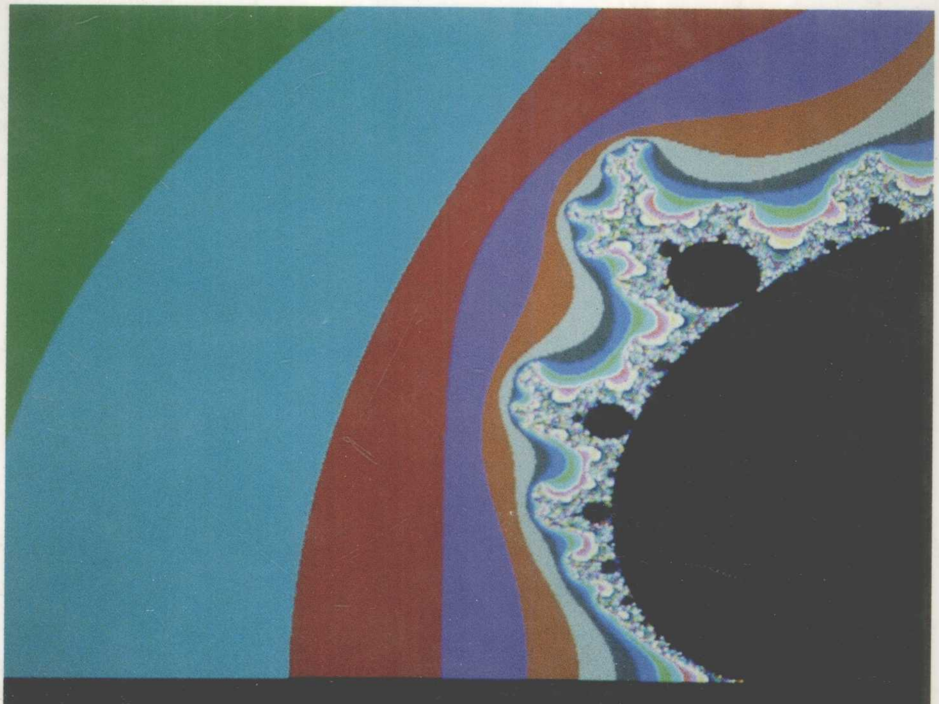
MODULAR LEARNING IN NEURAL NETWORKS

A Modularized Approach to Neural Network Classification



Tomas Hrycej

Sixth-Generation Computer Technology Series
Branko Souček, Series Editor

Modular Learning in Neural Networks

A Modularized Approach to Neural Network Classification

TOMAS HRYCEJ

A Wiley-Interscience Publication

JOHN WILEY & SONS, INC.

New York - Chichester - Brisbane - Toronto - Singapore

Library of Congress Cataloging in Publication Data:

Hrycej, Tomas, 1954–

Modular learning in neural networks: a modularized approach to neural network classification/Tomas Hrycej.

p. cm.—(Sixth-generation computer technology series)

Includes index.

ISBN 0-471-57154-7

1. Neural networks (Computer science) I. Title. II. Series.

QA76.87.H78 1992

006.3′1–dc20 92-2554

CIP

Printed in the United States of America

10 9 8 7 6 5 4 3 2

Modular Learning in Neural Networks

Sixth-Generation Computer Technology Series

Branko Souček, Editor
University of Zagreb

Neural and Massively Parallel Computers: The Sixth Generation
Branko Souček and Marina Souček

Neural and Concurrent Real-Time Systems: The Sixth Generation
Branko Souček

Neural and Intelligent Systems Integration: Fifth and Sixth Generation Integrated Reasoning Information Systems
Branko Souček and the IRIS Group

Dynamic, Genetic, and Chaotic Programming: The Sixth Generation
Branko Souček and the IRIS Group

Fuzzy, Holographic, and Parallel Intelligence: The Sixth Generation Breakthrough
Branko Souček and the IRIS Group

Fast Learning and Invariant Object Recognition: The Sixth Generation Breakthrough
Branko Souček and the IRIS Group

Modular Learning in Neural Networks: A Modularized Approach to Neural Network Classification
Tomas Hrycej

To Emily, Anita, and Nathalie

PREFACE

This book is about *modular learning in artificial neural networks*. Why is this topic important? In the mainstream of neural network research and applications, neural networks are viewed as unstructured black boxes. This view is convenient as long as no problems (e.g., with learning speed or convergence) occur. However, we cannot expect this state to persist if our artificial neural networks grow and our applications become more difficult. The reasons that a modularization of networks and algorithms is desirable can be grouped in the following way:

1. *Engineering aspects.* If learning is viewed as a monolithic black-box task, there are no intermediate solution stages, and the success of each stage cannot be independently verified.
2. *Complexity aspects.* With growing network complexity, scaling and convergence problems of neural network learning arise.
3. *Psychological aspects.* Findings of developmental psychology show the incremental character of human learning, in which the success of each stage is conditioned by successful accomplishment of the preceding stage.
4. *Neurobiological aspects.* The human brain has a complex structure of cooperating modules.

The conceivable approaches to modularization of learning are very diverse. Those discussed in this book include:

1. Decomposition of learning into modules using various learning types (i.e., supervised and unsupervised learning)
2. Decomposition of the mapping to be represented (e.g., to linear and nonlinear parts)
3. Decomposition of the neural network to minimize harmful interactions during learning
4. Decomposition of the application task into subtasks that are learned separately
5. Decomposition into a knowledge-based part and a learning part

Special regard has been given to the fact that linear algorithms such as Hebbian learning are frequently orders of magnitude faster and more reliable than nonlinear algorithms such as the gradient method. This is why as large parts of the learning problem as possible should be solved by linear algorithms. Nonlinear algorithms would then be confined to the inevitable nonlinear hard core of the problem.

In some cases, such as self-organization and network prestructuring, formal analysis has been necessary to develop useful concepts or to prove important properties of network models. Sometimes, formal treatment has been useful for clarifying the *relationship of neural network methods to classical approaches* such as Bayesian classification or principal components analysis. I believe that this relationship is of particular interest for industrial researchers like myself. However, the *primary goal* of this book is to provide *evidence that modular learning* based on some of the approaches presented *is helpful in improving learning performance.* Since the ultimate goal is always to reach better performance for practical classification problems, most of the methods proposed have been tested on two benchmark cases of considerable size and application interest: (1) a medical classification problem of realistic size (7,200 cases of thyroid disorder), and (2) a handwritten-digit classification problem (20,000 cases). Some of the methods proposed led to substantial improvements in solution quality and learning speed (up to 100 times) as well as enhanced robustness with regard to learning control parameters.

I hope that this book will be stimulating both for *scientists* in suggesting yet undiscovered relationships resulting from the integrating view of learning modularization, and for neural network *application engineers*, in showing how neural network technology can be made more controllable by the decomposition of application tasks.

I would like to thank Professor Wolfram Büttner for many helpful remarks concerning related research and some general statements made in this book. I am also greatly indebted to Dr. Dieter Haban for reading the manuscript and polishing my English.

TOMAS HRYCEJ

Modular Learning in Neural Networks

CONTENTS

PREFACE xi

1 Introduction 1

1.1 Motivation for Decomposition of Learning, 1
1.2 Goals of This Book, 7
1.3 Computational Experiments and Data Sets Used, 8
1.4 Overview of Decomposition Methods, 10
1.5 Structure of This Book, 14

I NEURAL NETWORKS 15

2 Introduction to Neural Networks 17

2.1 Sources of Motivation for Neural Networks, 17
2.2 Basic Components of Neural Network Models, 21
2.3 Knowledge Representation, 31
2.4 Learning Types, 33
2.5 Some Important Models, 35
2.6 Task Types That Can Be Solved by Neural Networks, 45
2.7 Relationship to Other Research Fields, 51

3 Structure of the Brain 59

3.1 Macrostructure of the Brain, 60
3.2 Microstructure of Brain Regions, 63
3.3 Functional Principles, 65
3.4 Functional Circuits, 69
3.5 Information-Theoretic View of Brain Processing, 78

II MODULAR LEARNING 83

4 Decomposition of Learning into Unsupervised and Supervised Learning 85

4.1 Learning Architecture, 85
4.2 Existing Models of Self-Organization, 86

5 Supporting Supervised Learning by Feature Extraction 91

5.1 Neural Network Feature-Discovery Principles, 92
5.2 Computational Experience and Conclusions, 107
5.3 Related Work, 110
5.4 Feature Extraction Network as a Learning Module, 113

6 Supporting Supervised Learning by Quantization 115

6.1 Case For Single-Layer Learning Principles, 115
6.2 Supporting Single-Layer Perceptrons by Quantization, 118
6.3 Supervised Learning Module, 123
6.4 Computational Results with Quantization, 142
6.5 Quantization Network as a Learning Module, 149

7 Finding Optimal Features for a Given Task 151

7.1 Features Found by the Hidden Layer of a Multilayer Perceptron, 151
7.2 Supervised Feature Discovery, 153
7.3 Computational Experiments, 156
7.4 Related Work, 158
7.5 Supervised Feature Discovery and Adaptive Resonance Theory, 160

8 Decomposition of the Represented Mapping **165**

8.1 Decomposition into Linear and Nonlinear Parts, 166
8.2 Computational Experiments, 167
8.3 Linear Classifier as a Learning Module, 169

9 Decomposing the Network to Minimize Interactions **171**

9.1 Interactions Between the Weights of a Multilayer Network, 171
9.2 Layer-by-Layer Modification of the Backpropagation Algorithm, 173
9.3 Computational Experience, 174
9.4 Related Work, 176

10 Modularizing the Application Task **179**

10.1 Decomposing Network Structure According to Task, 180
10.2 Using Multiple Subtask Networks, 183
10.3 Decomposing the Task into Successive Phases, 184
10.4 Interacting Neural Networks, 185

11 Decomposing Network Construction into Knowledge-Based and Learning Parts **189**

11.1 Network Parameters Versus Network Topology, 190
11.2 Relationship Between Causal and Neural Networks, 191
11.3 Supporting Learning by Using Explicit Knowledge, 210
11.4 Related Work, 211

12 Conclusions **215**

REFERENCES ***219***

INDEX ***231***

CHAPTER 1

Introduction

1.1 MOTIVATION FOR DECOMPOSITION OF LEARNING

Although there have been several proposals for more structured views of neural networks in the past, and a growing number of them at present, the current *mainstream neural network technology and applications* share two limiting characteristics:

1. A neural network is viewed as a black box, whose structure is not explicit. The only conceptual interfaces on which results are observable are the input and output of the network. This is also true for network architectures whose input and output are not assigned to dedicated input or output units, such as nonlayered feedback networks. Input and output are then associated with certain states of the networks, and the sole items of interest are the initial (i.e., input) state and the terminal (i.e., output) state. Information about intermediate states is neither observed nor exploited for evaluation of computational progress.
2. The structure of neural networks is viewed as monolithic. There are no functional or task-specific differentiations.

As an example, we can take the most frequently used model, a layered feedforward network with backpropagation learning algorithm

(Werbos [166]; Rumelhart et al. [139]). Several aspects of this monolithic, black-box view can be observed with this network model.

1. Besides the trivial differentiation into input, output, and hidden units, they all play the same role in neural data processing. There are no subtasks that would be assigned to network parts.
2. The learning algorithm is applied uniformly to all network weights. For each weight, a gradient of the global error function is computed, and the weight is changed proportional to this gradient. There is no differentiation concerning the interdependencies between weight changes or the depth to which gradient propagation takes place.
3. All weights in all layers are changed simultaneously. This is another reason why interdependencies cannot be observed and analyzed. Another consequence of this parallelism (regardless of whether genuine or simulated) is that it reduces learning to a single-pass procedure.
4. A consequence of the preceding item is the lack of developmental stages, with whose help we would be able to evaluate the partial progress of the solution of the learning task.

This approach is to a certain degree deliberate. It is based on the observation that some large parts of human brain are relatively structureless or, more exactly, that the complexity of their visible structure is substantially lower than that of the tasks they are solving. It is particularly true of the neocortex, to which most human intelligence is attributed. The neocortex is composed of billions of neurons, but these neurons are merely of a few types, and their connections are organized in relatively few schemes or basic circuits (see, e.g., Shepherd [147]) that are repeated millions of times. It has even always been an explicit ambition of neural network research to find principles by which complex computations can be performed by a large number of equal, "anonymous," and simple computational units.

Today, we can say that several such powerful principles have been found. Although we cannot claim that our knowledge of these basic principles is complete, we seem to have reached a point at which it may be fruitful to allocate some part of research resources to the problem of how the huge task of cognitive learning can be solved by cooperation of several known principles rather than searching for a single, yet more powerful master principle. This thesis is supported by observations confirming the idea of brain function as consisting of a relatively large number of highly specialized dedicated structures. In some cases, concrete processing sequences of various neural network principles are assumed (e.g., Rolls's model of hippocampus operation [135]).

Under these circumstances it makes sense to investigate for methods of decomposition that would make neural network learning more efficient. In addition to neurobiological arguments, there are important aspects related to engineering, complexity, and developmental psychology. In the following sections we present arguments in favor of such decomposition. Further arguments can be found in the structure of human brain, which is the topic of Chapter 3.

1.1.1 Engineering Aspects

If an industrial application of nontrivial complexity is developed, the development process is usually broken down horizontally (i.e., to relatively independent parallel tasks) and vertically (i.e., to successive development stages). For example, developing a large software application package is decomposed to at least two vertical stages, system analysis and program coding, and to a number of modules (horizontal decomposition), which are independently designed, coded, and tested.

In addition to organization and management aspects, one of the major benefits of this developmental decomposition consists of the possibility of evaluating the success of each partial task independently. For each partial task, a partial success criterion and verification procedure is defined. If the entire application fails to operate correctly, the search for the cause of the failure can, in turn, be decomposed in correspondence with the partial tasks.

The currently widespread black-box view of neural networks is hardly reconcilable with such a development procedure. Its roots seem to be in the fact that distributed representations in neural networks are difficult to assign to any application *concepts*. However, this technical detail does not necessarily imply that independent *subtasks* cannot be implemented by individual subnetworks. The performance of such subnetworks can then be evaluated by individual performance measures. These individual performance measures may be different from the overall performance measure (typically, mean-squared error or misclassification rate). This is exactly what is necessary to make neural network technology acceptable in an industrial engineering environment.

1.1.2 Complexity Aspects

In the present state of neural network technology, the technology seems to be proven to have a large application potential. To date, a considerable number of successful application prototypes have been reported. A common feature of particularly successful applications is that they are based on relatively small networks. The reason for this success seems to be the careful choice of a realistically simple application rather than the computational power of small networks. It can be expected that in a near future, more complex tasks will be tackled by neural networks.

It is natural to expect that the growing complexity of mappings that are materialized by neural network tasks, and growing network size, lead to growing complexity of learning. For arbitrary mappings, learning is known to be NP-complete (Judd [84]). We know that this is not always disastrous. There are many NP-complete practical tasks for which solutions of satisfactory quality are found routinely. It is also clear that human information processing makes use of some methods for coping with NP-completeness. One of the usual approaches to this is certainly decomposing such complex tasks into parts of manageable size.

We can expect that this approach is also applicable to learning. There are at least two ways to benefit from the decomposition of learning:

1. The learning task is decomposed to several relatively independent subtasks each of which can be solved, despite its NP-completeness. The complexity of the overall task is then linear in the number of subtasks. This will not, of course, pass by the NP-completeness of the global task, but there is a high probability of getting good suboptimal global solutions if at least an intuitively good decomposition can be found without excessive computational costs.
2. For an NP-complete learning algorithm we substitute algorithms of less complexity. Such lower-complexity algorithms