

**MATHEMATICAL MODELLING OF HIDDEN LAYER
ARCHITECTURE IN ARTIFICIAL NEURAL NETWORKS**

Nimalka Mihirini Wagarachchi

(118024B)

Degree of Doctor of Philosophy

Department of Computational Mathematics

University of Moratuwa

Sri Lanka

August 2018

MATHEMATICAL MODELLING OF HIDDEN LAYER ARCHITECTURE IN ARTIFICIAL NEURAL NETWORKS

Nimalka Mihirini Wagarachchi

(118024B)

Thesis submitted in partial fulfillment of the requirements for the degree Doctor of
Philosophy

Department of Computational Mathematics

University of Moratuwa

Sri Lanka

August 2018

Declaration

I declare that this is my own work and this thesis does not incorporate without acknowledgement any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief, it does not contain any material previously published or written by another person except where the acknowledgement made is made in the text.

Also, I hereby grant to University of Moratuwa the non-exclusive right to reproduce and distribute my thesis, in whole or part in print, electronic or other medium. I retain the right to use this content in whole or part in future works (such as articles or books)

Signature:

Date:

The above candidate has carried out the research for the PhD thesis under my supervision.

Signature of the supervisor:

Date:

Dedicated to

My beloved Father and Mother

Acknowledgements

Many people have helped their best to successfully completion of this research. I acknowledge all of them for their valuable thoughts and constant encouragement given me to make my project a reality.

First and foremost, I acknowledge my supervisor Senior Professor Asoka Karunananda for accepting me as his research student and giving excellent support and advices. Prof. Karunananda is a great mentor who guided whilst giving me all the freedom and encouragement to accompany with my own ideas. Without his patient listening and creative thoughts this work would not have been possible at all.

Very special thank goes to Prof. Sarath Pieris and Dr. Uditha Rathnayake for their invaluable comments and guidance as my examiners of bi-annual review panels.

I acknowledge all the office bearers and the staff of the HETC project for granting me financial assistance by awarding the HETC scholarship to smooth functioning of this research. Also express my sincere thanks to the staff of OTS office, university of Ruhuna for their wholehearted support.

I graciously acknowledge the support of Senior Prof. Gamini Senanayake, The Vice Chancellor, Senior Prof. Susirith Mendis, former Vice Chancellor, Dr. Nayana Alagiyawanna, former Dean/ Faculty of Engineering and the present Deputy Vice Chancellor University of Ruhuna for selecting me as an HETC candidate of the University of Ruhuna and giving their utmost support and guidance throughout.

I wish to extend my sincere thanks for the support I received from all the members of the administration office and members of the Faculty of Information Technologies, University of Moratuwa. Especially I thank Ms. Dilini Kulawansa, Dr. Subha Fernando and Dr. Thushari Silva for their important roles.

Also, I thank all the academic and non-academic staff of Faculty of Engineering, University of Ruhuna for their kind-hearted help to fulfill my research work.

My graciously acknowledgment to the friendly assistance given by Dr. M. K. Abeyrathne, Dr. Subashi, Ms. Malkanthi, Mr. Samantha and all my colleagues of the Department of Interdisciplinary Studies, Faculty of Engineering, University of Ruhuna.

Very special and heartfelt thanks for Budditha and Chinthanie for their gracious associations throughout the last couple of years.

I acknowledge the sacrificial dedication of my family members, especially my husband Pramud and our daughter Dinithi Navodya for their encouragement and corporate by managing all the works while I was busy with my works on this research.

Abstract

The performance of an Artificial Neural Network (ANN) strongly depends on its hidden layer architecture. The generated solution by an ANN does not guarantee that it has always been devised with the simplest neural network architecture suitable for modeling the particular problem. This results in computational complexity of training of an ANN, deployment, and usage of the trained network. Therefore, modeling the hidden layer architecture of an ANN remains as a research challenge. This thesis presents a theoretically-based approach to prune hidden layers of trained artificial neural networks, ensuring better or the same performance of a simpler network as compared with the original network.

