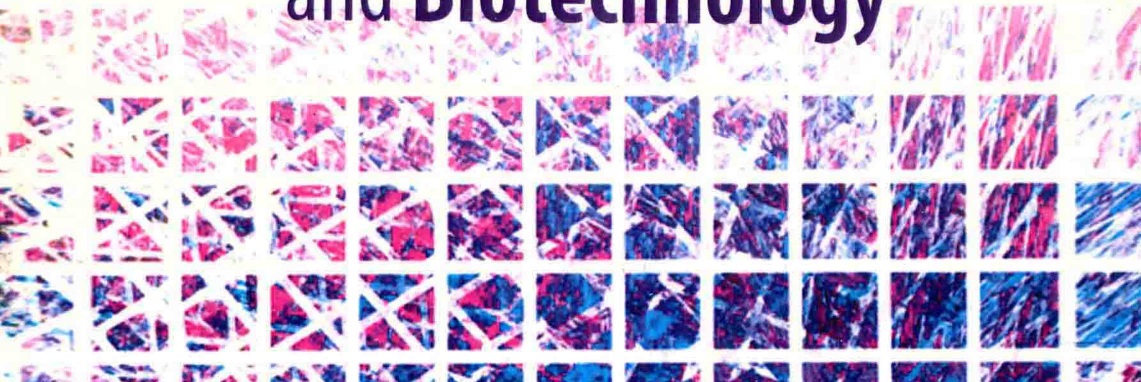
SUPPORT VECTOR MACHINES and Their Application in Chemistry and Biotechnology



Yizeng Liang • Qing-Song Xu
Hong-Dong Li • Dong-Sheng Cao

CRC Press
Taylor & Francis Group

SUPPORT VECTOR MACHINES and Their Application in Chemistry and Biotechnology

Yizeng Liang • Qing-Song Xu
Hong-Dong Li • Dong-Sheng Cao

深圳大学图书馆
藏书章

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

© 2011 by Taylor and Francis Group, LLC
CRC Press is an imprint of Taylor & Francis Group, an Informa business

No claim to original U.S. Government works

Printed in the United States of America on acid-free paper
10 9 8 7 6 5 4 3 2 1

International Standard Book Number: 978-1-4398-2127-5 (Hardback)

This book contains information obtained from authentic and highly regarded sources. Reasonable efforts have been made to publish reliable data and information, but the author and publisher cannot assume responsibility for the validity of all materials or the consequences of their use. The authors and publishers have attempted to trace the copyright holders of all material reproduced in this publication and apologize to copyright holders if permission to publish in this form has not been obtained. If any copyright material has not been acknowledged please write and let us know so we may rectify in any future reprint.

Except as permitted under U.S. Copyright Law, no part of this book may be reprinted, reproduced, transmitted, or utilized in any form by any electronic, mechanical, or other means, now known or hereafter invented, including photocopying, microfilming, and recording, or in any information storage or retrieval system, without written permission from the publishers.

For permission to photocopy or use material electronically from this work, please access www.copyright.com (http://www.copyright.com/) or contact the Copyright Clearance Center, Inc. (CCC), 222 Rosewood Drive, Danvers, MA 01923, 978-750-8400. CCC is a not-for-profit organization that provides licenses and registration for a variety of users. For organizations that have been granted a photocopy license by the CCC, a separate system of payment has been arranged.

Trademark Notice: Product or corporate names may be trademarks or registered trademarks, and are used only for identification and explanation without intent to infringe.

Library of Congress Cataloging-in-Publication Data

Support vector machines and their application in chemistry and biotechnology / Yizeng
Liang ... [et al.].
p. cm.
Includes bibliographical references and index.
ISBN 978-1-4398-2127-5 (hardback)
1. Support vector machines. 2. Chemometrics. I. Liang, Yizeng.

Q325.5.S866 2011
542'.85--dc22 2010041498

Visit the Taylor & Francis Web site at
http://www.taylorandfrancis.com

and the CRC Press Web site at
http://www.crcpress.com

SUPPORT VECTOR MACHINES and Their Application in Chemistry and Biotechnology

Preface

> "Make everything as simple as possible, but not simpler."
>
> **Albert Einstein**

Developed for pattern recognition and later extended to multivariate regression, support vector machines (SVMs) were originally proposed by Vapnik et al. and seem a very promising kernel-based machine-learning method. What distinguishes SVMs from traditional learning methods, in our opinion, lies in their exclusive objective function, which minimizes the structural risk of the model. The introduction of the kernel function into SVMs made the method extremely attractive, because it opened a new door for chemists and biologists to use SVMs to solve difficult nonlinear problems in chemistry and biotechnology through the simple linear transformation technique. The distinctive features and excellent empirical performance of SVMs have drawn chemists and biologists so much that a number of papers, mainly concerned with the applications of SVMs, have been published in chemistry and biotechnology in recent years. These applications cover a large range of meaningful chemical or biological problems, for example, spectral calibration, drug design, quantitative structure–activity and property relationships (QSAR/QSPR), food quality control, chemical reaction monitoring, metabolic fingerprint analysis, protein structure and function prediction, microarray data-based cancer classification, and so on.

