http://dx.doi.org/10.5392/IJoC.2011.7.3.001

A Statistical Perspective of Neural Networks for Imbalanced Data Problems

Sang-Hoon Oh
Department of Information Communication Engineering
Mokwon University, Daejon, 305-755, Korea

ABSTRACT

It has been an interesting challenge to find a good classifier for imbalanced data, since it is pervasive but a difficult problem to solve. However, classifiers developed with the assumption of well-balanced class distributions show poor classification performance for the imbalanced data. Among many approaches to the imbalanced data problems, the algorithmic level approach is attractive because it can be applied to the other approaches such as data level or ensemble approaches. Especially, the error back-propagation algorithm using the target node method, which can change the amount of weight-updating with regards to the target node of each class, attains good performances in the imbalanced data problems. In this paper, we analyze the relationship between two optimal outputs of neural network classifier trained with the target node method. Also, the optimal relationship is compared with those of the other error function methods such as mean-squared error and the n-th order extension of cross-entropy error. The analyses are verified through simulations on a thyroid data set.

***Keywords**: Optimal Solution, Imbalanced Data, Error Function, Statistical Analysis.*

1. INTRODUCTION

There have been reports that, in a wide area of classifications, unusual or interesting class is rare among a general population [1]-[9]. This imbalanced class distributions have posed a serious difficulty for most classifiers which are trained under the assumption that class priors are relatively balanced and error costs of all classes are equal [1][2]. However, applications require a fairly high rate of correct detection in the minority class [3]. In order to achieve the requirement, there have been many attempts which can be categorized into the data level [3]-[7], algorithmic level [7]-[9], and ensemble approaches [1][4]. Among the three approaches, the algorithmic level approach is attractive because it can be adopted in the data level or ensemble approaches.

Feed-forward neural networks are widely applied to pattern classification problems and a popular method of training is the error back-propagation (EBP) algorithm using the mean-squared error (MSE) [10]. When applying the EBP algorithm to the imbalanced data, majority class samples have a greater chance of training and the boundary of majority class is enlarged towards the minority class boundary [4]. This is so-called “the boundary distortion”. As a result, the minority class samples have a less chance to be classified. One effective classification method to deal with the imbalanced data is the threshold moving method, which adjusts the threshold of each class such that the minority class is detected with more possibility [8].

If there is a severe imbalance of data distribution, outputs of neural networks have a high probability of “incorrect saturation” [11][12]. That is, outputs of neural networks are on the wrong extreme side of the sigmoid activation function. Although the EBP algorithm using the *n*-th order extension of cross-entropy (*n*CE) error function greatly reduces the incorrect saturation [12], it does not deal with the boundary distortion problem. In order to improve the EBP algorithm for the imbalanced data, *n*CE error function is modified such that weights associated with the target node of minority class are more strongly updated than those associated with the target node of majority class [13]. In this paper, we analyze the relationship between two optimal outputs of the neural network classifier. The analyses provide considerable insights of the neural network classifier for the imbalanced data. In Section 2, the EBP algorithm for the imbalanced data is briefly introduced. The statistical analyses of optimal solutions for MSE, *n*CE and the target node methods are conducted in Section 3 and they are verified through simulations of a thyroid data in Section 4. Finally, Section 5 concludes this paper.

2. ERROR BACK-PROPAGATION ALGORITHM FOR IMBALANCED DATA

Consider a feed-forward neural network-so called “an MLP (multilayer perceptron)” consisting of *N* inputs, *H* hidden nodes, and *M* output nodes. When a *p*-th training sample $\mathbf{x}^{(\mathrm{p})} = [x_1^{(p)}, x_2^{(p)}, \ldots, x_N^{(p)}]$ $(p = 1,2,\ldots,P)$ is presented to the MLP, the *j*-th hidden node is given by