The method described in the thesis is inspired by the finding from neuroscience that the human brain has a neural network with nearly 100 billion neurons, yet our activities are performed by a much simpler neural network with a much lesser number of neurons. Furthermore, in biological neural networks, the neurons which do not significantly contribute to the performance of the network will naturally be disregarded. According to neuroplasticity, biological neural networks can also solicit activations of neurons in the proximity of the active neural network to improve the performance of the network. On the same token, it is hypothesized that for a given complex-trained ANN, we can discover an ANN, which is much more simplified than the original given architecture.

This research has discovered a theory to reduce certain number of hidden layers and to eliminate disregarding neurons from the remaining hidden layers of a given ANN architecture. The procedure begins with a complex neural network architecture trained with backpropagation algorithm and reach to the optimum solution by two phases. First, the number of hidden layers is determined by using a peak search algorithm discovered by this research. The newly discovered simpler network with lesser number of hidden layers and highest generalization power considered for pruning of its hidden neurons. The pruning of neurons in the hidden layers has been theorized by identifying the neurons, which give least contribution to the network performances. These neurons are identified by detecting the correlations regarding minimization of error in training. Experiments have shown that the simplified network architecture generated by this approach exhibits same or better performance as compared with the original large network architecture. Generally, it reduces more than 80% of neurons while increasing the generalization by about 30%. As such, the proposed approach can be used to discover simple network architecture relevant to a given complex architecture of an ANN solution. Due to its architectural simplicity, the new architecture has been computationally efficient in training, usage and further training.

Keywords: Artificial neural networks, backpropagation algorithm, delta value, hidden layer architecture, neuroplasticity

CONTENTS

CHAPTER 1 – INTRODUCTION	1
1.1 Prolegomena	1
1.2 Aims and Objectives	1
1.3 Background and Motivation	2
1.4 The Problem in Brief	3
1.5 Current Approaches to Modelling Hidden Layer Architecture	4
1.6 The Proposed Solution	5
1.7 Resource requirements	7
1.8 Organization	7
1.9 Summary	8
CHAPTER 2 – FUNDAMENTALS OF ARTIFICIAL NEURAL NETWORKS	9
2.1 Introduction	9
2.2 Preamble to the Artificial Neural Networks	9
2.3 The History	12
2.4 Structure of Artificial Neural Networks	13
2.4.1 Feedforward networks	15
2.4.2 Recurrent networks	18
2.5 Activation Functions	19
2.5.1 Hard limit activation functions	19
2.5.2 Linear function	20
2.5.3 Sigmoid functions	21
2.6 Neural Network Learning	22
2.6.1 Supervised learning	22
2.6.2 Reinforcement learning	22
2.6.3 Unsupervised learning	23
2.7 Learning Algorithms	24
2.7.1 Hebbian learning	24
2.7.2 Error correction learning rules	26

2.7.2.1	Perceptron Learning Rule	26
2.7.2.2	Backpropagation Learning	28
2.7.3	Boltzmann learning	33
2.7.4	Competitive learning	34
2.8	Summary	34
CHAPTER 3 – CHALLENGES IN DESIGNING OF NEURAL NETWORKS		36
3.1	Introduction	36
3.2	The Problem of Designing the Optimal Architecture in ANN	36
3.3	Pruning Algorithms	37
3.3.1	Sensitivity calculation method	38
3.3.2	Penalty methods	45
3.4	Constructive methods	49
3.4.1	Cascade correlation algorithm	49
3.4.2	Dynamic node creation algorithm	51
3.4.3	Tiling algorithm	51
3.4.4	Tower algorithm	52
3.4.5	Pyramid algorithm	53
3.5	Evolutionary Methods	54
3.6	Summary	55
CHAPTER 4 – A THEORETICAL BASIS FOR MODELING HIDDEN LAYERS		56
4.1	Introduction	56
4.2	The History of the Neuroplasticity	56
4.3	Types of Neuroplasticity	58
4.3.1	Activity - dependent plasticity	58
4.3.2	Competitive plasticity	59
4.3.3	Positive and negative plasticity	60
4.4	Structure of the Biological Neuron	61
4.5	Neuronal Structure of the Human Brain	62
4.6	The Anatomy of the Human Brain	63
4.7	Functions of the Neocortex	66

