



Eighth edition
November 1996

©1996 The University of Amsterdam. Permission is granted to distribute *single* copies of this book for non-commercial use, as long as it is distributed as a whole in its original form, and the names of the authors and the University of Amsterdam are mentioned. Permission is also granted to use this book for non-commercial courses, provided the authors are notified of this beforehand.

The authors can be reached at:

Ben Kröse
Faculty of Mathematics & Computer Science
University of Amsterdam
Kruislaan 403, NL-1098 SJ Amsterdam
THE NETHERLANDS
Phone: +31 20 525 7463
Fax: +31 20 525 7490
email: krose@fwi.uva.nl
URL: <http://www.fwi.uva.nl/research/neuro/>

Patrick van der Smagt
Institute of Robotics and System Dynamics
German Aerospace Research Establishment
P. O. Box 1116, D-82230 Wessling
GERMANY
Phone: +49 8153 282400
Fax: +49 8153 281134
email: smagt@dlr.de
URL: <http://www.op.dlr.de/FF-DR-RS/>

Contents

Preface	9
I FUNDAMENTALS	11
1 Introduction	13
2 Fundamentals	15
2.1 A framework for distributed representation	15
2.1.1 Processing units	15
2.1.2 Connections between units	16
2.1.3 Activation and output rules	16
2.2 Network topologies	17
2.3 Training of artificial neural networks	18
2.3.1 Paradigms of learning	18
2.3.2 Modifying patterns of connectivity	18
2.4 Notation and terminology	18
2.4.1 Notation	19
2.4.2 Terminology	19
II THEORY	21
3 Perceptron and Adaline	23
3.1 Networks with threshold activation functions	23
3.2 Perceptron learning rule and convergence theorem	24
3.2.1 Example of the Perceptron learning rule	25
3.2.2 Convergence theorem	25
3.2.3 The original Perceptron	26
3.3 The adaptive linear element (Adaline)	27
3.4 Networks with linear activation functions: the delta rule	28
3.5 Exclusive-OR problem	29
3.6 Multi-layer perceptrons can do everything	30
3.7 Conclusions	31
4 Back-Propagation	33
4.1 Multi-layer feed-forward networks	33
4.2 The generalised delta rule	33
4.2.1 Understanding back-propagation	35
4.3 Working with back-propagation	36
4.4 An example	37
4.5 Other activation functions	38

4.6	Deficiencies of back-propagation	39
4.7	Advanced algorithms	40
4.8	How good are multi-layer feed-forward networks?	42
4.8.1	The effect of the number of learning samples	43
4.8.2	The effect of the number of hidden units	44
4.9	Applications	45
5	Recurrent Networks	47
5.1	The generalised delta-rule in recurrent networks	47
5.1.1	The Jordan network	48
5.1.2	The Elman network	48
5.1.3	Back-propagation in fully recurrent networks	50
5.2	The Hopfield network	50
5.2.1	Description	50
5.2.2	Hopfield network as associative memory	52
5.2.3	Neurons with graded response	52
5.3	Boltzmann machines	54
6	Self-Organising Networks	57
6.1	Competitive learning	57
6.1.1	Clustering	57
6.1.2	Vector quantisation	61
6.2	Kohonen network	64
6.3	Principal component networks	66
6.3.1	Introduction	66
6.3.2	Normalised Hebbian rule	67
6.3.3	Principal component extractor	68
6.3.4	More eigenvectors	69
6.4	Adaptive resonance theory	69
6.4.1	Background: Adaptive resonance theory	69
6.4.2	ART1: The simplified neural network model	70
6.4.3	ART1: The original model	72
7	Reinforcement learning	75
7.1	The critic	75
7.2	The controller network	76
7.3	Barto's approach: the ASE-ACE combination	77
7.3.1	Associative search	77
7.3.2	Adaptive critic	78
7.3.3	The cart-pole system	79
7.4	Reinforcement learning versus optimal control	80
III	APPLICATIONS	83
8	Robot Control	85
8.1	End-effector positioning	86
8.1.1	Camera-robot coordination is function approximation	87
8.2	Robot arm dynamics	91
8.3	Mobile robots	94
8.3.1	Model based navigation	94
8.3.2	Sensor based control	95

9	Vision	97
9.1	Introduction	97
9.2	Feed-forward types of networks	97
9.3	Self-organising networks for image compression	98
9.3.1	Back-propagation	99
9.3.2	Linear networks	99
9.3.3	Principal components as features	99
9.4	The cognitron and neocognitron	100
9.4.1	Description of the cells	100
9.4.2	Structure of the cognitron	101
9.4.3	Simulation results	102
9.5	Relaxation types of networks	103
9.5.1	Depth from stereo	103
9.5.2	Image restoration and image segmentation	105
9.5.3	Silicon retina	105
IV	IMPLEMENTATIONS	107
10	General Purpose Hardware	111
10.1	The Connection Machine	112
10.1.1	Architecture	112
10.1.2	Applicability to neural networks	113
10.2	Systolic arrays	114
11	Dedicated Neuro-Hardware	115
11.1	General issues	115
11.1.1	Connectivity constraints	115
11.1.2	Analogue vs. digital	116
11.1.3	Optics	116
11.1.4	Learning vs. non-learning	117
11.2	Implementation examples	117
11.2.1	Carver Mead's silicon retina	117
11.2.2	LEP's LNeuro chip	119
	References	123
	Index	131

