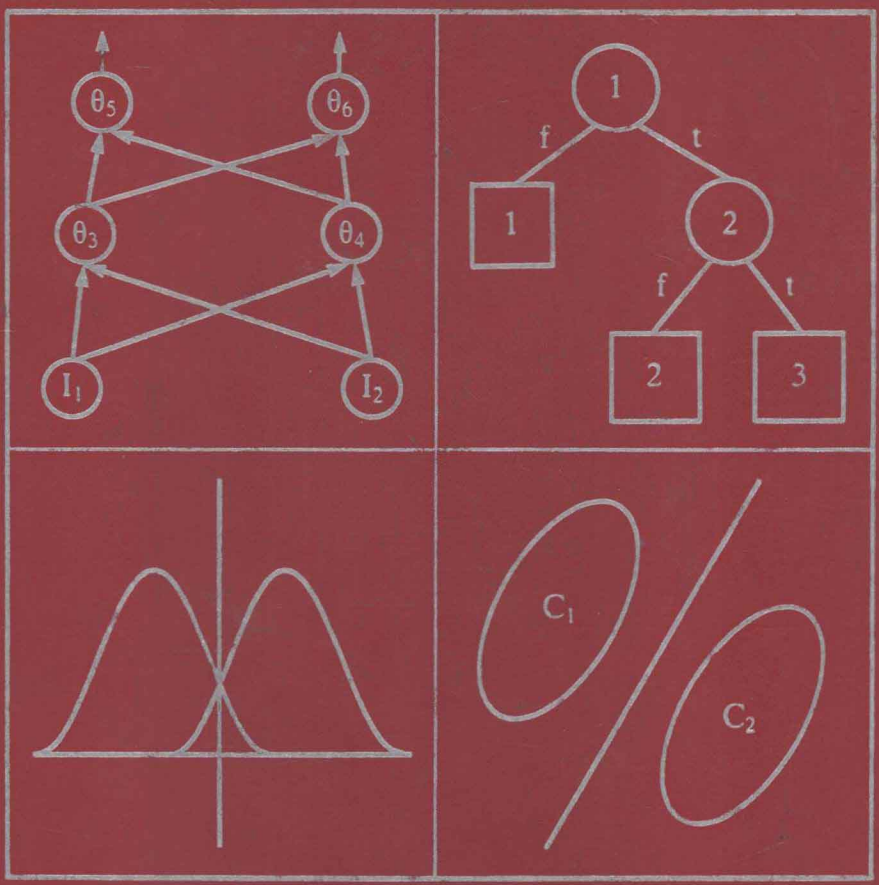
Computer Systems That Learn

Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems



Sholom M. Weiss • Casimir A. Kulikowski

Computer Systems That Learn

Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems

Sholom M. Weiss
Rutgers University

Casimir A. Kulikowski
Rutgers University

MORGAN KAUFMANN PUBLISHERS, INC.
SAN MATEO, CALIFORNIA

Sponsoring Editor: Bruce M. Spatz
Production Editor: Sharon E. Montooth
Cover Designer: John Edeen
Copyeditor: Bob Klinginsmith
Proofreader: Martha Ghent

Library of Congress Cataloging-in-Publication Data

Weiss, Sholom M.
Computer systems that learn : classification and prediction methods from statistics, neural nets, machine learning, and expert systems / Sholom M. Weiss, Casimir Kulikowski.
p. cm.
Includes bibliographical references and index.
ISBN 1-55860-065-5
1. Machine learning. 2. Classification. 3. Forecasting.
I. Kulikowski, Casimir A. II. Title.
Q325.5.W45 1990
006.3--dc20 90-48187
CIP

ABCDEFGHIJ-DO-93210

Morgan Kaufmann Publishers, Inc.
2929 Campus Drive, Suite 260
San Mateo, California 94403

Computer Systems That Learn

Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems

Preface

Why do we feel a need to write a book about systems that learn to classify from sample data? Surely, one of the many books already available on this classical topic should be adequate.