[*] *Corresponding author. E-mail : shoh@mokwon.ac.kr*

Manuscript received Jul 22, 2011 ; accepted Aug.26, 2011

$$h_j^{(p)} = h_j(\mathbf{x}^{(\mathrm{p})}) \\ = \tanh((w_{j0} + \sum_{i=1}^{N} w_{ji} x_i^{(p)})/2), \ j = 1,2,\ldots,H. \qquad (1)$$

Here, w_{ji} denotes the weight connecting x_i to h_j and w_{j0} is a bias. The *k*-th output node is

$$y_k^{(p)} = y_k(\mathbf{x}^{(\mathrm{p})}) = \tanh(\hat{y}_k^{(p)}/2), \ k = 1,2,\ldots,M, \qquad (2)$$

where

$$\hat{y}_k^{(p)} = v_{k0} + \sum_{j=1}^{H} v_{kj} h_j^{(p)} . \qquad (3)$$

Also, v_{k0} is a bias and v_{kj} denotes the weight connecting h_j to y_k. Let the desired output vector corresponding to the training sample $\mathbf{x}^{(\mathrm{p})}$ be $\mathbf{t}^{(\mathrm{p})} = [t_1^{(p)}, t_2^{(p)}, \ldots, t_M^{(p)}]$, where the class from which $\mathbf{x}^{(\mathrm{p})}$ originates is coded as follows:

$$t_k^{(p)} = \begin{cases} +1, & \text{if } \mathbf{x}^{(p)} \text{ originates from class } k, \\ -1, & \text{otherwise.} \end{cases} \qquad (4)$$

Here, y_k is the target node of class *k*.

The conventional MSE function for *P* training samples is

$$E_{MSE} = \frac{1}{2} \sum_{p=1}^{P} \sum_{k=1}^{M} \left(t_k^{(p)} - y_k^{(p)}\right)^2 . \qquad (5)$$

To minimize E_{MSE}, weights v_{kj}'s are iteratively updated by

$$\Delta v_{kj} = -\eta \frac{\partial E_{MSE}}{\partial v_{kj}} = \eta \delta_k^{(p)} h_j^{(p)}, \qquad (6)$$

where

$$\delta_k^{(p)} = -\frac{\partial E_{MSE}}{\partial \hat{y}_k^{(p)}} = \left(t_k^{(p)} - y_k^{(p)}\right) \frac{\left(1 - y_k^{(p)}\right)\left(1 + y_k^{(p)}\right)}{2} \qquad (7)$$

is the error signal and η is the learning rate. Also, weights w_{ji}'s are updated by

$$\Delta w_{ji} = -\eta \frac{\partial E_{MSE}}{\partial w_{ji}} = \eta x_i^{(p)} \sum_{k=1}^{M} v_{kj} \delta_k^{(p)} . \qquad (8)$$

The above weight-updating procedure is the EBP algorithm [10].

Let us assume that there are two classes, where one is the minority class C_1 with P_1 training samples and the other is the majority class C_2 with P_2 training samples ($P_1 << P_2$). If we use the conventional EBP algorithm to train the MLP [10], weight-updating is overwhelmed by P_2 samples of the majority class and this severely distorts the class boundary between the two classes. That is, the boundary of the majority class is enlarged to the boundary of the minority class [4]. This gives a less chance to be classified for the minority samples while samples in the majority class have a greater chance to be classified. Finally, we attain poor classification performance for the minority class in spite of a high misclassification cost for the minority class.

The easiest way to deal with the imbalanced class distribution is the threshold moving method [8]. In the testing phase after training of MLP, the classification threshold of C_1 is decreased so that the minority class samples are classified with more possibility.

