

# Natural Computing Series

Series Editors: G. Rozenberg  
Th. Bäck A.E. Eiben J.N. Kok H.P. Spaink



Leiden Center for Natural Computing

---

Advisory Board: S. Amari G. Brassard K.A. De Jong C.C.A.M. Gielen T. Head  
L. Kari L. Landweber T. Martinetz Z. Michalewicz M.C. Mozer E. Oja  
G. Păun J. Reif H. Rubin A. Salomaa M. Schoenauer H.-P. Schwefel C. Torras  
D. Whitley E. Winfree J.M. Zurada

For further volumes:  
<http://www.springer.com/series/4190>

Daniel S. Yeung · Ian Cloete · Daming Shi ·  
Wing W.Y. Ng

# Sensitivity Analysis for Neural Networks

 Springer

### *Authors*

Prof. Daniel S. Yeung  
School of Computer Science  
and Engineering  
South China University of Technology  
Wushan Rd.  
TianHe District  
Guangzhou, China  
danyeung@ieee.org

Prof. Daming Shi  
School of Electrical Engineering and Computer  
Science  
Kyungpook National University  
Buk-gu, Daegu  
South Korea  
asdmsi@ntu.edu.sg

Prof. Ian Cloete  
President  
Campus 3  
International University in Germany  
76646 Bruchsal, Germany  
ian.cloete@i-u.de; president@i-u.de

Dr. Wing W.Y. Ng  
School of Computer Science  
and Engineering  
South China University of Technology  
Wushan Rd.  
TianHe District  
Guangzhou, China  
wingng@ieee.org

### *Series Editors*

G. Rozenberg (Managing Editor)  
rozenber@liacs.nl

Th. Bäck, J.N. Kok, H.P. Spaijk  
Leiden Center for Natural Computing  
Leiden University  
Niels Bohrweg 1  
2333 CA Leiden, The Netherlands

A.E. Eiben  
Vrije Universiteit Amsterdam  
The Netherlands

ISSN 1619-7127

ISBN 978-3-642-02531-0

e-ISBN 978-3-642-02532-7

DOI 10.1007/978-3-642-02532-7

Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2009938187

ACM Computing Classification (1998): I.2.6, F.1.1

© Springer-Verlag Berlin Heidelberg 2010

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilm or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

The use of general descriptive names, registered names, trademarks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

*Cover design:* KuenkelLopka GmbH

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

# Preface

Neural networks provide a way to realize one of our human dreams to make machines think like us. Artificial neural networks have been developed since Rosenblatt proposed the Perceptron in 1958. Today, many neural networks are not treated as black boxes any more. Issues such as robustness and generalization abilities have been brought to the fore. The advances in neural networks have led to more and more practical applications in pattern recognition, financial engineering, automatic control and medical diagnosis, to name just a few.

Sensitivity analysis dates back to the 1960s, when Widrow investigated the probability of misclassification due to weight perturbations, which are caused by machine imprecision and noisy input. For the purpose of analysis, these perturbations can be simulated by embedding disturbance into the original inputs or connection weights. The initial idea of sensitivity analysis was then extended to optimization and to applications of neural networks, such as sample reduction, feature selection, active learning and critical vector learning.

This text should primarily be of interest to graduate students, academics, and researchers in branches of neural networks, artificial intelligence, machine learning, applied mathematics and computer engineering where sensitivity analysis of neural networks and related concepts are used. We have made an effort to make the book accessible to such a cross-disciplinary audience.

The book is organized into eight chapters, of which Chap. 1 gives an introduction to the various neural network structures and learning schemes. A literature review on the methodologies of sensitivity analysis is presented in Chap. 2. Different from the traditional hypersphere model, the hyper-rectangle model described in Chap. 3 is especially suitable for the most popular and general feedforward network: the multilayer Perceptron. In Chap. 4, the activation function is also involved in the calculation of the sensitivity analysis by parameterizing. The sensitivity analysis of radial basis function networks is discussed in Chaps. 5 and 6, with the former giving a generalization error model whereas the latter concerns optimizing the hidden neurons. In Chap. 7, sensitivity is measured in order to encode prior knowledge into a neural network. In Chap. 8, sensitivity analysis is applied in many applications, such as dimensionality reduction, network optimization and selective learning.

We would like to express our thanks to many colleagues, friends and students who provided reviews of different chapters of this manuscript. They include Minh

Nhut Nguyen, Xiaoqin Zeng, Patrick Chan, Xizhao Wang, Fei Chen and Lu He. We often find ourselves struggling with many competing demands for our time and effort. As a result, our families, especially our beloved spouses, are the ones who suffer the most. We are delighted to dedicate this work to Foo-Lau Yeung, Wilma Cloete and Jian Liu.

It is with great humility that we would like to acknowledge our Good Lord as the true creator of all knowledge. This work is the result of our borrowing a small piece of knowledge from Him.

