MATHEMATICS
IN SCIENCE
AND
ENGINEERING

Volume 52

Sequential Methods in Pattern Recognition and Machine Learning

SEQUENTIAL METHODS IN PATTERN RECOGNITION AND MACHINE LEARNING

K. S. FU

School of Electrical Engineering
Purdue University
Lafayette, Indiana

ACADEMIC PRESS New York and London 1968

ACADEMIC PRESS, INC.
111 Fifth Avenue, New York, New York 10003

United Kingdom Edition published by
ACADEMIC PRESS, INC. (LONDON) LTD.
Berkeley Square House. London W.1

LIBRARY OF CONGRESS CATALOG CARD NUMBER: 68-8424

PRINTED IN THE UNITED STATES OF AMERICA

PREFACE

During the past decade there has been a considerable growth of interest in problems of pattern recognition and machine learning. This interest has created an increasing need for methods and techniques for the design of pattern recognition and learning systems. Many different approaches have been proposed. One of the most promising techniques for the solution of problems in pattern recognition and machine learning is the statistical theory of decision and estimation. This monograph treats the problems of pattern recognition and machine learning by use of sequential methods in statistical decision and estimation theory.

The material presented in this volume is primarily based on the research carried out by the author and his co-workers, Dr. G. P. Cardillo, Dr. C. H. Chen, Dr. Y. T. Chien, and Dr. Z. J. Nikolic during the past several years. In presenting the material, emphasis is placed upon the development of basic theory and computation algorithms in systematic fashion. Although many different types of experiments have been performed to test the methods discussed, for illustrative purpose, only experiments in English-character recognition have been presented. The monograph is intended to be of use both as a reference for system engineers and computer scientists and as a supplementary textbook for courses in pattern recognition and adaptive and learning systems. The presentation is kept concise. As a background to this monograph, it is assumed that the reader has adequate preparation in college mathematics and an introductory course on probability theory and mathematical statistics.

The subject matter may be divided into two majors parts: (1) pattern recognition and (2) machine learning. Roughly speaking, six approaches are presented, they are equally divided from Chapter 2 to Chapter 7. After a brief review of several important approaches in pattern recognition in Chapter 1, two methods for feature selection and ordering in terms of information theoretic approach and Karhunen–Loève expansion are presented in Chapter 2. In addition to the application of Wald's sequential probability ratio test and the generalized sequential probability ratio test to pattern classification problems, three techniques are discussed, namely, the modified sequential probability ratio test with time-varying stopping boundaries (Chapter 3), the backward procedure using dynamic programming (Chapter 4), and the nonparametric sequential ranking procedure (Chapter 5). The application of dynamic programming to both feature ordering and pattern classification is also included in Chapter 4. A brief introduction to sequential analysis is given in Appendix A.

Bayesian estimation techniques (Chapter 6) and the stochastic approximation procedure (Chapter 7) are introduced as learning techniques in sequential recognition systems. Both supervised and nonsupervised learning schemes are discussed. Relationships between Bayesian estimation techniques and the generalized stochastic approximation procedure are demonstrated. Methods are also suggested for the learning of slowly time-varying parameters. The method of potential functions, because of its close relationship to the stochastic approximation procedure, is briefly presented in Appendix G.

Some of the material in the monograph has been discussed in several short courses at Purdue University, Washington University, and UCLA. Most of the material has been taught in both regular and seminar courses at Purdue University and the University of California at Berkeley. For a regular course in pattern recognition and machine learning, many other approaches should also be discussed. Unfortunately, because of the limited scope of the monograph, those promising approaches cannot be covered in detail here. Instead, a very brief remark on other related approaches and interesting research problems is given in the last section of each chapter. It is no doubt that there are still some works not mentioned even in these remarks due to the author's oversight or ignorance.