In order to prevent the boundary distortion, Oh proposed the error function which can intensify weight-updating associated with the target node of the minority class and weaken weight-updating associated with the target node of the majority class [13]. Accordingly, the proposed error function in [13] was defined by

$$E_{TN} = -\sum_{p=1}^{P} [\int \frac{t_1^{(p)^{n+1}} \left(t_1^{(p)} - y_1^{(p)}\right)^n}{2^{n-2}\left(1 - y_1^{(p)^2}\right)} dy_1^{(p)} \\ + \int \frac{t_2^{(p)^{m+1}} \left(t_2^{(p)} - y_2^{(p)}\right)^m}{2^{m-2}\left(1 - y_2^{(p)^2}\right)} dy_2^{(p)}], \qquad (9)$$

where *n* and *m* (*n*<*m*) are positive integers and the MLP has two output nodes whose desired values are given by (4). If *n*=*m*, the proposed error function is the same as the *n*CE error function proposed in [12] which dramatically reduces the incorrect saturation of output nodes.

The error signal based on E_{TN} is given by

$$\delta_k^{(p)} = -\frac{\partial E_{TN}}{\partial \hat{y}_k^{(p)}} = \begin{cases} t_1^{(p)^{n+1}} (t_1^{(p)} - y_1^{(p)})^n / 2^{n-1} \text{ for } k = 1, \\ t_2^{(p)^{m+1}} (t_2^{(p)} - y_2^{(p)})^m / 2^{m-1} \text{ for } k = 2. \end{cases} \qquad (10)$$

Since *n*<*m*, $\left|\delta_1^{(p)}\right| \geq \left|\delta_2^{(p)}\right|$ for $-1 < y_k^{(p)} < 1$. Associated weights are updated proportional to $\delta_k^{(p)}$ given by (10). E_{TN} can prevent the boundary distortion as well as the incorrect saturation of output nodes.

3. ANALYSES OF RELATIONSHIP BETWEEN OPTIMAL SOLUTIONS

In the limit $P \to \infty$, the minimizer of E_{MSE} converges (under certain regularity conditions, Theorem 1 in [14]) towards the minimizer of the function

$$E\{E_{MSE}(\mathbf{X})\} = E\left\{\frac{1}{2}\sum_{k=1}^{M}(T_k - y_k(\mathbf{X}))^2\right\}, \qquad (11)$$

where $E\{\cdot\}$ is the expectation operator, T_k is the random variable of the desired value and $\mathbf{X}$ is the random input vector. The optimal solution minimizing the criterion (11) [in the space of all functions taking values in (-1,1)] is given by $\mathbf{b}(\mathbf{X})$ whose components are [12][14]

$$b_k(\mathbf{x}) = E\{T_k \mid \mathbf{x}\} = 2Q_k(\mathbf{x}) - 1, k = 1,2,\ldots,M. \qquad (12)$$

Here, $Q_k(\mathbf{x}) = \Pr[\mathbf{X} \text{ originates from class } k \mid \mathbf{X} = \mathbf{x}]$ is the posterior probability. We assume that the MLP has two outputs in order to cope with the bi-class imbalanced data problems. Then, by substituting

$$Q_2(\mathbf{x}) = 1 - Q_1(\mathbf{x}) \qquad (13)$$

into (12), the relationship between the two optimal outputs is given by

$$b_2(\mathbf{x}) = -b_1(\mathbf{x}). \qquad (14)$$

For the *n*CE error function given by

$$E_{nCE} = -\sum_{p=1}^{P}\sum_{k=1}^{2}\int \frac{t_k^{(p)n+1}\left(t_k^{(p)} - y_k^{(p)}\right)^n}{2^{n-2}\left(1-y_k^{(p)2}\right)} dy_k^{(p)}, \quad (15)$$

the optimal solutions are [12]

$$b_k(\mathbf{x}) = g(h_n(Q_k(\mathbf{x}))),\ k = 1,2. \quad (16)$$

Here,

$$g(u) = \frac{1-u}{1+u} \ \text{and}\ h_n(q) = \left(\frac{1-q}{q}\right)^{1/n}. \quad (17)$$

