From Statistics to Neural Networks

Theory and Pattern Recognition Applications

NATO ASI Series

Advanced Science Institutes Series

A series presenting the results of activities sponsored by the NATO Science Committee, which aims at the dissemination of advanced scientific and technological knowledge, with a view to strengthening links between scientific communities.

The Series is published by an international board of publishers in conjunction with the NATO Scientific Affairs Division

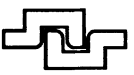
A Life Sciences	Plenum Publishing Corporation
B Physics	London and New York
C Mathematical and Physical Sciences	Kluwer Academic Publishers
D Behavioural and Social Sciences	Dordrecht, Boston and London
E Applied Sciences	
F Computer and Systems Sciences	Springer-Verlag
G Ecological Sciences	Berlin Heidelberg New York
H Cell Biology	London Paris Tokyo Hong Kong
I Global Environmental Change	Barcelona Budapest

NATO-PCO DATABASE

The electronic index to the NATO ASI Series provides full bibliographical references (with keywords and/or abstracts) to more than 30000 contributions from international scientists published in all sections of the NATO ASI Series. Access to the NATO-PCO DATABASE compiled by the NATO Publication Coordination Office is possible in two ways:

- via online FILE 128 (NATO-PCO DATABASE) hosted by ESRIN, Via Galileo Galilei, I-00044 Frascati, Italy.
- via CD-ROM "NATO Science & Technology Disk" with user-friendly retrieval software in English, French and German (© WTV GmbH and DATAWARE Technologies Inc. 1992).

The CD-ROM can be ordered through any member of the Board of Publishers or through NATO-PCO, Overijse, Belgium.



Series F: Computer and Systems Sciences, Vol. 136

From Statistics to Neural Networks

Theory and Pattern Recognition Applications

Edited by

Vladimir Cherkassky

Department of Electrical Engineering, University of Minnesota
Minneapolis, MN 55455, USA

Jerome H. Friedman

Department of Statistics, Stanford University
Stanford, CA 94309, USA

Harry Wechsler

Computer Science Department, George Mason University
Fairfax, VA 22030, USA

Springer-Verlag
Berlin Heidelberg New York London Paris Tokyo
Hong Kong Barcelona Budapest
Published in cooperation with NATO Scientific Affairs Division

Proceedings of the NATO Advanced Study Institute From Statistics to Neural Networks, Theory and Pattern Recognition Applications, held in Les Arcs, Bourg Saint Maurice, France, June 21–July 2, 1993

CR Subject Classification (1991): G.3, I.5, I.2.6, I.2.10

ISBN-13: 978-3-642-79121-5 e-ISBN-13: 978-3-642-79119-2
DOI: 10.1007/978-3-642-79119-2

Library of Congress Cataloging-in-Publication Data. NATO Advanced Study Institute From Statistics to Neural Networks, Theory, and Pattern Recognition Applications (1993: Bourg-Saint-Maurice, France). From statistics to neural networks: theory and pattern recognition applications/edited by Vladimir Cherkassky, Jerome H. Friedman, Harry Wechsler. p. cm. – (NATO ASI series. Series F, Computer and systems sciences; v. 136) 1. Neural networks (Computer science) 2. Pattern recognition systems. I. Cherkassky, Vladimir S. II. Friedman, J. H. (Jerome H.) III. Wechsler, Harry. IV. Title. V. Series. QA76.87.N382 1993 006.3'1–dc20 94-34254

Softcover reprint of the hardcover 1st edition 1994

Typesetting: Camera ready by editors
SPIN 10130792 45/3140 - 5 4 3 2 1 0 - Printed on acid-free paper

Preface

The NATO Advanced Study Institute From Statistics to Neural Networks, Theory and Pattern Recognition Applications took place in Les Arcs, Bourg Saint Maurice, France, from June 21 through July 2, 1993. The meeting brought together over 100 participants (including 19 invited lecturers) from 20 countries.

The invited lecturers whose contributions appear in this volume are: L. Almeida (INESC, Portugal), G. Carpenter (Boston, USA), V. Cherkassky (Minnesota, USA), F. Fogelman Soulie (LRI, France), W. Freeman (Berkeley, USA), J. Friedman (Stanford, USA), F. Girosi (MIT, USA and IRST, Italy), S. Grossberg (Boston, USA), T. Hastie (AT&T, USA), J. Kittler (Surrey, UK), R. Lippmann (MIT Lincoln Lab, USA), J. Moody (OGI, USA), G. Palm (Ulm, Germany), B. Ripley (Oxford, UK), R. Tibshirani (Toronto, Canada), H. Wechsler (GMU, USA), C. Wellekens (Eurecom, France) and H. White (San Diego, USA).

