|  |
| --- |
| ***Optimal Solutions for SoftMax Outputs*** |
| Sang-Hoon Oh\*  \*Mokwon University, Korea  E-mail:ohsanghoon16@gmail.com |
|  |

# Introduction

We use FNNs(Feedforward Neural Networks) in many applications based on the proof that FNN is a universal approximator which can approximate any function with enough number of hidden nodes[1]. Rumelhart and McClelland proposed the EBP(Error Back-Propagation) algorithm to train FNNs[2]. However, the EBP algorithm to minimize MSE(mean-squared error function) of FNNs has a weakness with slow learning convergence and poor generalization performance[3]. There are many error functions to improve the performance of EBP algorithm for sigmoidal outputs[4].

Especially, DNNs(deep neural networks) adopt the cross-entropy error function with softmax outputs. Although there have been research results regrading optimal solutions of various error functions with sigmidal output, there is not the derivation of optimal solutions of the cross-entropy error function with softmax outputs. In this paper, we derive the optimal solutions of the cross-entropy error function.

# Feedforward Neural Networks (FNNs)

As shown in Fig. 1, FNN consists of an input vector **x**, a hidden node vector **h**, an output node vector **y**, and their connection weights. When an input vector  is presented to the FNN, a weighted sum to is given by and then the hidden node value is given by . Here,  is a bias and  is a weight between and. The *k*-th output node  is calculated through the same procedure of weighted sum and sigmoid transform using the weight and the hidden node value . When  is given for a specific training sample, we usually updates weights  and  to minimize the MSE function . Here, *P* is the number of training samples and *M* is the number of output nodes. EBP algorithm provides updating procedure of  and as follows[2]:

 (1)

 (2)

In classifications, we adopt the one-hot coding of the target value as follows:

(3)

In the limit that the number of training samples goes to infinity, the minimizer of for whole training samples converges (under certain regularity conditions, Theorem 1 in [5]) towards the minimizer of the function

, (4)

where is the expectation operator, is the random variable denoting the desired value, and is the random vector denoting an input sample. Since the desired values are coded as Eq. (3), the optimal solutions of output nodes [in the space of all functions taking values in (0,1)] to minimize Eq. (4) are given by

(5)

where is the posterior probability [3][4][5]

# Optimal Solutions of Cross-Entropy Error with SoftMax Output

In DNNs(deep neural networks) as shown in Fig. 2, we use the softmax output given by

(6)

where is the weighted sum to the output node[6][7]. For applying DNNs to classification probems, we usually use the cross-entropy error function given by

(7)

To derive the optimal solution of Eq. (7), we propose a new probabilistic coding of the target node. Then, we derive the optimal solutions regarding whether the output node is the target node or not. Here, we use the term ‘target node’ to denote that the output node identifies the correct classification. Also, we plot the optimal solutions and interpret the physical meaning of them.

# 4. Discussion and Conclusion

In this presentation, we briefly introduced FNN and derive optimal solutions of cross-entropy error fucntion with softmax output nodes.

5. References

[1] K. Hornik, M. Stinchcombe, and H. White, “Multilayer Feed-forward Networks are Universal Approximators,”

Neural Networks, vol.2, 1989, pp. 359-366.

[2] D. E. Rumelhart and J. L. McClelland, Parallel Distributed Processing, MIT Press, Cambridge, MA, 1986.

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

[4] S.-H. Oh, “Statistical Analyses of Various Error Functions for Pattern Classifiers,” Proc. Int. Conf. Hybrid Information Tech., Sept. 22-24, Daejon, Korea, 2011, vol. 206, pp. 129-133.

[5] H. White, “Learning in artificial neural networks: A statistical perspective,” *Neural Computation,* 1989*,* **1**, pp. 425-464

[6] B. Chen, W. Deng, and J. Du, “Noisy softmax: improving the generalization ability of DCNN via postponing the early softmax saturation,” The IEEE Conf. Computer Vis. Patt. Recog., Honolulu, Hawaii, July 2017, pp. 5372-5381.

[7] S. Horiguchi, D. Ikami, and K. Aizawa, “Significance of softmax-based features in comparison to distance metric learning-based features,” Accepted in *IEEE Trans. Pattern Anal. Machine Intell.,* 2019, doi: 10.1109/TPAMI.2019.2911075

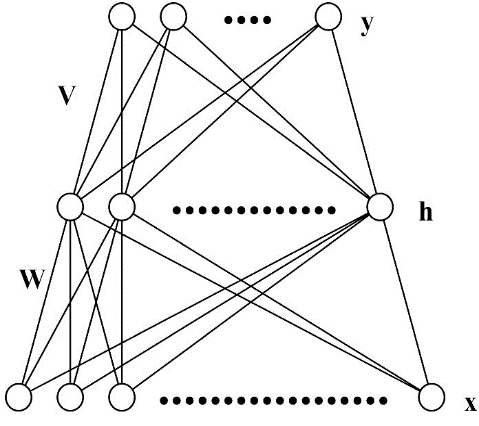


Figure 1 The architecture of FNN.

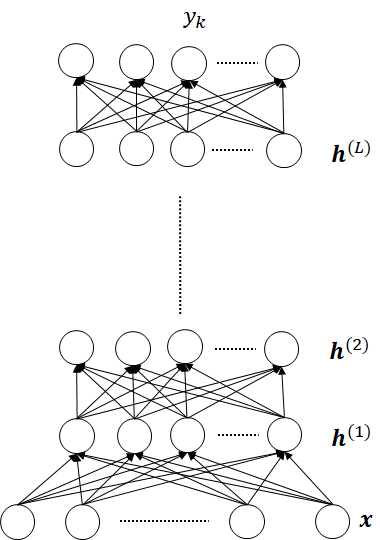


Figure 2. The architecture of Deep Neural Networks.