4.8	Classification of Effect of Neuroplasticity	69
4.8.1	Structural changes in Human brain	69
4.8.1.1	Neurogenesis	70
4.8.1.2	Neural Migration	71
4.8.1.3	Neural Cell Death	71
4.8.2	Synaptic plasticity	72
4.8.2.1	Synaptogenesis and Synaptic Pruning	72
4.8.3	Functional neuroplasticity	73
4.9	Positive and Negative Outcomes of Neuroplasticity	74
4.9.1	Positive outcomes of neuroplasticity	74
4.9.2	Negative outcomes of neuroplasticity	74
4.10	Artificial Neural Networks and Human Brain	76
4.11	Summary	78
CHAPTER 5 – A NOVEL APPROACH TO MODELLING HIDDEN LAYERS		79
5.1	Introduction	79
5.2	The Hypothesis	79
5.3	Inputs	80
5.4	Outputs	80
5.5	Process of the New Method	81
5.5.1	The Peak Search Algorithm	81
5.5.2	Performance of the algorithm	90
5.5.3	Upper limit for the hidden Layers	91
5.5.4	Determining number of hidden neurons	94
5.5.5	Merge the similar neurons	97
5.5.6	The new algorithm	98
5.6	Summary	100
CHAPTER 6 – EXPERIMENTAL DESIGN AND RESULTS		101
6.1	Introduction	101
6.2	Experimental Design	101
6.2.1	Experimental setup	101
6.2.2	Test cases	104
6.2.2.1	Breast Cancer Wisconsin data set (Cancer)	104
6.2.2.2	Credit card approval data set (Card)	104
6.2.2.3	Pima Indians diabetes data set (Diabetes)	105

6.2.2.4	Solar flare data set (Flare)	105
6.2.2.5	User knowledge modeling data set (Knowledge)	105
6.2.3	Testing strategies	105
6.3	Experimental Results	106
6.3.1	The variation of network performance with the number of layers.	106
6.3.2	Determining the number of hidden layers	113
6.3.3	Correlation between the sum of Delta values and the output error	121
6.3.4	Correlation between the sum of Delta values and the output error	121
6.3.5	Removing neurons	126
6.4	Comparison with Existing Method	132
6.5	Summary	134
CHAPTER 7 – USING PSDV FOR DEEP NEURAL NETWORKS		135
7.1	Introduction	135
7.2	Preamble to Deep Neural Networks	135
7.2.1	Convolutional Neural Networks	137
7.2.2	Deep Belief Neural Networks	139
7.3	Using PSDV for Deep Neural Networks	140
7.3.1	Applying PSDV to Convolutional Neural Networks	140
7.3.2	Applying PSDV to Deep Belief Neural Networks	141
7.4	Summary	142
CHAPTER 8 – CONCLUSION AND FUTURE WORKS		143
8.1	Introduction	143
8.2	Modelling Hidden Layer Architecture in ANN	143
8.3	Objectives-wise Achievement	144
8.4	Limitations and Future Directions	146
8.5	Summary	147
REFERENCES		148
APPENDIX A – DATA SETS		160
APPENDIX B – DETERMINING THE NUMBER OF HIDDEN LAYERS		165