The ASI consisted of lectures overviewing major aspects of statistical and neural network learning, their links to biological learning and non-linear dynamics (chaos), and real-life examples of pattern recognition applications. As a result of lively interactions between the participants, the following topics emerged as major themes of the meeting:
(1) Unified framework for the study of Predictive Learning in Statistics and Artificial Neural Networks (ANNs);
(2) Differences and similarities between statistical and ANN methods for non-parametric estimation from examples (learning);
(3) Fundamental connections between artificial learning systems and biological learning systems.

These major themes are summarized below:

Predictive Learning and Statistics

A learning system is a computer program that constructs rules for predicting values for some property of a real system (response/output), given the values of other properties (predictors/inputs) of that system. In contrast to expert systems which attempt to organize the knowledge of human experts in the particular field, predictive learning systems attempt to construct useful prediction rules purely by processing data taken from past successfully solved cases; that is, cases for which the values of both the response and predictors have been determined. Learning

systems are generic programs that do not contain any domain specific knowledge. All information is presumed to be contained in the supplied data. It is the job of the learning algorithm to (automatically) extract and organize that information to obtain an accurate prediction rule.

Methodology and theory for computer learning have been traditionally developed in the fields of applied mathematics (multivariate function approximation), statistics (multiple regression and classification), and engineering (pattern recognition). Recently renewed interest and excitement has been generated by research in artificial intelligence (machine learning) and biologically motivated methods for data modeling (artificial neural networks), both in terms of widened applications and methodological development.

For the most part the leading developments in each of these fields has progressed independently of the corresponding developments in the other fields. There has been relatively little cross-referencing of the respective literatures. Many important "discoveries" in one field were in fact well known in others that study the predictive learning problem. A goal of this ASI was to bring together leading researchers and practitioners from these respective fields so that participants could learn their respective views of the problem and its solutions.

Since statistics is one of the oldest disciplines to study data based learning, it has perhaps seen the greatest duplication of its approaches in other fields. To quote B. Efron "Statistics has been the most successful information science, and those who ignore it are condemned to reinvent it." However it is also true that statistics has been slow to embrace modern (computer oriented) approaches to the problem. This reluctance is based largely on its history that has shaped the attitudes of those who are educated in the discipline.

Statistics was invented to study the limits of inference from data. That is, to what extent a collection of measurements actually characterizes the system under study, or is simply an artifact of a particular (random) sample taken from that system. To quote R. A. Fisher "It is the scientist (user), not the statistician, who constructs the structural model. It is the role of the statistician to study the inferential limitations of that model under various uncertainty (error) mechanisms." This involves the construction of probability models for the error mechanism and probability calculus to perform the inference. In this sense the statistician is often cast in the role of the messenger with bad news, since his main role is to inform the user of the limitations associated with what he wants to do, rather than the most optimistic opportunities. Statistical answers tend to be vague and highly qualified.

In the past the clients of statisticians were from the "softer" sciences (medicine, psychology, political science, economics, etc.) where sample sizes are small, noise levels very high and the issue is often whether there is any signal at all. In these settings inference from such data must be done with great care. This was the basis of the "scientific method". Statisticians are trained to be very careful and to always understate the strengths of methods they propose and stress the weaknesses. Only methods that have been extensively validated mathematically, and over a long period of use, are proposed for application.

This very cautious approach has caused statistics to evolve into a very scholarly mathematically oriented discipline that is initially suspicious of new ideas, often with good reason. As with all good things, caution and resistance to new

ideas can be overdone. There is a large market for successful predictive learning methods and computers are here to stay. Monte Carlo methods can be used to validate new procedures for which the mathematics becomes too difficult for direct solution. Large data bases generated by systems for which the signal to noise is high are now being routinely produced, especially in engineering and the physical sciences. For such data traditional statistical tools are not flexible enough to extract all the available information. The challenges associated with these types of data underlie the motivations of neural network and machine learning approaches.

In the field of statistics a small cadre of researchers has also taken up these challenges, sometimes producing surprisingly similar methods. For example, projection pursuit is quite similar to (feedforward) neural networks and recursive partitioning (CART) is nearly identical to many decision tree induction methods in machine learning. The statisticians attending this ASI are among those at the forefront of this effort. They bring with them a shared excitement for the new challenges associated with modern approaches to predictive learning, along with the care and scholarship that has traditionally characterized the field of statistics.