Since $Q_2(\mathbf{x}) = 1 - Q_1(\mathbf{x})$ and

$$Q_1(\mathbf{x}) = h_n^{-1}\left(g^{-1}(b_1(\mathbf{x}))\right), \quad (18)$$

we can get

$$b_2(\mathbf{x}) = g(h_n(Q_2(\mathbf{X}))) = g\left(h_n\left(1 - h_n^{-1}\left(g^{-1}(b_1(\mathbf{x}))\right)\right)\right). \quad (19)$$

Using

$$g^{-1}(v) = \frac{1-v}{1+v} \ \text{and}\ h_n^{-1}(r) = \frac{1}{1+r^n}, \quad (20)$$

(19) can be rewritten as

$$b_2(\mathbf{x}) = g\left(\frac{1+b_1(\mathbf{x})}{1-b_1(\mathbf{x})}\right) = -b_1(\mathbf{x}) \quad (21)$$

which is the same result with (14). Because E_{MSE} and E_{nCE} have optimal solutions which are not varying with respect to *k* (as given by (12) and (16) respectively), the relationship between two optimal outputs is a straight line with a negative slope.

The optimal solution minimizing E_{TN} can be derived as

$$b_1(\mathbf{x}) = g(h_n(Q_1(\mathbf{x}))) \ \text{and}\ b_2(\mathbf{x}) = g(h_m(Q_2(\mathbf{x}))), \quad (22)$$

since E_{TN} is a modification of E_{nCE} with the parameters *n* and *m* related to the outputs y_1 and y_2, respectively. Thus, we can take

$$b_2(\mathbf{x}) = g\left(h_m\left(1 - h_n^{-1}\left(g^{-1}(b_1(\mathbf{x}))\right)\right)\right). \quad (23)$$

By substituting (17) and (20) into (23), the relationship is given by

$$b_2(\mathbf{x}) = \frac{1 - \left[g^{-1}(b_1(\mathbf{x}))\right]^{-n/m}}{1 + \left[g^{-1}(b_1(\mathbf{x}))\right]^{-n/m}}. \quad (24)$$

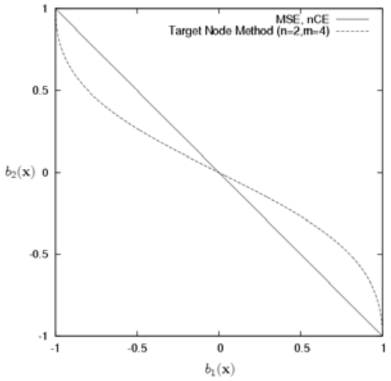


Fig. 1. $b_2(\mathbf{x})$ vs. $b_1(\mathbf{x})$ for MSE, *n*CE, and target node methods, respectively. $b_k(\mathbf{x})$ denotes the optimal solution of the *k*-th output in each method.

Table 1. Data set distribution of "Ann-thyroid13" for training and test.

	Minority Class	Majority Class	Total Samples	Minority Ratio [%]
Training	93	3488	3581	2.60
Test	73	3178	3251	2.25