Our objective is to produce a practical guide to the application of classification learning systems. While many existing texts serve as excellent references, we feel that they do not completely satisfy the needs of our "applied" view of the field. Here are the main reasons:

- Research on this topic has been pursued by different research communities with different perspectives and different methods. Most books on classification are written by specialists from one area. It would be convenient to have the most prominent methods from each specialty combined in one book.
- Books on these topics are mostly written by academics who cover the formal theory very well, but often omit practical information on the application of learning methods.
- There are some relatively new learning methods, particularly those from machine learning and neural networks, that are not adequately covered in older texts. These methods are coming of age because of the availability of considerable computational power, even in inexpensive desktop machines.
- We found in most of our readings that the emphasis is on classification, and not on prediction. The key test of any learning

method is that its predictions on new data hold up. The methodology for such analysis is often ignored. Yet, such a performance evaluation should be the fundamental basis for understanding and analyzing any learning system.

Our book is an attempt to cover classification and prediction in a single text geared towards the practitioner. Learning systems should not be considered a black box, governed by complicated mathematics, with answers that may surprise or disappoint us. We have attempted to give an intuitive explanation of each method with a minimum of mathematics. For each method, we have tried to state simply the known theoretical results, the known tendencies of the method, and our recommendations for how to get the best results from the method. Some may disagree with one or another of these recommendations, but at least the practitioner will have advice with which to proceed and which can be tested empirically.

In Chapter 1, we introduce the tasks of classification and prediction.

Chapter 2 describes the techniques for estimating the true error rate on future cases. This is of fundamental importance for comparing classifiers on the same samples and also for selecting key characteristics of many of the newer classifiers. With limited samples, the best techniques are resampling methods that simulate the presentation of new cases by repeatedly hiding some test cases.

Chapter 3 reviews the most prominent methods from statistical pattern recognition.

Chapter 4 considers perceptrons and back-propagation neural networks. This is an area of renewed research and excitement in learning systems because of the strong theoretical and applied results that have been obtained recently.

Chapter 5 describes the machine learning methods for generating decision trees and inducing production rules. In contrast to the other techniques, these methods present solutions in a form compatible with typical human reasoning.

In Chapter 6, an empirical comparison is made of all the methods on several sample applications. We follow this with comments on the practicality of each of the methods, and recommendations for selecting and applying a learning system.

Chapter 7 contrasts learning systems that use sample data with rule-based expert systems that attempt to explicitly capture the knowledge of human experts in a computer program. The potential advantages of combining empirical learning with expert systems are discussed.

The topics covered are suitable for a course at the graduate level in computer science, statistics, and most engineering specialties. Alternatively, anyone who has practical experience with any specialized learning system, perhaps having used a programmed application of one of the learning methods, should be able to absorb the material in this book.

We thank Kevin Kern for his outstanding programming support. Ioannis Kapouleas and Nitin Indurkhya worked with us in many data analyses, and we benefitted greatly from their insights into these applications and learning methods. Professor Michael S. Watanabe, with his pioneering work on feature selection and classification, inspired us to the study and research of classical pattern recognition problems. We thank Bruce Spatz, our editor, and three anonymous referees, whose reviews of a draft were of great help in preparing the final manuscript. We also acknowledge the many years of support from the National Institutes of Health for research on expert systems and empirical learning.

Contents

1

Overview of Learning Systems

1.1 What Is a Learning System?

A *learning system* is a computer program that makes decisions based on the accumulated experience contained in successfully solved cases. Unlike an *expert system*, which solves problems using a computer model of expert human reasoning, a pure learning system can use many different techniques for exploiting the computational power of a computer, regardless of their relation to human cognitive processes. These techniques include many highly mathematical methods, as well as others that can search systematically over large numbers of possibilities.

The goals for learning systems are no different from those frequently cited for expert systems:

- To deal with complex real-world decision-making problems
- To solve these problems in the sense of reaching correct conclusions.

People are constantly faced with making important decisions. In choosing how to make these decisions, they almost always rely on prior experience. Some individuals may be considered experts in a field because they have accumulated considerable experience, and are

known to make accurate decisions with the professional knowledge to explain and support their conclusions.

The present book describes how computers can be used to help make decisions. We concentrate on learning systems that extract decision criteria from samples of solved cases stored in a computer. Learning systems attempt to solve specific problems without significant human intervention. Our emphasis is on practical methods that have consistently proven successful in building learning systems.