Statistical Versus ANN Methods for Learning from Examples

There is a range of conflicting opinions on the utility of ANNs for statistical inference. On one extreme, high expectations of neural network enthusiasts are usually caused by their statistical ignorance. On the other hand, negative attitude towards empirical neural network research on the part of some statisticians can be traced to the view that algorithmic approaches are inferior to analytical proofs. Some of the tension between the two fields is caused by the difference in research methodology. ANN researchers focus first on developing an algorithm for solving a particular application problem, while statisticians concentrate first on theoretical assumptions/analysis, implementation being a secondary issue. The following differences between the two approaches have been noted during this ASI:

Problem/model complexity. Usually (but not always) ANNs deal with large amount of training data (i.e. thousands of samples), whereas statistical methods use much smaller training samples. Hence, ANN models usually have higher complexity (number of parameters or weights) than statistical methods.

Goals of modeling. In statistics, the usual goal is *interpretability*, which favors structured models (i.e. classification trees and linear regression). In ANN research, the main objective is *generalization/prediction*. Hence, the ANN models usually have little, if any, interpretability. However, for many high-dimensional problems even structured methods are difficult to interpret, due to large model size (i.e. large classification tree). Interpretation of complex models produced by adaptive methods can be enhanced by computer visualization techniques.

Comparisons and search for the best method. Most statistical and ANN methods are asymptotically "good", i.e. can guarantee faithful estimates when the number of training samples grows very large. Unfortunately, the real world provides finite and usually sparse data sets. For such ill-posed problems, asymptotic performance is irrelevant, and the best method should conform to the properties of data at hand. Hence, no single method dominates all others for all possible data

sets. The real research goal is not to find the best method, but to characterize the class of functions/mappings (along with assumptions about the noise, smoothness etc.) for which a given method works best. Another important relevant problem is characterization of real data sets from important application domains.
Batch vs. flow-through processing. Most statistical methods utilize the whole training set (batch mode), whereas ANNs favor iterative processing (one sample at a time) known as flow-through methods in statistics. Iterative ANN processing requires many presentations of training data and uses slow computational paradigm (gradient descent). Statistical methods are usually much faster.
Computational complexity and usability. Since statistical methods extract information from the entire training set, they tend to be more computationally complex and difficult to use by non-statisticians. ANN methods tend to be simple computationally, albeit at the expense of recycling (multiple presentation) of the training data. Hence, ANN methods can be easily understood and applied by novice users. Also, hardware implementations favor simpler ANN methods.
Robustness and quality of prediction estimates. ANN methods appear more robust than statistical ones with respect to parameter tuning. Even suboptimal choice of parameters (network size, learning rate etc.) usually gives reasonable results. Another important aspect of robustness is the quality of prediction estimates produced by a modeling method, i.e. providing confidence intervals for prediction estimates. Confidence intervals are routinely provided in statistical methods, but usually are lacking in ANN application studies. Usually, the quality of solutions generated by ANNs cannot be guaranteed. This is partly due to the "black box" structure of ANNs. More work needs to be done to make them more transparent.

Towards Intelligent Information Processing: Biological Connections

Whereas the connection between artificial networks and statistical inference is generally well understood, the relationship/relevance of predictive learning framework to biological pattern recognition is not so clear. However, several common principles underlying the operation of ANNs and biological perceptual systems can be identified as discussed next.
Problem/system complexity. A key characteristic of intelligent behavior is the visual ability to sense, perceive and interpret the environment. Most perceptual (e.g. visual) tasks are complex mostly because they are under-constrained (or ill-posed). At the same time, human visual system exhibits robust performance, despite limited and slow computational resources. Perceptual systems are inherently more complex than (artificial) data-driven methods. It is widely believed that the human perceptual system handles complexity through the use of Active and Selective Perceptual strategies. The term Active Selective Perception means that the observer learns where to search for the data and what type of data to use for processing. ANNs and statistical methods do not have yet such attentive capabilities.
Representation bias. The basic tasks performed by biological neural networks involve learning adaptive mappings for approximating/mapping sensory input sig-