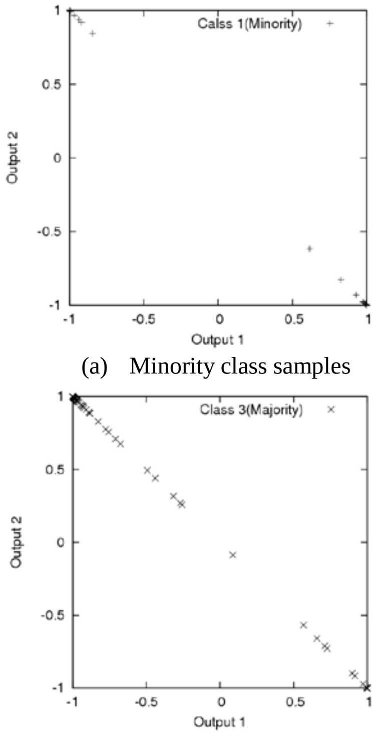
Fig. 1 shows the curves of (14) and (24) with the range of $-1 < b_k(\mathbf{x}) < +1$. For E_{MSE} and E_{nCE}, $b_2(\mathbf{x})$ vs. $b_1(\mathbf{x})$ is a straight line with a negative slope. On the contrary, E_{TN} has the curve of $b_2(\mathbf{x})$ vs. $b_1(\mathbf{x})$ with a steep slope at both ends of the horizontal axis. During the training of MLP based on E_{TN}, weights associated with y_1 is more strongly updated than weights associated with y_2. Therefore, after successful training of MLP, y_1 varies much less than y_2 near the desired vector points (+1,-1) and (-1,+1). This explanation coincides with the optimal curve for E_{TN}.

4. SIMULATIONS

The analyses are verified through simulations of "Ann-thyroid13" [4] data set. The "Ann-thyroid13" data was transformed from "Ann-thyroid" data [15], in which class 1 is the minority class while class 3 is treated as the majority class. Table 1 describes the data set distribution for training and test.

MLP consisting of 21 inputs, 16 hidden and 2 output nodes is trained for the "Ann-thyroid13" data using MSE, *n*CE, and the target node methods. The initial weights of MLP were drawn at random from a uniform distribution on $[-1\times10^{-4}, 1\times10^{-4}]$. Learning rates η's are derived so that $E\{\eta \mid \delta_k^{(p)} \mid\}$ has the same value in each method. As a result, learning rates of 0.006, 0.005, and 0.004 are used for the conventional EBP using MSE, *n*CE with *n*=4, and the target node method with *n*=2 and *m*=4, respectively. After training of 20,000 epochs, we plotted y_2 vs. y_1 by presenting test samples to each trained MLP.

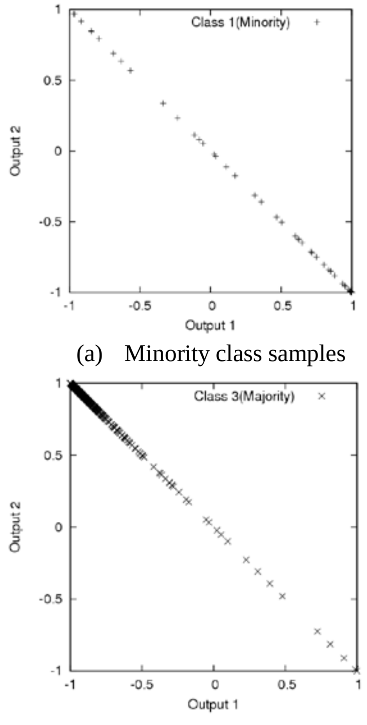
Fig. 2 shows the plots of MLP outputs trained with the MSE function. Fig. 2(a) corresponds to the test samples in the minority class whose desired point is T_1 at (+1,-1). Also, Fig. 2(b) corresponds to the test samples in the majority class whose desired point is T_2 at (-1,+1). All the points of Fig. 2 are on the line between T_1 and T_2, which coincides with the analysis result in Fig. 1. In the figures, the straight line from (-1,-1) to (+1,+1) is the decision line for classification based on the Max. rule. That is, samples in the area below the decision line is classified as C_1 and samples in the opposite area is classified as C_2. As shown in Fig. 2(a), the minority class samples below the decision line are correctly classified ones while those above the decision line are incorrectly classified ones. Also, at Fig. 2(b), the majority class samples above the decision line are correctly classified. Although the desired point



(a) Minority class samples

(b) Majority class samples

Fig. 2. Plots of MLP outputs trained with MSE function.