8 July 2009

Daniel S. Yeung  
Ian Cloete  
Daming Shi  
Wing W.Y. Ng

# Contents

<b>1</b>	<b>Introduction to Neural Networks</b>	1
1.1	Properties of Neural Networks	3
1.2	Neural Network Learning	5
1.2.1	Supervised Learning	5
1.2.2	Unsupervised Learning	5
1.3	Perceptron	6
1.4	Adaline and Least Mean Square Algorithm	8
1.5	Multilayer Perceptron and Backpropagation Algorithm	9
1.5.1	Output Layer Learning	11
1.5.2	Hidden Layer Learning	11
1.6	Radial Basis Function Networks	12
1.7	Support Vector Machines	13
<b>2</b>	<b>Principles of Sensitivity Analysis</b>	17
2.1	Perturbations in Neural Networks	17
2.2	Neural Network Sensitivity Analysis	18
2.3	Fundamental Methods of Sensitivity Analysis	21
2.3.1	Geometrical Approach	21
2.3.2	Statistical Approach	23
2.4	Summary	24
<b>3</b>	<b>Hyper-Rectangle Model</b>	25
3.1	Hyper-Rectangle Model for Input Space of MLP	25
3.2	Sensitivity Measure of MLP	26
3.3	Discussion	27
<b>4</b>	<b>Sensitivity Analysis with Parameterized Activation Function</b>	29
4.1	Parameterized Antisymmetric Squashing Function	29
4.2	Sensitivity Measure	30
4.3	Summary	31
<b>5</b>	<b>Localized Generalization Error Model</b>	33
5.1	Introduction	33
5.2	The Localized Generalization Error Model	35
5.2.1	The Q-Neighborhood and Q-Union	36

- 5.2.2 The Localized Generalization Error Bound . . . . . 36
- 5.2.3 Stochastic Sensitivity Measure for RBFNN . . . . . 38
- 5.2.4 Characteristics of the Error Bound . . . . . 40
- 5.2.5 Comparing Two Classifiers Using the Error Bound . . . . . 42
- 5.3 Architecture Selection Using the Error Bound . . . . . 42
  - 5.3.1 Parameters for MC<sup>2</sup>SG . . . . . 44
  - 5.3.2 RBFNN Architecture Selection Algorithm for MC<sup>2</sup>SG . . . . . 44
  - 5.3.3 A Heuristic Method to Reduce the Computational Time for MC<sup>2</sup>SG . . . . . 45
- 5.4 Summary . . . . . 45
- 6 Critical Vector Learning for RBF Networks . . . . . 47**
  - 6.1 Related Work . . . . . 47
  - 6.2 Construction of RBF Networks with Sensitivity Analysis . . . . . 48
    - 6.2.1 RBF Classifiers’ Sensitivity to the Kernel Function Centers . . . . . 49
    - 6.2.2 Orthogonal Least Square Transform . . . . . 51
    - 6.2.3 Critical Vector Selection . . . . . 52
  - 6.3 Summary . . . . . 52
- 7 Sensitivity Analysis of Prior Knowledge . . . . . 55**
  - 7.1 KBANNs . . . . . 56
  - 7.2 Inductive Bias . . . . . 58
  - 7.3 Sensitivity Analysis and Measures . . . . . 59
    - 7.3.1 Output-Pattern Sensitivity . . . . . 59
    - 7.3.2 Output-Weight Sensitivity . . . . . 60
    - 7.3.3 Output-H Sensitivity . . . . . 61
    - 7.3.4 Euclidean Distance . . . . . 61
  - 7.4 Promoter Recognition . . . . . 61
    - 7.4.1 Data and Initial Domain Theory . . . . . 62
    - 7.4.2 Experimental Methodology . . . . . 63
  - 7.5 Discussion and Conclusion . . . . . 64
- 8 Applications . . . . . 69**
  - 8.1 Input Dimension Reduction . . . . . 69
    - 8.1.1 Sensitivity Matrix . . . . . 70
    - 8.1.2 Criteria for Pruning Inputs . . . . . 70
  - 8.2 Network Optimization . . . . . 71
  - 8.3 Selective Learning . . . . . 74
  - 8.4 Hardware Robustness . . . . . 76
  - 8.5 Measure of Nonlinearity . . . . . 77
  - 8.6 Parameter Tuning for Neocognitron . . . . . 78
    - 8.6.1 Receptive Field . . . . . 79
    - 8.6.2 Selectivity . . . . . 80
    - 8.6.3 Sensitivity Analysis of the Neocognitron . . . . . 80
- Bibliography . . . . . 83**

# Chapter 1

## Introduction to Neural Networks

The human brain consists of ten billion densely interconnected nerve cells, called *neurons*; each connected to about 10,000 other neurons, with 60 trillion connections, *synapses*, between them. By using multiple neurons simultaneously, the brain can perform its functions much faster than the fastest computers in existence today. On the other hand, a neuron can be considered as a basic information-processing unit, whereas our brain can be considered as a highly complex, nonlinear and parallel biological information-processing network, in which information is stored and processed simultaneously. Learning is a fundamental and essential characteristic of biological neural networks. The ease with which they can learn led to attempts to emulate a biological neural network in a computer.

In the 1940s, McCulloch and Pitts proposed a model for biological neurons and biological neural networks. A stimulus is transmitted from dendrites to a soma via synapses, and axons transmit the response of one soma to another, as shown in Fig. 1.1. Inspired by the mechanism for learning in biological neurons, artificial neurons and artificial neural networks can perform arithmetic functions, with cells corresponding to neurons, activations corresponding to neuronal firing rates, connections corresponding to synapses, and connection weights corresponding to synaptic strengths, as shown in Fig. 1.1. The analogy between biological neurons and artificial neurons is made in Table 1.1. However, neural networks are far too simple to serve as realistic brain models on the cell level, but they might serve as very good models for the essential information processing tasks that organisms perform. This remains an open question because we have so little understanding of how the brain actually works (Gallant, 1993).

In a neural network, *neurons* are joined by directed arcs – *connections*. The neurons and arcs constitute the network *topology*. Each arc has a numerical *weight* that specifies the influence between two neurons. Positive weights indicate reinforcement; negative weights represent inhibition. The weights determine the behavior of the network, playing somewhat the same role as in a conventional computer program. Typically, there are many *inputs* for a single neuron, and a subsequent *output* of an *activation function* (or *transfer function*). Some frequently used activation functions include: