NEURAL NETWORKS
AND
FUZZY SYSTEMS

A DYNAMICAL SYSTEMS APPROACH
TO MACHINE INTELLIGENCE

NEURAL NETWORKS
AND
FUZZY SYSTEMS

A DYNAMICAL SYSTEMS APPROACH
TO MACHINE INTELLIGENCE

Bart Kosko
University of Southern California

Editorial/production supervision
    and interior design: Joe Scordato
Jacket design: Wanda Lubelska
Manufacturing buyers: Linda Behrens and Dave Dickey
Acquisitions editor: Pete Janzow

The author and publisher of this book have used their best efforts in preparing this book. These efforts include the development, research, and testing of the theories and programs to determine their effectiveness. The author and publisher make no warranty of any kind, expressed or implied, with regard to these programs or the documentation contained in this book. The author and publisher shall not be liable in any event for incidental or consequential damages in connection with, or arising out of, the furnishing, performance, or use of these programs.
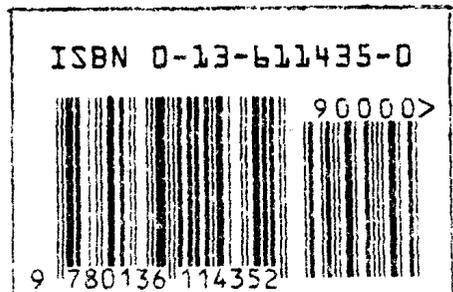
IBM is a registered trademark of International Business Machines Corporation.

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

ISBN 0-13-611435-0

ISBN 0-13-611435-0

9 780136 114352   90000>

FOREWORD

Lotfi A. Zadeh

It is hard to overestimate the importance of this textbook. Bart Kosko has broken new ground with an outstanding work on a subject, adaptive fuzzy systems, certain to play an increasingly central role in our understanding of human cognition and our ability to build machines that simulate human decision making in uncertain and imprecise environments.

This is what artificial intelligence (AI) was supposed to do when it was conceived in the mid-1950s. Since then, traditional AI, based almost entirely on symbol manipulation and first-order logic, has attracted a great deal of attention, a large following, and massive financial support. The AI community can point with pride to its accomplishments in expert systems, game-playing systems, and, to a lesser extent, natural language processing. Yet many of us believe that traditional AI has not lived up to its expectations. AI has not come to grips with common sense reasoning. It has not contributed significantly to the solution of real-world problems in robotics, computer vision, speech recognition, and machine translation. And AI arguably has not led to a significantly better understanding of thought processes, concept formation, and pattern recognition.

I believe AI would have made much more progress toward its goals if it had not

committed itself so exclusively to symbol manipulation and first-order logic. This commitment has made AI somewhat inhospitable to methods that involve numerical computations, including neural and fuzzy methods, and has severely limited its ability to deal with problems where we cannot benignly neglect uncertainty and imprecision. Most real-world problems fall into this category.

With this in view, we can better understand the growing popularity of numerical methods that deal with a wide range of real-world problems, problems AI has failed to solve if even address. Prominent among these numerical techniques are neural network theory and fuzzy theory. Separately and in combination, neural networks and fuzzy systems have helped solve a wide variety of problems ranging from process control and signal processing to fault diagnosis and system optimization. Professor Kosko's *Neural Networks and Fuzzy Systems*, along with its companion applications volume *Neural Networks for Signal Processing*, is the first book to present a comprehensive account of neural-network theory and fuzzy logic and how they combine to address these problems. Having contributed so importantly to both fields, Professor Kosko is uniquely qualified to write a book that presents a unified view of neural networks and fuzzy systems. This unified view is a direction certain to grow in importance in the years ahead.

Interpolation plays a central role in both neural network theory and fuzzy logic. Interpolation and learning from examples involve the construction of a model of a system from the knowledge of a collection of input-output pairs. In neural networks, researchers often assume a feedforward multilayer network as an approximation framework and modify it with, say, the backpropagation gradient-descent algorithm. In the case of fuzzy systems, we usually assume the input-output pairs have the structure of fuzzy if-then rules that relate linguistic or fuzzy variables whose values are words (fuzzy sets) instead of numbers. Linguistic variables facilitate interpolation by allowing an approximate match between the input and the antecedents of the rules. Generally, fuzzy systems work well when we can use experience or introspection to articulate the fuzzy if-then rules. When we cannot do this, we may need neural-network techniques to generate the rules. Here arise adaptive fuzzy systems.

One cannot be but greatly impressed by Professor Kosko's accomplishment as author of *Neural Networks and Fuzzy Systems*. This seminal work is a landmark contribution that will shape the development of neural networks and fuzzy systems for years to come.

Lotfi A. Zadeh
Department of Electrical Engineering
   and Computer Science
Computer Science Division
University of California at Berkeley

FOREWORD

James A. Anderson

We live in a world of marvelous complexity and variety, a world where events never repeat exactly. Heraclitus commented two and a half millennia ago that "We never step twice into the same river." But even though events are never exactly the same, they are also not completely different. There is a thread of continuity, similarity, and predictability that allows us to generalize, often correctly, from past experience to future events.

This textbook joins together two techniques—neural networks and fuzzy systems—that seem at first quite different but that share the common ability to work well in this natural environment. Although there are other important reasons for interest in them, from an engineering point of view much of the interest in neural networks and fuzzy systems has been for dealing with difficulties arising from uncertainty, imprecision, and noise. The more a problem resembles those encountered in the real world—and most interesting problems are these—the better the system must cope with these difficulties.

Neural networks, neurocomputing, or 'brainlike' computation is based on the wistful hope that we can reproduce at least some of the flexibility and power of the human brain by artificial means. Neural networks consist of many simple computing

elements—generally simple nonlinear summing junctions—connected together by connections of varying strength, a gross abstraction of the brain, which consists of very large numbers of far more complex neurons connected together with far more complex and far more structured couplings.

Neural-network architectures cover a wide range. In one sense, every computer is a neural net, because we can view traditional digital logic as constructed from interconnected McCullough-Pitts 'neurons'. McCullough-Pitts neurons were proposed in 1943 as models of biological neurons and arranged in networks for the specific purpose of computing logic functions. The architectures of current neural networks are massively parallel and concerned with approximating input-output relationships, with very many units arranged in large parallel arrays and computing simultaneously. Massive parallelism is of great importance now that the speed of light begins to constrain computers of standard serial design. Large-scale parallelism provides a way, perhaps the only way, to significantly increase computer speed. Even limited insights and crude approximations into how the brain combines slow and noisy computing devices into powerful systems can offer considerable practical value.

As this textbook shows, even the simple networks we now work with can perform interesting and useful computations if we properly choose the problem. Indeed, the problems where artificial neural networks have the most promise are those with a real-world flavor: signal processing, speech recognition, visual perception, control and robotics.

Neural networks help solve these problems with natural mechanisms of generalization. To oversimplify, suppose we represent an object in a network as a pattern of activation of several units. If a unit or two responds incorrectly, the overall pattern stays pretty much the same, and the network still responds correctly to stimuli. Or, if an object, once seen, reappears, but with slight differences, then the pattern of activation representing the object closely resembles its previous appearance, and the network still tends to respond almost as it did before. When neural networks operate, similar inputs naturally produce similar outputs. Most real-world perceptual problems have this structure of input-output continuity.

If neural networks, supposedly brain-like, show intrinsic generalization, we might wonder if we observe such effects in human psychology. Consider the psychological problem of categorization. Why do we call a complex manufactured object found in a house, an object we have not seen before, a "chair" because it has a more-or-less flat part a couple of feet off the floor, has four legs, consists of wood, and so on? One approach to categorization—popular with computer scientists—makes a list of properties and matches the new object with the property list. If the new object matches a listed property, then we conclude that the object is an example of the category; otherwise, we conclude that it is not. One quickly discovers with this approach that it does not work in practice. Natural categories tend to be messy: Most birds fly, but some do not. Chairs can consist of wood, plastic, or metal and can have almost any number of legs, depending on the whims of the designer. It seems practically impossible to come up with a property list

for any natural category that excludes all examples that are not in the category and includes all examples that are in the category.

The "prototype" model provides a model for human categorization with a good deal of psychological support. Instead of forming a property list for a category, we store a "best example" of the category (or possibly a few best examples). The system computes the similarity between a new example and the prototype and classifies the new example as an example of the category in the nearest-neighbor sense—if the new example is "close enough" to the prototype.

This computational strategy leads to some curious human psychology. For example, it seems that most people in the United States imagine a prototype bird that looks somewhat like a robin or a sparrow. (Of course, the prototype will depend, sometimes in predictable ways, on individual experience.) So Americans tend to judge ostriches or penguins as "bad" birds because these birds do not resemble the prototype bird, even though they are birds. "Badness" shows up in a number of ways: when people are asked to give a list of examples of "birds," prototypical birds tend to head the list; the response times to verify sentences such as "Penguins are birds" tend to be longer then to "Robins are birds;" and they put the prototypes into sentences as defaults in comprehension—the bird in "I saw a bird on the lawn" is probably not a turkey.

Neural networks naturally develop this kind of category structure. In fact, we can hardly stop neural networks from doing it, which points out a serious potential weakness of neural networks. Classification by similarity causes neural networks great distress in situations where we cannot trust similarity.

A famous example is "parity"—whether there are an even or an odd number of ones in a bit vector of ones and zeros. If we change only one element, then the parity changes. So nearest neighbors always have opposite parity. Parity causes no difficulties for digital logic. But it is so difficult for simple neural networks to compute the parity function that it, and related problems, caused the engineering community to lose interest in neural networks in the 1960's when computer scientists first pointed out this limitation. Yet such a pattern of computational strengths and weaknesses was exactly what excited the interest of psychologists and cognitive scientists at the same time. The problems that neural networks solved well and solved poorly were those where humans showed comparable strengths and weaknesses in their "cognitive computations." For this reason until quite recently most of the study of neural networks has been carried out by psychologists and cognitive scientists who sought models of human cognitive function.

Engineering techniques for dealing with uncertainty are sometimes as much statements about human psychology as they are about engineering. Neural networks deal with uncertainty as humans do, not by deliberate design, but as a byproduct of their parallel-distributed structure. It would be equally possible, and perhaps desirable, for us to directly build these insights about categorization into an artificial system. Fuzzy systems take this approach..

Fuzzy or multivalued set theory develops the basic insight that categories are not absolutely clear cut: a particular example of a category can "belong" to

lesser or greater degree to that category. This assumption captures quite nicely the psychological observation that a particular object can be a better or worse example of chair, depending on other members of the category. For example, an appropriately shaped rock in the woods can be a "chair" even though it is a very bad example of the chair category. So we can consider the rock as weakly belonging to the chair category. When such insights are properly (and elegantly) quantified, as in this textbook, "fuzzy" systems can be just as well defined and useful as the more traditional formulations of statistics and probability.

The theory of probability arose historically from an attempt to quantify odds in gambling, particularly the statistics of thrown dice, and in the more respectable type of gambling called insurance. Assumptions that color the basic structure of probability theory and statistics may arise from trying to explain a system that was specifically designed by humans for a particular purpose. For example, the rules for wining and losing in a game should be clear and precise because money may change hands. Every possible game outcome must fall into a predetermined category. These categories are noise free: the sum of the dice equals six and not seven; a coin comes up heads or tails. Individuals in the vital statistics of a population are alive or dead, baptised or unbaptised.

Because games are precise by design, traditional probability theory assumes an accuracy and precision of categorization of the world that may not represent many important problems. We must wonder what would have happened if, instead of being concerned with gambling or insurance, probability theory had been initially developed to predict the weather, where there are continuous gradations between overlapping linguistic categories: dense fog, drizzle, light rain, heavy rain, and downpour. Perhaps fuzzy systems would have become the mainstream of uncertainty-reasoning formalisms and 'traditional' probability an extreme approximation useful in certain special cases. The reaction of most people when they first hear about fuzzy logic is the subjective feeling "Yes, this formulation makes sense psychologically."

Because general statements about both human psychology and the structure of the world embed so deeply in both neural networks and fuzzy systems, the introductory parts of this book contain several examples drawn from philosophy, biology, cognitive science, and even art and law. These examples and references are not there to show the author's erudition, but to illuminate and make explicit the basic assumptions made when building the models. Unfortunately, neuroscientists and engineers often lie in unconscious bondage to the ideas of dead philosophers and psychologists when they assume that the initial formulations and basic assumptions of their abstract systems are "obvious." Like social customs, these assumptions are obvious only if you grew up with them.

There are also significant differences between neural networks and fuzzy systems. There are formal similarities between them, as Professor Kosko points out, but they are also very different in detail. The noise and generalization abilities of neural networks grow organically out of the structure of the networks, their dynamics, and their data representation. Fuzzy systems start from highly formalized insights about the psychology of categorization and the structure of categories found in the real

world. Therefore, the "theory of fuzziness" as developed is an abstract system that makes no further claims about biological or psychological plausibility. This abstract system may sometimes be easier to use and simpler to apply to a particular problem than neural networks may be. The reverse may also hold. Whether to use one or another technology depends on the particular application and on good engineering judgement.

Both neural networks and fuzzy systems break with the historical tradition, prominent in Western thought, that we can precisely and unambiguously characterize the world, divide it into categories, and then manipulate these descriptions according to precise and formal rules. Other traditions have a less positive approach to explicit, discrete categorization, one more in harmony with the ideas presented here. Huang Po, a Buddhist teacher of the ninth century, observed that "To make use of your minds to think conceptually is to leave the substance and attach yourself to form," and "from discrimination between this and that a host of demons blazes forth!"

> James A. Anderson
> Department of Cognitive and Linguistic Sciences
> Brown University

PREFACE

Neural networks and fuzzy theory have been underground technologies for many years. They have had far more critics than supporters for most of their brief histories. Until recently most neural and fuzzy researchers published and presented papers in non-neural and non-fuzzy journals and conferences. This has prevented standardization and suggested that the two fields represent pantheons of ad hoc models and techniques.

This textbook presents neural networks and fuzzy theory from a unified engineering perspective. The basic theory uses only elementary calculus, linear algebra, and probability, as found in upper-division undergraduate curricula in engineering and science. Some applications use more advanced techniques from digital signal processing, random processes, and estimation and control theory. The text and homework problems introduce and develop these techniques.

Neural networks and fuzzy systems estimate functions from sample data. Statistical and artificial intelligence (AI) approaches also estimate functions. For each problem, statistical approaches require that we guess how outputs functionally depend on inputs. Neural and fuzzy systems do not require that we articulate such a mathematical model. They are model-free estimators.

We can also view AI expert systems as model-free estimators. They map conditions to actions. Experts do not articulate a mathematical transfer function from the condition space to the action space. But the AI framework is symbolic. Symbolic processing favors a propositional and predicate-calculus approach to machine intelligence. It does not favor numerical mathematical analysis or hardware implementation. In particular symbols do not have derivatives. Only sufficiently smooth functions have derivatives. Symbolic systems may change with time, but they are not properly dynamical systems, not systems of first-order difference or differential equations.