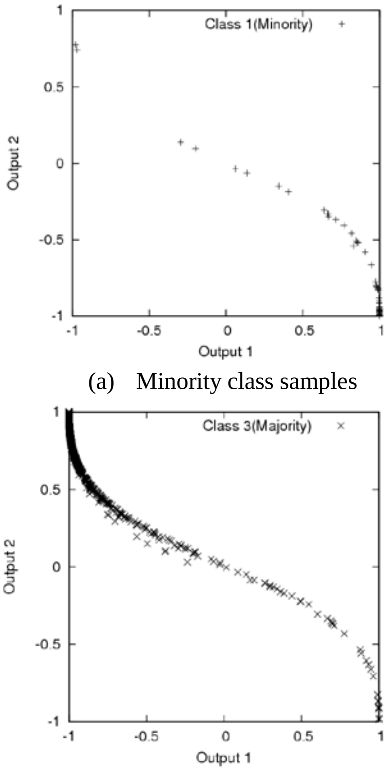
(a) Minority class samples

(b) Majority class samples

Fig. 3. Plots of MLP outputs trained with the n-th order extension of cross-entropy (*n*CE) error function (*n*=4).

of the minority class is T_1, there are some minority samples located very closely to T_2 (Fig. 2(a)) and these are the incorrectly saturated samples. As shown in Fig. 2(b), the majority samples very close to T_1 are incorrectly saturated, too.

Fig. 3 shows the plots of MLP outputs trained with the *n*CE error function. The points are on the straight line between T_1 and T_2, which coincides with the analysis result in Fig. 1. Comparing Fig. 2 with Fig. 3, the points in Fig. 2 are located more closely to T_1 or T_2 than the points in Fig. 3. This supports that MSE method has the weakness of over-fitting and *n*CE alleviates the degree of over-fitting [12]. Especially, the incorrectly saturated samples in Fig. 3 are less than the incorrectly saturated samples in Fig. 2. Thus, we can say that *n*CE method reduces the incorrect saturation of output nodes [12]. However, *n*CE cannot prevent that weights are mainly updated by the majority class samples.



(a) Minority class samples

(b) Majority class samples

Fig. 4. Plots of MLP outputs trained with the target node method (*n*=2, *m*=4).

Fig. 4 shows the plots of MLP outputs trained with the target node method. The points are on the curve having the same shape with the analysis result in Fig. 1. Comparing with Figs. 2(a) and 3(a), incorrectly saturated minority samples in Fig. 4(a) are much less. Also, the number of minority samples above the decision line is only four and the classification ratio of the minority class is 94.52%, the best among the comparison methods (Table 2). The target node method keeps the characteristic of *n*CE to prevent the incorrect saturation of output nodes. Also, by controlling the strength of error signal given by (10), the target node method can prevent the boundary distortion and improve the classification of minority class. Table 2 shows the classification ratio of test samples in each method. As expected, classification ratios of the minority class in MSE and *n*CE methods are around eighty percent. In the target node method, on the contrary, the classification ratio of the minority class is much improved without severe degradation of the majority class classification ratio.

Table 2. Classification ratio of test samples [%].

	MSE	nCE	Target Node
Minoroty	82.19	80.82	94.52
Majority	99.28	99.62	98.80

5. CONCLUSION

In this paper, we considered the optimal outputs of feed-forward neural network classifier trained for the imbalanced data. Through statistical analyses, we derived the relationship between the two optimal outputs of neural network classifier. The derived results coincided with the plots through simulations of “Ann-thyroid” data.

By plotting outputs of the neural network classifier trained with the MSE, we verified that the classifier was over-fitted and some outputs were incorrectly saturated. In the case of *n*CE, the output plots showed that the over-fitting and incorrect saturation were alleviated. When the classifier was trained with the target node method, the minority target node varies much less than the majority target node near the target points. This characteristic prevented the boundary distortion problem and improved the classification of interesting minority class samples.