nals. The task of finding optimal signal representations is fundamental to any biological task (such as cognition and classification), and is arguably even more important than higher-level learning. Biological studies suggest representation of the sensory input through decomposition using local (kernel) bases ("receptive fields"). The problem of representation corresponds to selection of basis function in function approximation and predictive learning. Since various data modeling methods differ primarily in the choice of the basis functions, this problem also corresponds to the problem of model selection in predictive learning. Even though statistical studies suggest that there is no single best method/representation that dominates others for every possible data set, biological systems clearly favor local representations. This may be a biological cue to statisticians: pay more attention to local adaptive methods. Examples of local methods that do not have adequate statistical interpretation include wavelets and fuzzy inference systems (where each fuzzy rule can be interpreted as a local basis function). A particular challenging problem is to re-examine the curse of dimensionality, as it relates to local methods and real-world recognition problems. Even though theoretical limitations of local methods in high-dimensional spaces are well-known in statistics, empirical evidence provided by biological systems suggests that for many practical applications the curse of dimensionality is not relevant. It is most likely that assumptions underlying theoretical treatment of the curse do not hold for many real applications, possibly because of locality.

Hybrid systems. The complexity of biological neural systems and the wide range of tasks involved in perception require a large repertoire of intelligent modules. Hence, an advanced perceptual system should be able to develop hybrid adaptation systems driven by multistrategy learning schemes. It appears that employing hybrid learning systems accounts for success of several successful connectionist applications. The analogy between perception and ANNs is evident in the system approach used in the development of full fledged perceptual systems. The bottom-up ("data-driven") part corresponds to preprocessing and feature extraction. Once the features are derived, image classification ("understanding and classification") takes place. The top-down ("model-driven") part corresponds to post-processing ("disambiguation") and active selective perception ("priming") using global regularization constraints. This perceptual architecture is characteristic of hybrid systems where modularity is functionally defined. One important conclusion reached during this ASI was that future intelligent systems based on ANN technology should be task-driven and that their functionality could be enhanced by modular design using hybrid systems approaches and multistrategy learning.

Even though most lectures address one of the above themes, it became clear during the meeting that there was a strong connection between several of the topics. This lead to numerous discussions both in the formal setting (during lectures and panel discussions) and informal personal interactions between the participants.

The feedback from the participants regarding the contents of presented lectures and the social contacts during this meeting was very positive. This ASI was made possible by the efforts of the organizing committee: Vladimir Cherkassky (University of Minnesota, USA), Francoise Fogelman Soulie (LRI, France), Harry Wechsler (George Mason University, USA) and Christian Wellekens (Eurecom, France). We are grateful for financial support provided by the North

Atlantic Treaty Organization, Scientific Affairs Division, and for the matching funds provided by the US Office of Naval Research (ONR) and the US Advanced Research Projects Agency (ARPA).

Finally, special thanks are due to Filip Mulier from the University of Minnesota who spent endless hours on organization of the ASI and provided invaluable assistance in editing this volume.

July 1994

Vladimir Cherkassky, Minneapolis (Director)
Jerome H. Friedman, Stanford (Co-director)
Harry Wechsler, Fairfax (Co-director)

Contents

In the case of several authors, * indicates who presented the paper

An Overview of Predictive Learning and Function Approximation

Jerome H. Friedman

Department of Statistics
and
Stanford Linear Accelerator Center
Stanford University

Abstract: Predictive learning has been traditionally studied in applied mathematics (function approximation), statistics (nonparametric regression), and engineering (pattern recognition). Recently the fields of artificial intelligence (machine learning) and connectionism (neural networks) have emerged, increasing interest in this problem, both in terms of wider application and methodological advances. This paper reviews the underlying principles of many of the practical approaches developed in these fields, with the goal of placing them in a common perspective and providing a unifying overview.

1 The problem

The predictive learning problem is remarkably simple to state, if difficult to solve in general. One has a system under study characterized by several (possibly many) simultaneously measurable (observable) quantities, called variables. The variables are divided into two groups. The variables in one group are referred to (respectively) as independent variables (applied mathematics), explanatory/predictor variables (statistics), or input variables (neural networks/machine learning). The variables of the other group also have different names depending on the field of study: dependent variables (applied mathematics), responses (statistics), or output variables (neural networks/machine learning). The goal is to develop a computational relationship between the inputs and the outputs (formula/algorithm) for determining/predicting/estimating values for the output variables given only the values of the input variables.