1.2 Motivation for Building Learning Systems

In many professional fields expertise is scarce, and the codification of knowledge can be quite limited in practice. Expertise, in the form of records of solved cases, may be the sole source of knowledge. In other fields, good tests and measurements may be available, but methods of applying this information may be insufficiently understood or systematized. Examples abound in many areas of expertise. Physicians, for instance, are always searching for the best test to make a particular diagnosis. Investors must decide whether to buy or sell a stock based on information about the company and its economic prospects. Tax authorities must decide who should be audited based on tax data. Banks and other credit institutions must approve or disapprove credit applications based on personal finance information.

From a systems design perspective, there are several reasons why there has been an increased interest in learning systems. New formal methods and new techniques of implementation have been developed. Both the cost and speed of running learning systems have improved dramatically over what was feasible in the late 1950s, when the first computer learning systems were developed.

Computer aids to decision-making were among some of the earliest research programs in medical diagnosis, signal analysis, image analysis, and other pattern recognition applications. Statistical and heuristic approaches saw widespread implementation in the 1960s, and were augmented by expert knowledge-based approaches in the 1970s and 1980s. The former, which included an early version of neural nets, called perceptrons, extracted decision-making rules or procedures from collections of samples of solved cases. While these approaches had the advantage of learning from the experience stored in the cases, they could not exploit the many pieces of other

heterogeneous knowledge that an expert would bring to bear in solving a problem. The expert system approach, in contrast, was designed to make it easy to codify such expert "rules of thumb," but because of the greater complexity of knowledge representation, made automatic learning very much harder. As a result, expert systems are often criticized for being limited in their abilities to surpass the level of existing experts. Another often cited problem is the great effort required to build and maintain a knowledge base, and the shortage of trained *knowledge engineers* to interview experts and capture their knowledge in a set of decision rules or other representational elements. This process, known as *knowledge acquisition*, is quite time-consuming, leading to lengthy development, which must be continued if the system is to be maintained in routine use at a high level of performance.

The argument in favor of learning systems is that they have the potential to exceed the performance of experts and the potential to discover new relationships among concepts and hypotheses by examining the record of successfully solved cases. Thus, the process of learning automatically holds out the promise of incorporating knowledge into the system without the need of a knowledge engineer.

Yet, there are strong practical reasons to expect that what can be learned directly from sample experience alone is limited, if it ignores the context within which problem solving is carried out. Thus, there is a need to combine domain-specific expert knowledge with learning approaches. Work along these lines is still in its infancy, but progress in learning methods has been sufficient over the past decade to warrant a comparison of techniques and results for practical application problems.

In this book, we review some of the most prominent methods for building learning systems. Expert systems, based on explicit encodings of an expert's knowledge, are viewed as alternatives to learning systems. In some applications, where expertise is limited, these learning methods may surpass an expert system in performance, as they can aggregate knowledge that has yet to be formalized. In other instances, learning approaches may provide a minimal threshold of performance that must be surpassed in order to justify the investment of building an expert system. Since learning systems need not exclude building on the foundation of an expert knowledge base, in the future, the two types of systems may be increasingly integrated.

1.3 Types of Practical Empirical Learning Systems

The description we have given of learning systems is quite broad, but in this book we confine our attention to the most prominent and basic learning task: *classification* or *prediction*. Classification is the most widely used name, is often associated with statistical pattern recognition, and more recently has been used to characterize many expert system applications. In statistics the classification problem is sometimes called the *prediction problem*, and in the field of machine learning it is often called *concept learning*. Each of the examples mentioned earlier can be readily posed as classification tasks. As illustrated in Figure 1.1, the fundamental goal of empirical learning is to extract a decision rule from sample data that will be applicable to new data. A typical learning system is designed to work with some general model, such as a decision tree, a discriminant function, or a neural net. "Learning" consists of choosing or adapting parameters within the model structure that work best on the samples at hand and others like them.