REFERENCES

[1] Y. Sun, M. S. Kamel, A. K. C. W, and Y. Wang, "Cost-Sensitive Boosting for Classification of Imbalanced Data," Pattern Recognition, vol.40, 2007, pp. 3358-3378.

[2] F. Provost and T. Fawcett, "Robust Classification for Imprecise Environments," Machine Learning, vol.42, 2001, pp. 203-231.

[3] N. V. Chawla, K. W. Bowyer, L. O. all, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," J. Artificial Intelligence Research, vol.16, 2002, pp. 321-357.

[4] P. Kang and S. Cho, "EUS SVMs: ensemble of under-sampled SVMs for data imbalance problem, " *Proc. ICONIP’06*, 2006, p. 837-846.

[5] Y.-M. Huang, C.-M. Hung, and H. C. Jiau, "Evaluation of Neural Networks and Data Mining Methods on a Credit Assessment Task for Class Imbalance Problem," Nonlinear Analysis, vol.7, 2006, pp. 720-747.

[6] N. V. Chawla, D. A. Cieslak, L. O. Hall, and A. Joshi, "Automatically Countering Imbalance and Its Empirical Relationship to Cost," Data Mining and Knowledge Discovery, vol.17, no.2, 2008, pp. 225-252.

[7] Z.-H. Zhou and X.-Y. Liu, "Training Cost-Sensitive Neural Networks with Methods Addressing the Class Imbalance Problem," IEEE Trans. Know. and Data Eng., vol.18, no. 1, Jan. 2006, pp. 63-77.

[8] H. Zhao, "Instance Weighting versus Threshold Adjusting for Cost-Sensitive Classification," Knowledge and Information Systems, vol.15, 2008, pp. 321-334.

[9] L. Bruzzone and S. B. Serpico, "Classification of Remote-Sensing Data by Neural Networks," Pattern Recognition Letters, vol.18, 1997, pp. 1323-1328.

[10] D. E. Rumelhart and J. L. McClelland, *Parallel Distributed Processing*, Cambridge, MA, 1986.

[11] Y. Lee, S.-H. Oh, and M. W. Kim,"An Analysis of Premature Saturation in Back-Propagation Learning," Neural Networks, vol.6, 1993, pp. 719-728.

[12] S.-H. Oh, "Improving the Error Back-Propagation Algorithm with a Modified Error Function," IEEE Trans. Neural Networks, vol.8, 1997, pp. 799-803.

[13] S.-H. Oh, "Classification of Imbalanced Data Using Multilayer Perceptrons," J. Korea Contents Association, vol.9, no.4, July 2009, pp.141-148.

[14] H. White, "Learning in Artificial Neural Networks: A Statistical Perspective," Neural Computation, vol.1, no.4, Winter 1989, pp.425-464.

[15] A. Frank and A. Asuncion, *UCI Machine Learning Repository*, University of California, Irvine, School of Information and Computer Sciences http://archive.ics.uci.edu/ml, 2010.

Sang-Hoon Oh
received his B.S. and M.S degrees in Electronics Engineering from Pusan National University in 1986 and 1988, respectively. He received his Ph.D. degree in Electrical Engineering from Korea Advanced Institute of Science and Technology in 1999. From 1988 to 1989, he worked for LG semiconductor, Ltd., Korea. From 1990 to 1998, he was a senior research staff in Electronics and Telecommunication Research Institute (ETRI), Korea. From 1999 to 2000, he was with Brain Science Research Center, KAIST. In 2000, he was with Brain Science Institute, RIKEN, Japan, as a research scientist. In 2001, he was an R&D manager of Extell Technology Corporation, Korea. Since 2002, he has been with the Department of Information Communication Engineering, Mokwon University, Daejon, Korea, and is now an associate professor. Also, he was with the Division of Computational Science and Engineering, Georgia Institute of Technology, Atlanta, USA, as a visiting scholar from August 2008 to August 2009. His research interests are machine learning, speech signal processing, pattern recognition, and bioinformatics.