For example, the system under study might be a manufacturing process. The inputs would be the various parameters that control the process such as chemical concentrations, baking time, precision of various machine tools, etc. The outputs would be measures of quality of the final product(s). The goal here would be to forecast the resulting quality(ies) from knowledge of the input parameter values without having to actually run the process. Realizing this goal could eliminate much experimentation with considerable financial benefit. In another example the "system" might be people potentially sick with one of several diseases or maladies. The inputs would consist of the existence/severity of various symptoms and a collection of medical laboratory

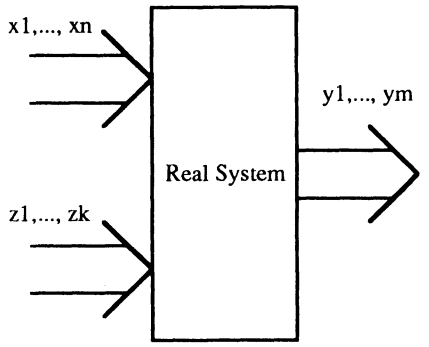


Figure 1. Diagram of the predictive learning problem

measurements. The output(s) would be indicators/severity of the potential diseases.

Figure 1 displays a pictorial representation of the problem. The system is represented by the rectangular box, the inputs as lines on the left side of the box and outputs $\{y_1, \cdots, y_K\}$ as lines on the right. The inputs are grouped into two sets. The first $\{x_1, \cdots, x_n\}$ are the inputs whose values are actually measured or observed. The other set $\{z_1, \cdots, z_L\}$ represent other quantities that relate to (affect) the outputs but whose values are neither observed nor controlled. Sometimes this second set of input variables does not exist ($L = 0$). One might be able to identify and measure all possible input variables that relate to the outputs. Often, however, this is not the case. It is unlikely that all possible things that affect the quality of a manufactured product can be measured, such as operator mood, distractions, etc. In medical diagnosis all symptoms are seldom observed and all possible diagnostic tests are not usually made. This means that a simultaneous set of observed input values $\{x_1, \cdots, x_n\}$ does not uniquely specify the output values. There is some uncertainty in the outputs reflecting the lack of knowledge of the unobserved input values.

By drawing the input lines on the left and outputs on the right (Fig. 1) one is tempted to associate a causal relationship between the values of the inputs and outputs. That is, changing the input values causes a change in the values of the outputs. This is often the case, as in the manufacturing process example. Sometimes, however, the reverse is true. Often it is the existence/severity of a disease that causes the existence/severity of the symptoms. Other times neither is true; changes in values of the inputs and outputs are both reflections of changes in other (unobserved) factors. In

many problems all three of these mechanisms exist. The point is that causality is not necessary for a derived relationship between the inputs and outputs to be either accurate, or useful, as in medical diagnosis. The converse is also true; the existence of an accurate input/output relationship need not reflect causality.

1.1 Types of variables

The input and output variables can each be of two different fundamental types: real or categorical. A real valued variable assumes values over a subset of the real line R^1. Examples are height, weight, voltage, course grade (F–A, mapped to 0–4). Values of these variables have an order relation and a distance defined between all pairs of values. Categorical variables have neither. For a categorical variable, two values are either equal or unequal. Examples are brand names, type of disease, nationality, etc. If a categorical variable assumes only two values then one of its values can be mapped to the real value zero and the other to real value one. It can then be treated as a real valued variable with no loss of generality. This is not the case if a categorical variable takes on more than two values.

When a categorical variable assumes more than two (say K) values it can be converted into K (0/1) real valued variables, one real variable for each categorical value. If the categorical variable assumes its kth value, its corresponding (kth) real valued surrogate is set to one and all of the other real surrogates, corresponding to different categorical values, are set to zero. If the categorical variable can assume only one of its values at a time then there is a linear degeneracy among the real variable surrogates and only $K-1$ of them are needed. This technique of mapping a categorical variable to real valued variables is referred to as "dummy variables" in statistics.

The dummy variable technique (trick) can always be used to deal with categorical variables. For categorical inputs it is not always the best way to do so. Methods differ on how well they can accommodate (extract information from) categorical inputs when they exist. (They don't occur for many problems.) Categorical outputs, however, occur often and represent an important class of problems referred to as pattern recognition in engineering, and classification/discriminant analysis in statistics. They are also the main focus of machine learning in artificial intelligence. Medical disease diagnosis is an example. For a categorical output variable the dummy variable trick is nearly always used. A single categorical output variable is converted into its real valued surrogates and the problem becomes a multiple (real-valued) output problem. This is discussed in more detail in Section 6.0. The point here is that it is sufficient to consider only real valued outputs in the supervised learning problem for the present. Translation of real-valued solutions back to the categorical output problem are deferred to Section 6.0.