K. S. Fu

Lafayette, Indiana
August, 1968

ACKNOWLEDGMENTS

It is the author's pleasure to acknowledge the encouragement of Dr. M. E. VanValkenburg, Dr. L. A. Zadeh, Dr. T. F. Jones, Dr. W. H. Hayt, Jr., Dr. J. C. Hancock, and Dr. J. R. Lehmann. He owes a debt of gratitude to Dr. Richard Bellman, who read the manuscript and contributed many valuable suggestions. The author is also indebted to his colleagues and students at Purdue University and the University of California at Berkeley, who, through many helpful discussions during office and class hours, coffee breaks, and late evenings, assisted in the preparation of the manuscript. Particular suggestions and errata lists were provided by Dr. Z. J. Nikolic and Dr. Y. T. Chien. The author and his co-workers at Purdue have been very fortunate in having the consistent support from National Science Foundation for the research in pattern recognition and machine learning. The major part of the manuscript was completed during the author's sabbatical year (1967) at the Department of Electrical Engineering and Computer Science, University of California, Berkeley. The environment and the atmosphere in Cory Hall and on Telegraph Avenue definitely stimulated the improvement and the early completion of the manuscript. In addition, the author wishes to thank Mrs. Patricia Gress for her efficient and careful typing of the manuscript.

CONTENTS

CHAPTER 1

INTRODUCTION

1.1 Pattern Recognition

The problem of pattern recognition is that of classifying or labeling a group of objects on the basis of certain subjective requirements. Those objects classified into the same pattern class usually have some common properties. The classification requirements are subjective since different types of classifications occur under different situations. For example, in recognizing English characters, there are twenty-six pattern classes. However, in distinguishing English characters from Chinese characters, there are only two pattern classes, i. e., English and Chinese. Human beings perform the task of pattern recognition in almost every level of the nervous system. Recently, engineers faced the problem of designing machines for pattern recognition. Preliminary results have been very encouraging. There have been some successful attempts to design or to program machines to read printed or typed characters, identify bank checks, classify electrocardiograms, recognize some spoken words, play checkers and chess, and sort photographs. Other applications of pattern recognition include handwritten characters or word recognition, general medical diagnosis, system's fault identification, seismic wave classification, target detection, weather prediction, speech recognition, etc. The simplest approach for pattern recognition is probably the approach of "template-matching." In this case, a set of templates or prototypes, one for each pattern class, is stored in the machine. The input pattern (with unknown classification) is compared with the template of each class, and the classification is based on a preselected matching criterion or similarity criterion. In other words, if the input pattern matches the template of *i*th pattern class better than it matches any other template, then the input is classified as from the *i*th pattern class. Usually, for the simplicity of the machine, the templates are stored in their raw-data form. This approach has been used for some existing printed-character recognizers and bank-check readers.

The disadvantages of the template-matching approach is that it is sometimes difficult to select a good template from each pattern class and to define a proper matching criterion. The difficulty is especially remarkable when large variations and distortions are expected in all the patterns belonging to one class. The recognition of handwritten characters is a good example in this case. A more sophisticated approach is that instead of matching the input pattern with the templates, the classification is based on a set of selected measurements extracted from the input pattern. These selected measurements, called "features," are supposed to be invariant or less sensitive with respect to the commonly encountered variations and distortions, and to also contain less redundancies. Under this proposition, pattern recognition can be considered as consisting of two subproblems. The first subproblem is what measurements should be taken from the input patterns. Usually, the decision of what to measure is rather subjective and also dependent on the practical situations (for example, the availability of measurements, the cost of measurements, etc.). Unfortunately, at present there is very little general theory for the selection of feature measurements. However, there are some investigations concerned with the selection of a subset and the ordering of features in a given set of measurements. The criterion of feature selection or ordering is often based on either the importance of the features in characterizing the patterns or the contribution of the features to the performance of recognition (i.e., the accuracy of recognition).

The second subproblem in pattern recognition is the problem of classification (or making a decision on the class assignment to the input patterns) based on the measurements taken from the selected features. The device or machine which extracts the feature measurements from input patterns is called a *feature extractor*. The device or machine which performs the function of classification is called a *classifier*. A simplified block diagram of a pattern recognition system

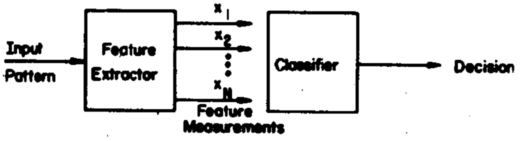

Fig. 1.1. A pattern recognition system.

is shown in Fig. 1.1.[†] Thus, in general terms, the template-matching approach may be interpreted as a special case of the second approach—"feature-extraction" approach, where the templates are stored in terms of feature measurements and a special classification criterion (matching) is used for the classifier.

