor Gaussian. But, it is still true that they have finite variances. Thus, we need to calculate only $E[a_i], {\sigma_{a_i}}^2$, and r_{ij} according to the statistical property of the additive noises, and we can use the same procedure for computing the misclassification probability since the weighted sums to hidden nodes is a jointly Gaussian random vector by the central limit theorem.

VI. Conclusion

In this paper, we investigate the noise immunity of MLPs and derive the misclassification probability of a well-trained single-hidden layer perceptron when the input patterns are contaminated with random noises. Since the derived result involves high dimensional integration, we suggest a simpler method which needs only one dimensional integration. The simpler method considers only a subset of output nodes consisting of the target node and the non-target node which has the largest noise effect, and utilizes the function approximation capability of MLPs. This result is verified with a handwritten digit recognition problem, which shows better result than that using linear approximations. The proposed method can be applied to select the optimal weight set from many weight sets trained. Extension of this result to random binary additive noises will be practically valuable.

VII. Acknowledgement

This research was partly funded by Korea Telecom and the Ministry of Communication. The authors wish to thank Dr. El Hang Lee for his continuing support and guidance for this research.

References

[1] R. P. Lippmann, "An introduction to computing with neural nets," *IEEE ASSP Magazine*, pp. 4–22, Apr. 1987.

[2] M. Stevenson, R. Winter, and B. Widrow, "Sensitivity of feedforward neural networks to weight errors," *IEEE Trans. Neural Networks*, vol. 1, pp. 71–90, Mar. 1990.

[3] Y. Xie and M. A. Jabri, "Analysis of the effects of quantization in multilayer neural networks using a statistical model," *IEEE Trans. Neural Networks*, vol. 3, pp. 334–338, Mar. 1992.

[4] J. Y. Choi and C.-H. Choi, "Sensitivity analysis of multilayer perceptron with differentiable activation functions," *IEEE Trans. Neural Networks*, vol. 3, pp. 101–107, 1992.

[5] S.-H. Oh and Y. Lee, "An analysis on the classification performance of multilayer perceptrons," *Proc. IJCNN'92 Beijing*, vol. II, pp. 787–792, Nov. 1992.

[6] Q. Xue, Y. Hu, and W. J. Tompkins, "Analyses of the hidden units of backpropagation model by singular value decomposition," *Proc. IJCNN June 17-21, 1990, San Diego, USA*, vol. I, pp. 739–742, June 1990.

[7] S.-H. Oh and Y. Lee, "Effect of nonlinear transformations on correlation between weighted sums in multilayer perceptrons," *IEEE Trans. Neural Networks*, vol. 5, pp. 508–510, 1994.

[8] Y. S. Abu-Mostafa, "Information theory, complexity, and neural networks," *IEEE Communication Magazine*, pp. 22–28, Nov. 1989.

[9] Y. Lee and H. K. Song, "Analysis on the efficiency of pattern recognition layers using information measures," *Proc. IJCNN'93 Nagoya*, vol. III, pp. 2129–2132, Oct. 1993.

[10] A. Papoulis, *Probability, Random variables, and Stochastic Processes.* McGraw-Hill, New York, 2nd edition, 1984.

[11] Y. Ito, "Approximation of continuous functions on r^d by linear combinations of shifted rotations of a sigmoid function with and without scaling," *Neural Networks*, vol. 5, pp. 105–115, 1992.