1.2 Statistical model

The underlying mathematical model associated with the problem (Fig. 1) is

$$y_k = g_k(x_1, \cdots, x_n, z_1, \cdots, z_L), \quad k = 1, K. \tag{1}$$

Here y_k is the kth output value and g_k is a (real) single valued deterministic function of all possible (observed and unobserved) inputs that relate to changing values of y_k. To reflect the uncertainty associated with the nonobserved inputs $\{z_1, \cdots, z_L\}$, this relation (1) is replaced by a statistical model

$$y_k = f_k(x_1, \cdots, x_n) + \varepsilon_k, \quad k = 1, K. \tag{2}$$

Here f_k is a (single-valued deterministic) function of the observed inputs $\{x_1, \cdots, x_n\}$ only, and an additional random (stochastic) component ε_k is added to reflect the fact that simultaneous specification of a set of (observed) input values $\{x_1, \cdots, x_n\}$ does not uniquely specify an output value (unless ε_k is a constant). A specific set of input values specifies a distribution of (random) y_k values, characterized by the distribution of the random variable ε_k.

The fundamental models (1) (2) are represented by separate relationships for each output whose generic form is given by suppressing the k index. This suggests that they can be treated as separate problems without regard to the commonality of their input variable sets. This can be done and sometimes it represents the best way to proceed. Strategies have been proposed, however, that attempt to exploit possible associations among the output values to improve accuracy. Discussion of these strategies and the conditions under which they may (or may not) yield improvement over treating each output independently is deferred to Section 5.0. As noted there, the issue is not yet settled and is still an open topic for research. Until then we will treat the multiple output problem as separate single output problems. In many (most) problems, interest usually focuses on a single output.

In the following we denote

$$\boldsymbol{x} = \{x_1, \cdots, x_n\} \tag{3}$$

as a simultaneous set of input values. Our statistical model for each output (2) becomes

$$y = f(\boldsymbol{x}) + \varepsilon, \tag{4}$$

with $f(\boldsymbol{x})$ being a single valued deterministic function of an n-dimensional argument, and ε a random variable which is presumed to follow some probabilistic law (probability distribution), $\varepsilon \sim F_\varepsilon(\varepsilon)$. That is, $F_\varepsilon(\varepsilon')$ gives the probability of observing a value of $\varepsilon \leq \varepsilon'$. Using statistical notation we will denote with the symbol $E[\cdot]$ the average of a quantity over this distribution, i.e.

$$E_\varepsilon[h] = \int_{-\infty}^{\infty} h(\varepsilon') dF_\varepsilon(\varepsilon').$$

The model (2) is ambiguous since one can add a constant value to ε and subtract the same (constant) value from $f(\boldsymbol{x})$ and leave all values of the output y unchanged. This (definitional) ambiguity is usually resolved by the additional definition

$$E(\varepsilon) = 0 \tag{5}$$

for all values of $\boldsymbol{x}$. In this case one has

$$f(\boldsymbol{x}) = E_{\varepsilon}[y \mid \boldsymbol{x}]. \tag{6}$$

That is $f(\boldsymbol{x})$ is defined as the (unique) average output value y, for the specified set of input values $\boldsymbol{x}$.

1.3 Supervised learning

The goal of any learning approach is to obtain a useful approximation $\hat{f}(\boldsymbol{x})$, to the true ("target") function $f(\boldsymbol{x})$ (4) (6) that underlies the predictive relationship between the inputs and output. There are many ways to do this. One could try to derive the fundamental equations corresponding to the physical or chemical laws that control the system. Experts could be consulted as to their knowledge of the system. Supervised learning attempts to learn the input/output relationship by example through a "teacher". This process is depicted in Figure 2. Here one observes the system under study, simultaneously measuring both the (observed) inputs and the corresponding output values. This is done repeatedly, collecting a "training" sample of N simultaneous sets of input/output values

$$\{y_i, x_{i1}, \cdots, x_{in}\}_1^N = \{y_i, \boldsymbol{x}_i\}_1^N. \tag{7}$$

The (observed) input values to the system are also input to an artificial system (learning algorithm – usually a computer program) that also produces outputs $\{\hat{f}(\boldsymbol{x}_i)\}_1^N$ in response to the inputs $\{\boldsymbol{x}_i\}_1^N$. The artificial system has the property that it can modify (under constraints) the input/output relationship $\hat{f}(\boldsymbol{x})$ that it produces in response to differences $\{y_i - \hat{f}(\boldsymbol{x}_i)\}_1^N$ (errors) between the artificial and real system outputs, as provided by a "teacher". This modification process is called learning (by example). Upon completion of the learning process the hope is that the artificial and real outputs will be close enough to be useful for all sets of (simultaneous) values of inputs likely to be encountered.