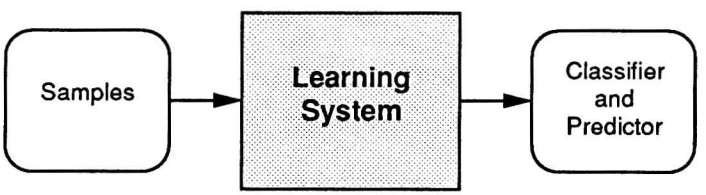


Figure 1.1
Learning System

1.3.1 Common Theme: The Classification Model

For classification problems, a learning system can be viewed as a higher-level system that helps build the decision-making system itself, called the classifier. The simplest way of representing a classifier is as a black box which produces a decision for every admissible pattern of data that is presented to it. Figure 1.2 illustrates the simple structure

of a classification system. It accepts a pattern of data as input, and produces a decision as output.

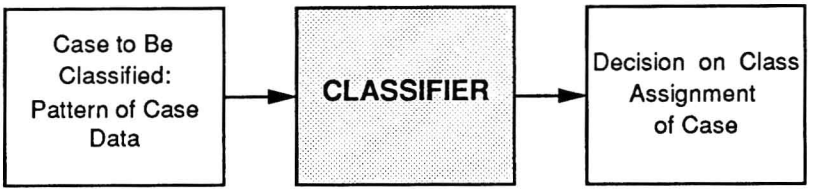


Figure 1.2
Classification System

The learning system has available to it a finite set of samples of solved cases. The data for each case consists of a pattern of observations and the corresponding correct classification. In addition, the general structure or type of classifier must be selected by the person who has specified the problem. The objective of the learning system is to customize the classifier structure to the specific problem by finding a general way of relating any particular pattern of observations to one of the specified classes. For example, in medical diagnosis, the physician has observations and test results, and the objective is to pick the correct diagnosis. At the race track, a bettor has a list of horses that are racing and a record of their previous race results, track conditions, and other relevant records.

The basic representation of the classification problem is therefore quite simple. Each sample of a solved problem consists of observations and the corresponding correct class membership. Usually a single conclusion or class will result for a given pattern of observations. While there may be problems where multiple class membership is important, these can often be decomposed into several subproblems involving individual class membership.

A set of samples therefore contains the data that the learning system will use to find the generalized decision rules for the classifier. Figure 1.3 illustrates the representational elements of the collection of solved sample cases, and the relation of the learning system to the classifier. The set of potential observations relevant to a particular problem are also referred to as *features*. Features also go by a host of other names, including *attributes*, *variables*, tests and measurements.

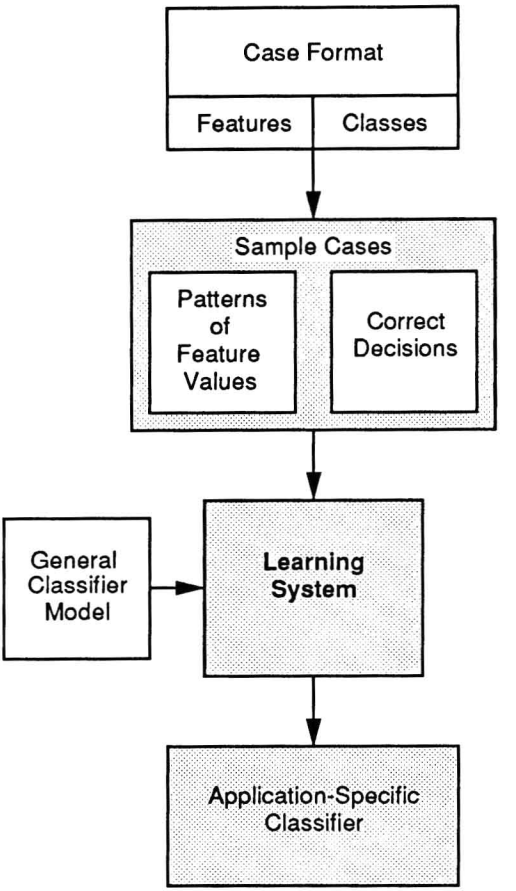


Figure 1.3
Classification Representation

Because only correctly solved cases will be used in building the specific classifier, the pattern of feature values for each case is associated with the correct classification or decision. Thus, learning in any of these systems can be viewed as a process of generalizing these observed empirical associations subject to the constraints imposed by the chosen classifier model.

By way of example, let's consider the problem of predicting whether the stock market will go up or down over the next three months. Here there are two classes: (1) stock average rises; or (2) stock average