1.2 Deterministic Classification Techniques

The concept of pattern classification may be expressed in terms of the partition of feature space (or a mapping from feature space to decision space). Suppose that N features are to be measured from each input pattern. Each set of N features can be considered as a vector x, called a feature (measurement) vector, or a point in the N-dimensional feature space Ω_X. The problem of classification is to assign each possible vector or point in the feature space to a proper pattern class. This can be interpreted as a partition of the feature space into mutually exclusive regions, and each region will correspond to a particular pattern class. Mathematically, the problem of classification can be formulated in term of "discriminant functions"[1] Let ω_1, ω_{2}.,..., ω_m be designated as the m possible pattern classes to be recognized, and let

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \tag{1.1}$$

be the feature (measurement) vector where x_i represents the ith feature measurement. Then the discriminant function $D_j(X)$ associated with pattern class ω_j, $j = 1,..., m$, is such that if the input pattern represented by the feature vector X is in class ω_i, denoted as $X \sim \omega_i$, the value of $D_i(X)$ must be the largest. That is, for all $X \sim \omega_i$,

$$D_i(X) > D_j(X), \qquad i,j = 1,\ldots, m, \quad i \neq j \tag{1.2}$$

Thus, in the feature space Ω_X the boundary of partition, called the decision boundary, between regions associated with class ω_i and class ω_j, respectively, is expressed by the following equation

$$D_i(X) - D_j(X) = 0 \tag{1.3}$$

[†] The division of two parts is primarily for convenience rather than necessity.

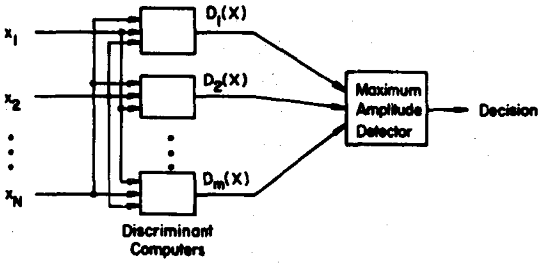

Fig. 1.2. A classifier.

A general clock diagram for the classifier using criterion (1.2) and a typical two-dimensional illustration of (1.3) are shown in Figs. 1.2 and 1.3, respectively. Many different forms satisfying condition (1.2) can be selected for $D_i(X)$. Several important discriminant functions are discussed in the following.

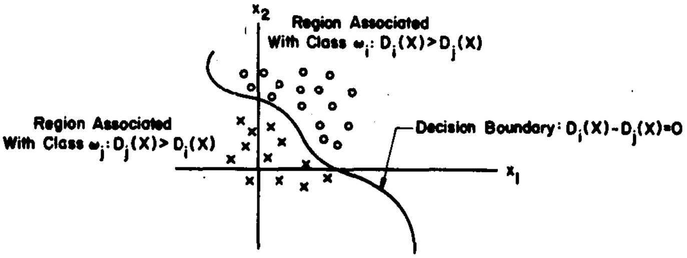

Fig. 1.3. An example of partition in a two-dimensional feature space.

A. *Linear Discriminant Functions*

In this case a linear combination of the feature measurements $x_1, x_2, \ldots, x_N$ is selected for $D_i(X)$, i.e.,

$$D_i(X) = \sum_{k=1}^{N} w_{ik}x_k + w_{i,N+1}, \qquad i = 1,\ldots, m \tag{1.4}$$

The decision boundary between regions in Ω_X associated with ω_i and ω_j is in the form of

$$D_i(X) - D_j(X) = \sum_{k=1}^{N} w_k x_k + w_{N+1} = 0 \tag{1.5}$$

with $w_k = w_{ik} - w_{jk}$ and $w_{N+1} = w_{i,N+1} - w_{j,N+1}$. Equation (1.5) is the equation of a hyperplane in the feature space Ω_X. A general linear discriminant computer is shown in Fig. 1.4. If $m = 2$, on the basis of (1.5), $i, j = 1, 2$ $(i \neq j)$, a threshold logic device as shown in Fig. 1.5 can be employed as a linear classifier (a classifier using linear discriminant functions). From Fig. 1.5, let $D(X) = D_1(X) - D_2(X)$, if

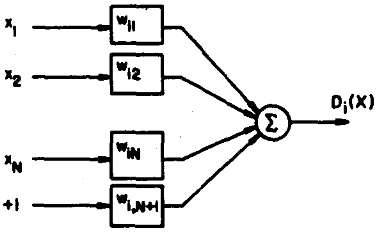

Fig. 1.4. A linear discriminant computer.