Neural and fuzzy systems are numerical model-free estimators, and dynamical systems. Numerical algorithms convert numerical inputs to numerical outputs. Neural theory embeds in the mathematical fields of dynamical systems, adaptive control, and statistics. Fuzzy theory overlaps with these fields and with probability, mathematical logic, and measure theory. Researchers and commercial firms have developed numerous neural and fuzzy integrated-circuit chips. High-speed modems, long-distance telephone calls, and some airport bomb detectors depend on adaptive neural algorithms. Fuzzy systems run subways, tune televisions and computer disc heads, focus and stabilize camcorders, adjust air conditioners and washing machines and vacuum sweepers, defrost refrigerators, schedule elevators and traffic lights, and control automobile motors, suspensions, and emergency braking systems. In these cases, and in general, we use neural networks and fuzzy systems to increase *machine IQ*.

The book contains two large sections. The first section develops neural network theory. It presents neural dynamical systems as stochastic gradient systems. Chapter 1 previews the main neural and fuzzy themes developed throughout the text, focusing on hybrid neural-fuzzy systems for estimation and control. Chapter 2 presents neurons as signal functions and establishes notation. Chapter 3 develops both simple and general models of how the neuron's membrane evolves in time. Chapter 4 focuses on unsupervised synaptic learning and reviews probability theory and random processes. Chapter 5 presents supervised synaptic learning as stochastic approximation. Chapter 6 combines the material in previous chapters, allows neurons and synapses to change simultaneously, and proves global equilibrium and stability theorems for feedforward and feedback neural networks.

The second section examines fuzziness and adaptive fuzzy systems. Chapter 7 presents the new geometric theory of fuzzy sets as points in unit hypercubes. Chapter 7, as its "Fuzziness versus Probability" title suggests, deals in part with the apparent conflict between probability theory and fuzzy theory. This unavoidably evokes a polemical flavor and may jolt the unsuspecting reader. The pungency of the underlying journal article, of the same title, has helped secure it a much wider audience than I had thought possible. After teaching and lecturing on fuzzy theory for several years, I have found that audiences always ask the questions Chapter 7 anticipates. For these reasons I have included it in this textbook in its fuzzy-versus-probability form. I develop formal probability theory in the first four chapters of the textbook. I view its relation to fuzzy theory as taxonomical not adversarial. Chap-

ter 8 develops a new theory of fuzzy systems and shows how to combine neural and fuzzy systems to produce adaptive fuzzy systems. Chapter 9 compares neural and adaptive fuzzy systems for backing up a truck, and truck-and-trailer, to a loading dock in a parking lot. Chapter 10 applies the adaptive fuzzy methodology to signal processing and compares a fuzzy image transform-coding system with a popular algorithmic approach. Chapter 11 benchmarks an adaptive fuzzy system against a Kalman-filter controller for real-time target tracking. The Appendix discusses how to use the accompanying neural-network and fuzzy-system software, developed by HyperLogic Corporation, and Olmsted & Watkins, and Togai Infralogic.

For this textbook I have also edited for Prentice Hall a companion volume of neural applications, *Neural Networks for Signal Processing*. The edited volume extends and applies several of the neural-network models developed in this textbook. All chapters in the edited volume include detailed homework problems to assist in classroom instruction.

ACKNOWLEDGMENTS

This book took over a year to write and rewrite. Many colleagues assisted me in the process. I would like to thank my colleagues who patiently read and commented on drafts of chapters. In particular I would like to thank Seong-Gon Kong, Jerry Mendel, Peter Pacini, Rod Taber, Fred Watkins, and Lotfi Zadeh. Of course I assume all responsibility for chapter content. I would like to thank the staffs of HyperLogic Corporation, Olmsted & Watkins, and Togai InfraLogic for their tireless development of the fuzzy and neural software, software instructions, and software homework problems. I would like to thank Tim Bozik, Executive Editor at Prentice Hall, for his courage in instigating and supporting the evolving manuscript. Most of all I want to thank Delsa Tan for her persistence and equanimity as she processed every word and equation of the text in countably many iterations.

Bart Kosko
Los Angeles, California

CONTENTS

3 NEURONAL DYNAMICS II: ACTIVATION MODELS 55