APPENDIX C – SELECTED CODES

177

APPENDIX D – PUBLICATIONS

182

List of Figures

Figure 1.1: Diagram representation of central nervous system.....	3
Figure 1.2: The structure of the human brain	7
Figure 2.1: Model of ANN proposed by McCulloch and Pitt.....	12
Figure 2.2: Single layer feedforward network.....	16
Figure 2.3: Multilayer feed forward network	17
Figure 2.4: Radial basis function network.....	18
Figure 2.5: Recurrent network.....	18
Figure 2.6: Classification of Neural Networks by architecture	19
Figure 2.7: Hard limit function.....	20
Figure 2.8: Linear function.....	20
Figure 2.9: Sigmoid functions	21
Figure 2.10: Block diagram for supervised learning	23
Figure 2.11: Block diagram for unsupervised learning	24
Figure 2.12: Perceptron Algorithm.....	27
Figure 2.13: The perceptron learning rule	28
Figure 2.14: Backpropagation learning algorithm.....	32
Figure 2.15: Boltzmann Machine	34
Figure 3.1: Adaptive linear neuron.....	44
Figure 3.2: Structure of the cascade algorithm.....	50
Figure 3.3: Tilling Algorithm	52
Figure 3.4: Tower Algorithm	52
Figure 3.5: Pyramid Algorithm	54
Figure 4.1: Structure of a biological nerve cell	57
Figure 4.2: Structure of a neurons and a synapse	62
Figure 4.3: Structure of the brain	64
Figure 4.4: The structural organization of levels in the brain	65
Figure 4.5: The hierarchy of the brain.....	66
Figure 4.6: The layered structure of the neocortex.....	69
Figure 4.7: Hippocampus area of the brain	70
Figure 4.8: Changes of synapses	73
Figure 4.9: Neurons of autistic (left) and normal brains (right)	76
Figure 5.1: Change of the generalization with the number of hidden layers	81
Figure 5.2: Change of generalization with hidden layers.....	82
Figure 5.3: Graphs for $gL > gm \geq gR$	84
Figure 5.4: Graphs for $gL \leq gm < gR$	84
Figure 5.5: Graphs when $gm1$ is the maximum.....	85
Figure 5.6: Graphs when $gm2$ is the maximum.....	86
Figure 5.7: Graphs when gm is the maximum	86
Figure 5.8: Flow diagram for peak search algorithm	88
Figure 5.9: The peak search algorithm	89
Figure 5.10: Binary comparison tree	90
Figure 5.11: Sigmoid functions	92
Figure 5.12: Sketch of the derivatives of sigmoid functions.....	92
Figure 5.13: xn for different initial values.....	94
Figure 5.14: Illustration of removing unimportant neurons	99
Figure 6.1: Changing performance with hidden layers in Cancer I problem	108

Figure 6.2: Generalization comparison of Cancer problems.....	108
Figure 6.3: Generalization comparison of Card problems	110
Figure 6.4: Generalization comparison of Diabetes problems	110
Figure 6.5: Generalization comparison of Flare problems.....	111
Figure 6.6: No. of epochs take to train Knowledge I Problem.....	112
Figure 6.7: Generalization comparison of Knowledge problems.....	113
Figure 6.8: Determining number of hidden layers in Cancer I.....	117
Figure 6.9: Determining number of hidden layers in Flare I problem	119
Figure 6.10: Correlations of Cancer I problem	122
Figure 6.11: Correlations of Card I problem	123
Figure 6.12: Correlation of the Banknote problem	124
Figure 6.13: Summary of Peak search algorithm	129
Figure 6.14: Reduction of neurons from the initial network configuration	131
Figure 6.15: Increase of the generalization comparing with the initial network.....	132
Figure 6.16: Comparison of PSDV with the other existing methods	133
Figure 7.1: Structure of a convolutional neural network.....	138

List of Tables

Table 2.1: Comparison of Von Neumann computer and the human brain.....	11
Table 6.1: Information of Data Sets	103
Table 6.2: Changing performance with hidden layers in Cancer problems	107
Table 6.3: Changing performance with hidden layers in Card problems	109
Table 6.4: Changing performance with hidden layers in Diabetes problems.....	109
Table 6.5: Changing performance with hidden layers in Flare problems	111
Table 6.6: Details of Initial networks	116
Table 6.7: Distribution of hidden neurons in Flare I data set	118
Table 6.8: Details of New architecture obtained by the Peak Search Algorithms ..	119
Table 6.9: Correlations of the Cancer I data set	121
Table 6.10: Correlations of the Card I data set.....	123
Table 6.11: Correlation between sum of delta values and output error.....	125
Table 6.12: Neural network architectures obtained by the new model	128
Table 6.13: Generalization of PSDV and other existing methods.....	133

Abbreviations

ADALINE – Adaptive linear neuron

AI – Artificial Intelligence

ANN – Artificial neural network

Bi-search algorithm – Binary search algorithm

CNS – Central nervous system

$\gamma_{\delta_h, E}$ – Correlation coefficient of the sum of the delta values of h^{th} hidden layer and the output error

etc. – etcetera

i.e. – That is

LTD – long term depression

LTP – long term potentiation

MADALINE – Many ADALINE

MBP – Magnitude based pruning

MLP – Multilayer perceptron

MRI – The Magnetic Resonance Imagine

NN – Neural network

OBD – Optimal brain damage

OBS – Optimal brain Surgeon

PNS – Peripheral nervous system

PSA – Peak search algorithm

PSDV – Peak search and delta value algorithm (The proposed algorithm)

RBF – Radial basis function

SOM – Self Organizing Map

SVZ – Sub ventricular zone