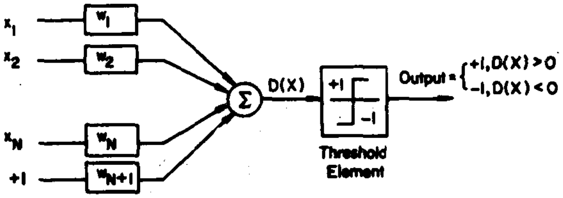

Fig. 1.5. A linear two-class classifier.

$$\begin{aligned} &\text{output} = +1, \quad \text{i.e.,} \quad D(X) > 0, \quad \text{then} \quad X \sim \omega_1 \\ \text{and if} \\ &\text{output} = -1, \quad \text{i.e.,} \quad D(X) < 0, \quad \text{then} \quad X \sim \omega_2 \end{aligned} \tag{1.6}$$

For the number of pattern classes more than two, $m > 2$, several threshold logic devices can be connected in parallel so that the combinations of the outputs from, say, M threshold logic devices will be sufficient for distinguishing m classes when $2^M \geqslant m$. Or, the general configuration of Figs. 1.2 and 1.4 can also be used.

B. *Minimum-Distance Classifier*

An important class of linear classifiers is that of using the distances between the input pattern and a set of reference vectors or prototype points in the feature space as the classification criterion. Suppose that m reference vectors $R_1, R_2, \ldots, R_m$ are given with R_j associated with the pattern class ω_j. A minimum-distance classification scheme with respect to $R_1, R_2, \ldots, R_m$ is to classify the input X as from ω_i, i.e.,

$$X \sim \omega_i \qquad \text{if} \quad | X - R_i | \text{ is the minimum} \tag{1.7}$$

where $| X - R_i |$ is the distance defined between X and R_i. For example, $| X - R_i |$ may be defined as

$$| X - R_i | = [(X - R_i)^T (X - R_i)]^{1/2} \tag{1.8}$$

where the superscript T represents the transpose operation to a vector. From (1.8),

$$| X - R_i |^2 = X^T X - X^T R_i - X R_i^T + R_i^T R_i \tag{1.9}$$

Since $X^T X$ is not a function of i, the corresponding discriminant function for a minimum-distance classifier is essentially

$$D_i(X) = X^T R_i + X R_i^T - R_i^T R_i, \qquad i = 1, \ldots, m \tag{1.10}$$

which is linear. Hence, a minimum-distance classifier is also a linear classifier. The performance of a minimum-distance classifier is of course dependent upon an appropriately selected set of reference vectors.

C. *Piecewise Linear Discriminant Functions*

The concept adopted in Section B can be extended to the case of minimum-distance classification with respect to sets of reference vectors. Let $R_1, R_2, \ldots, R_m$ be the m sets of reference vectors associated with classes $\omega_1, \omega_2, \ldots, \omega_m$, respectively, and let reference vectors in R_j be denoted as $R_j^{(k)}$, i.e.,

$$R_j^{(k)} \in R_j, \qquad k = 1, \ldots, u_j$$

where u_j is the number of reference vectors in set R_j. Define the distance between an input feature vector X and R_j as

$$d(X, R_j) = \operatorname*{Min}_{k=1,\ldots,u_j} | X - R_j^{(k)} | \tag{1.11}$$

That is, the distance between X and R_j is the smallest of the distances between X and each vector in R_j. The classifier will assign the input to a pattern class which is associated with the closest vector set. If the distance between X and $R_j^{(k)}$, $|X - R_j^{(k)}|$, is defined as (1.8), then the discriminant function used in this case is essentially

$$D_i(X) = \max_{k=1,\dots,u_i} \{X^T R_i^{(k)} + (R_i^{(k)})^T X - (R_i^{(k)})^T R_i^{(k)}\}, \qquad i = 1,\dots, m \tag{1.12}$$