List of Figures

2.1	The basic components of an artificial neural network.	16
2.2	Various activation functions for a unit.	17
3.1	Single layer network with one output and two inputs.	23
3.2	Geometric representation of the discriminant function and the weights.	24
3.3	Discriminant function before and after weight update.	25
3.4	The Perceptron.	27
3.5	The Adaline.	27
3.6	Geometric representation of input space.	29
3.7	Solution of the XOR problem.	30
4.1	A multi-layer network with l layers of units.	34
4.2	The descent in weight space.	37
4.3	Example of function approximation with a feedforward network.	38
4.4	The periodic function $f(x) = \sin(2x)\sin(x)$ approximated with sine activation functions.	39
4.5	The periodic function $f(x) = \sin(2x)\sin(x)$ approximated with sigmoid activation functions.	40
4.6	Slow decrease with conjugate gradient in non-quadratic systems.	42
4.7	Effect of the learning set size on the generalization	44
4.8	Effect of the learning set size on the error rate	44
4.9	Effect of the number of hidden units on the network performance	45
4.10	Effect of the number of hidden units on the error rate	45
5.1	The Jordan network	48
5.2	The Elman network	49
5.3	Training an Elman network to control an object	49
5.4	Training a feed-forward network to control an object	50
5.5	The auto-associator network.	51
6.1	A simple competitive learning network.	58
6.2	Example of clustering in 3D with normalised vectors.	59
6.3	Determining the winner in a competitive learning network.	59
6.4	Competitive learning for clustering data.	61
6.5	Vector quantisation tracks input density.	62
6.6	A network combining a vector quantisation layer with a 1-layer feed-forward neural network. This network can be used to approximate functions from \mathbb{R}^2 to \mathbb{R}^2 , the input space \mathbb{R}^2 is discretised in 5 disjoint subspaces.	62
6.7	Gaussian neuron distance function.	65
6.8	A topology-conserving map converging.	65
6.9	The mapping of a two-dimensional input space on a one-dimensional Kohonen network.	66

6.10	Mexican hat	66
6.11	Distribution of input samples.	67
6.12	The ART architecture.	70
6.13	The ART1 neural network.	71
6.14	An example ART run.	72
7.1	Reinforcement learning scheme.	75
7.2	Architecture of a reinforcement learning scheme with critic element	78
7.3	The cart-pole system.	80
8.1	An exemplar robot manipulator.	85
8.2	Indirect learning system for robotics.	88
8.3	The system used for specialised learning.	89
8.4	A Kohonen network merging the output of two cameras.	90
8.5	The neural model proposed by Kawato <i>et al.</i>	92
8.6	The neural network used by Kawato <i>et al.</i>	92
8.7	The desired joint pattern for joints 1. Joints 2 and 3 have similar time patterns.	93
8.8	Schematic representation of the stored rooms, and the partial information which is available from a single sonar scan.	95
8.9	The structure of the network for the autonomous land vehicle.	95
9.1	Input image for the network.	100
9.2	Weights of the PCA network.	100
9.3	The basic structure of the cognitron.	101
9.4	Cognitron receptive regions.	102
9.5	Two learning iterations in the cognitron.	103
9.6	Feeding back activation values in the cognitron.	104
10.1	The Connection Machine system organisation.	113
10.2	Typical use of a systolic array.	114
10.3	The Warp system architecture.	114
11.1	Connections between M input and N output neurons.	115
11.2	Optical implementation of matrix multiplication.	117
11.3	The photo-receptor used by Mead.	118
11.4	The resistive layer (a) and, enlarged, a single node (b).	119
11.5	The LNeuro chip.	120

Preface

This manuscript attempts to provide the reader with an insight in artificial neural networks. Back in 1990, the absence of any state-of-the-art textbook forced us into writing our own. However, in the meantime a number of worthwhile textbooks have been published which can be used for background and in-depth information. We are aware of the fact that, at times, this manuscript may prove to be too thorough or not thorough enough for a complete understanding of the material; therefore, further reading material can be found in some excellent text books such as (Hertz, Krogh, & Palmer, 1991; Ritter, Martinetz, & Schulten, 1990; Kohonen, 1995; Anderson & Rosenfeld, 1988; DARPA, 1988; McClelland & Rumelhart, 1986; Rumelhart & McClelland, 1986).

Some of the material in this book, especially parts III and IV, contains timely material and thus may heavily change throughout the ages. The choice of describing robotics and vision as neural network applications coincides with the neural network research interests of the authors.

Much of the material presented in chapter 6 has been written by Joris van Dam and Anuj Dev at the University of Amsterdam. Also, Anuj contributed to material in chapter 9. The basis of chapter 7 was form by a report of Gerard Schram at the University of Amsterdam. Furthermore, we express our gratitude to those people out there in Net-Land who gave us feedback on this manuscript, especially Michiel van der Korst and Nicolas Maudit who pointed out quite a few of our goof-ups. We owe them many *kwartjes* for their help.

The seventh edition is not drastically different from the sixth one; we corrected some typing errors, added some examples and deleted some obscure parts of the text. In the eighth edition, symbols used in the text have been globally changed. Also, the chapter on recurrent networks has been (albeit marginally) updated. The index still requires an update, though.

Amsterdam/Oberpfaffenhofen, November 1996
Patrick van der Smagt
Ben Kröse