1.4 Function approximation/estimation

The learning paradigm of the previous section has been the motivation for research into the supervised learning problem in the fields of machine learning (analogies to human reasoning) and neural networks (biological analogies

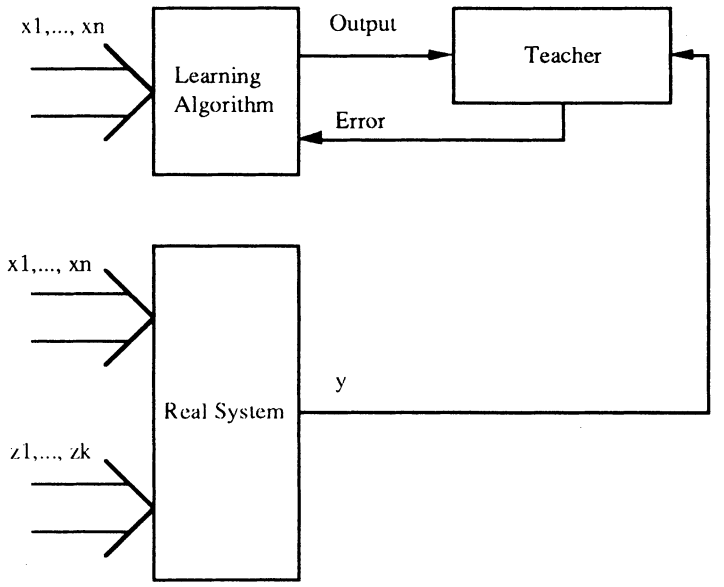


Figure 2. Diagram of supervised learning approach

to the brain). The approach taken in applied mathematics and statistics has been from the perspective of function approximation/estimation. In this view a simultaneous set of input values (3) is regarded as a point in an n-dimensional Euclidean space ($\boldsymbol{x} \in R^n$) and the target function (4) (6) as a function defined on that space. It is thus a surface (manifold) in the $(n+1)$-dimensional joint input/output space. The training sample (7) represents a point cloud in R^{n+1} related to the surface $[\boldsymbol{x}, f(\boldsymbol{x})] \in R^{n+1}$ by $\{\boldsymbol{x}_i, f(\boldsymbol{x}_i) + \varepsilon_i\}_1^N$ (4). The goal is to obtain a useful approximation to $f(\boldsymbol{x})$ for all $\boldsymbol{x}$ in some region of R^n, given its value (possibly contaminated with noise, $\varepsilon \neq 0$) only at the set of points represented by the training sample. Although somewhat less glamorous than the learning paradigm, treating supervised learning from the point of view of function approximation allows the geometrical concepts of Euclidean spaces and mathematical concepts of probabilistic inference to be applied to the problem. This will be the approach taken here.

Function approximation is often divided into two subjects depending on the existence of an error term ε (4). If it is everywhere equal to zero the problem is referred to as interpolation. In this case there are no unobserved inputs (1), $\{z_i \cdots\} = \text{null}$, and the observed inputs $\boldsymbol{x} = \{x_1 \cdots x_n\}$ are the

only ones that relate to changing output y values. In this case specifying a simultaneous set of inputs $\boldsymbol{x}$, uniquely specifies an output value. The interpolations problem is then to approximate true target function $f(\boldsymbol{x})$ everywhere within a region of the input space, given only its value at a finite number of points within the region (training sample). This is the problem treated in applied mathematics (multivariable function approximation).

When unobserved inputs do exist, the error term (4) is not generally zero and the output y becomes a random variable. Specifying a set of (observed) input values $\boldsymbol{x}$, specifies a distribution of output y-values whose mean is the target function $f(\boldsymbol{x})$ (6). This is the problem usually studied in the statistical literature and is referred to as nonparametric (flexible) multivariate regression. One goal is the same as in the interpolation problem; approximate the target function everywhere within a region of the input space given a training sample. The difference is that here the problem is more difficult since the target function is only approximately known at the training (input) points owing to the error term (4). Another (less ambitious) goal often addressed in the statistical literature is to estimate the output values only at the training sample points, given error contaminated values at those points. This is theoretically more tractable but seldom useful in practice.