Let

$$D_i^{(k)} = X^T R_i^{(k)} + (R_i^{(k)})^T X - (R_i^{(k)})^T R_i^{(k)} \tag{1.13}$$

Then

$$D_i(X) = \max_{k=1,\dots,u_i} \{D_i^{(k)}(X)\}, \qquad i = 1,\dots, m \tag{1.14}$$

It is noted that $D_i^{(k)}(X)$ is a linear combination of features, hence the class of classifiers using (1.12) or (1.14) is often called piecewise linear classifiers [1]. An example of the piecewise linear classifier is the α-perceptron which is shown in Fig. 1.6.

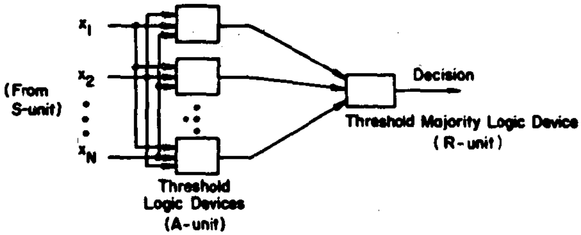
x_1
x_2
(From S-unit)
x_N
Decision
Threshold Majority Logic Device (R-unit)
Threshold Logic Devices (A-unit)

Fig. 1.6. An α-perceptron.

D. *Polynomial Discriminant Functions*

An rth-order polynomial discriminant function can be expressed as

$$D_i(X) = w_{i1} f_1(X) + w_{i2} f_2(X) + \cdots + w_{iL} f_L(X) + w_{i,L+1} \tag{1.15}$$

where $f_j(x)$ is of the form

$$x_{k_1}^{n_1} x_{k_2}^{n_2} \cdots x_{k_r}^{n_r} \qquad \text{for} \quad \begin{cases} k_1, k_2, \dots, k_r = 1, \dots, N \\ n_1, n_2, \dots, n_r = 0 \quad \text{and} \quad 1 \end{cases} \tag{1.16}$$

The decision boundary between any two classes is also in the form of an rth-order polynomial. Particularly, if $r = 2$, the discriminant function is called a quadric discriminant function.

In this case,

$$f_i(X) = x_{k_1}^{n_1} x_{k_2}^{n_2} \qquad \text{for} \quad k_1, k_2 = 1, \dots, N, \quad n_1, n_2 = 0 \text{ and } 1 \tag{1.17}$$

Typically,

$$D_i(X) = \sum_{k=1}^{N} w_{kk} x_k^2 + \sum_{j=1}^{N-1} \sum_{k=j+1}^{N} w_{jk} x_j x_k + \sum_{j=1}^{N} w_j x_j + w_{L+1} \tag{1.18}$$

$$L = \tfrac{1}{2} N(N+3) \tag{1.19}$$

In general, the decision boundary for quadric discriminant functions is a hyperhyperboloid. Special cases included hypersphere, hyperellipsoid, and hyperellipsoidal cylinder. A general quadric discriminant computer is shown in Fig. 1.7.

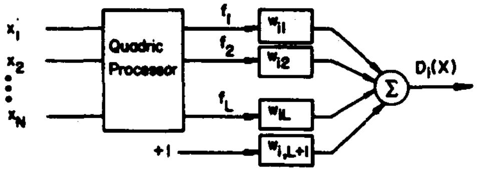

Fig. 1.7. A quadratic discriminant computer.

1.3 Training in Linear Classifiers

The two-class linear classifier discussed in Section 1.2 can easily be implemented by a single threshold logic device. If the patterns from different classes are linearly separable (can be separated by a hyperplane in the feature space Ω_X), then with correct values of the coefficients or weights, $w_1, w_2, \dots, w_{N+1}$ in (1.5), the achievement of a perfectly correct recognition is possible. However, in practice, the proper values of the weights are usually not available. Under such circumstances, it is proposed that the classifier be designed to have the capability of estimating the best values of the weights from the input patterns. The basic idea is that by observing patterns with known classifications, the classifier can automatically adjust the weights in order to achieve correct recognitions. The performance of the classifier is supposed to improve as more and more patterns are abserved. This process is called training or learning, and the