There are two distinct practical reasons for application of supervised learning/function approximation: prediction and interpretation. In prediction it is expected that in the future new observations will be encountered for which only the input values are known, and the goal is to predict (estimate) a likely output value for each such case. The function estimate $\hat{f}(\boldsymbol{x})$ obtained from the training data through the learning algorithm is to be used for this purpose. In this case the primary goal is accuracy; one would like the estimates $\hat{f}$ to be close to the (unobserved) output y over this future prediction sample. Let $\Delta(y, \hat{f})$ be some measure of distance (error) between the two quantities. Common examples are

$$\Delta[y, \hat{f}] = |y - \hat{f}|, \quad \text{or} \tag{8a}$$

$$\Delta[y, \hat{f}] = (y - \hat{f})^2, \tag{8b}$$

the later being the most popular because its minimization leads to the simplest algorithms. A reasonable measure of (lack of) performance would then be the global prediction error

$$\Delta_P = \frac{1}{N_P} \sum_{i=1}^{N_P} \Delta[y_i, \hat{f}(\boldsymbol{x}_i)]$$

where the sum is over the N_P (future) prediction samples. Generally the precise locations of the prediction sample are not known in advance but can be assumed to be a random sample from some probability density $p(\boldsymbol{x})$, whose

values give the relative probabilities of encountering a new observation to be predicted at $\boldsymbol{x}$. The global error then becomes

$$\Delta_P = \int E_\varepsilon \Delta[f(\boldsymbol{x}) + \varepsilon, \hat{f}(\boldsymbol{x})] p(\boldsymbol{x}) d\boldsymbol{x}$$

where ε is the noise term (4) and E_ε is the average over its distribution $F_\varepsilon(\varepsilon)$. In particular if (8b) is chosen, then the mean-squared prediction error is given by

$$\begin{aligned} mspe &= \int E_\varepsilon [f(\boldsymbol{x}) + \varepsilon - \hat{f}(\boldsymbol{x})]^2 p(\boldsymbol{x}) d\boldsymbol{x} \\ &= \int [f(\boldsymbol{x}) - \hat{f}(\boldsymbol{x})]^2 p(\boldsymbol{x}) d\boldsymbol{x} + \int \sigma^2(\boldsymbol{x}) p(\boldsymbol{x}) d\boldsymbol{x} \end{aligned} \tag{9}$$

where $\sigma^2(\boldsymbol{x}) = E_\varepsilon(\varepsilon^2|\boldsymbol{x})$ is the variance of the noise at $\boldsymbol{x}$. The first term (9) is known simply as the mean-squared-error

$$mse = \int [f(\boldsymbol{x}) - \hat{f}(\boldsymbol{x})]^2 p(\boldsymbol{x}) d\boldsymbol{x} \tag{10}$$

and it completely captures the dependence of the prediction error on $\hat{f}(\boldsymbol{x})$. Minimizinig (9) and (10) with respect to $\hat{f}(\boldsymbol{x})$ gives the same result. In particular, these results show that the target function $f(\boldsymbol{x})$ (4) (6) is the best (mean-squared-error) predictor of future outputs and accurate approximation and future prediction are equivalent goals.

Another reason for applying supervised learning is interpretation. The goal here is to use the structural form of the approximating function $\hat{f}(\boldsymbol{x})$ to try to gain insight and understanding concerning the mechanism that produced the data. In this context no future data is necessarily envisioned. The approximation is intended to serve primarily as a descriptive statistic for illuminating the properties of the input/output relationship. Some properties of interest might include identification of those input variables that are most relevant to (associated with) the variation in the output, the nature of the dependence of the output on the most relevant inputs, and how that changes with changing values of still other input values. Such information can be quite useful for understanding how the system works and perhaps how to improve it. Also, nearly all scientific research is based on understanding from data rather than (the narrow goal of) future prediction.

Accuracy (10) has some importance to this application since its not very useful to interpret an approximation that bears little resemblance to the true input/output relationship $f(\boldsymbol{x})$. However it is not the only goal. The informal criterion (to be optimized) is the amount of (correct) information learned about the system. Human engineering considerations are involved in determining the best ways to summarize and present the approximating equation/algorithm in the best form (graphical and tabular) for human