However, we should also admit that SVMs are not as widely used as traditional methods such as principal component analysis (PCA) and partial least squares (PLS) in chemistry and metabolomics. In order to efficiently apply this rather new technique to solve difficult problems in chemistry and biotechnology, one should have a sound in-depth understanding of what kind of information this new mathematical tool could really provide and what its statistical properties are. However, the difference in professions makes one feel worlds apart. The gap between the mathematicians and the chemists as well as biologists is actually very large

此为试读，需要完整PDF请访问：www.ertongbook.com

because most chemists and biologists are not quite familiar with the theoretical mathematical language and its abstract descriptions. Undoubtedly, this gap limits the applications of SVMs. It goes without saying that the deeper the understanding of SVMs one has, the better application one may achieve. However, to our best knowledge, there is currently no book that provides chemists and biologists with easy-to-understand materials on what SVMs are and how they work.

Thus it seems urgent to build a much needed bridge between the theory and applications of SVMs and hence lessen and even fill the gap. This book aims at giving a deeper and more thorough description of the mechanism of SVMs from the point of view of chemists and biologists and hence making it easy for those scientists to understand. We believe that we might have found the way to do this. Thus it could be expected that more and more researchers will have access to SVMs and further apply them in the solution of meaningful problems in chemistry and biotechnology. We would like to say the above discussion is our main motivation in writing such a book to construct a bridge between the theory and applications of SVMs.

This book is composed of eight chapters. The first four chapters mainly address the theoretical aspects of SVMs, and the latter four chapters are focused on the applications on the quantitative structure–activity relationship, near-infrared spectroscopy, traditional Chinese medicines, and OMICS studies, respectively.

We would like to acknowledge the contributions of many people to the conception and completion of this book. Lance Wobus of Taylor & Francis Books/CRC Press is gratefully acknowledged for his kind encouragement in writing this book as well as his great help which enabled its publication. Dr. Wei Fan in our group is also acknowledged for providing the near-infrared data as well as the helpful discussions on data analysis. Dr. Liang-Xiao Zhang in our group is acknowledged for his help with the literature.

Yizeng Liang
Changsha, Yuelu Mountain

Author Biographies

Yizeng Liang

Professor Yizeng Liang earned his PhD in analytical chemistry in 1988 from Hunan University, China. From June 1990 to October 1992, he worked at the University of Bergen, Norway on a postdoctoral fellowship from the Royal Norwegian Council for Scientific and Industrial Research (NTNF). In 1994, he received a DPhil from the University of Bergen, Norway. He is now a professor of analytical chemistry and chemometrics, doctoral supervisor, director of the Research Centre of Modernization of Traditional Chinese Medicines, and vice dean of the College of Chemistry and Chemical Engineering, Central South University, China. He is also a council member of the Chemical Society of China; a member of the Analytical Chemistry Commission, Chemical Society of China (since 1995); vice chairman of the Computer Chemistry Committee, Chemical Society of China (since 2001); member of the advisory editorial boards of the international journals *Chemometrics and Intelligent Laboratory Systems* (since 1998), *Near Infrared Analysis* (since 2001), *Journal of Separation Science* (since 2005), and the *Chinese Journal of Analytical Chemistry* (*Fenxi Huaxie,* since 1995); as well as editor of *Chemometrics and Intelligent Laboratory Systems* (since 2007).

Since 1989, Professor Liang has published more than 360 scientific research papers, over 300 of which were published in the source journals of the Scientific Citation Index (SCI) with an *h*-index of 30. In addition, he has published eight books (seven in Chinese and one in English) and has authored six chapters in three English-language books.

Dr. Liang has received a number of awards from various ministries of the Peoples' Republic of China and Hunan Province for his research on chemometric methods for white, gray, and black analytical systems (1994); multicomponent analytical systems (1995); complex multicomponent systems and computer expert systems for structure elucidation (2002 and 2003); and multivariate methods for complex multicomponent systems and their applications to traditional Chinese medicine (2009).

Professor Liang's research interests include analytical chemistry, chemometrics, chemical fingerprinting of traditional Chinese medicines, data mining in chemistry and Chinese medicines, metabolomics, and proteomics, among others.

Qing-Song Xu

Qing-Song Xu is a professor at the Institute of Probability and Mathematical Statistics, School of Mathematical Sciences and Computing Technology, Central South University, China. He obtained his PhD in applied mathematics from Hunan University, China, in 2001.

Qing-Song Xu's main research contributions have been in the field of applied statistics and chemometrics, and he has written dozens of papers in these areas. His current research focuses on applied problems in chemistry, biology, and medicine, in particular data analysis, classification, and prediction problems.

Contents

chapter one

Overview of support vector machines

Contents

1.1 Introduction

Machine learning is a scientific discipline related to the design and development of algorithms that allow computers to learn from the given training data. Generally, the learning algorithms can be classified into two taxonomies: unsupervised learning and supervised learning, according to whether an output vector is needed to supervise the learning process. Supervised learning can be further divided into two types: regression and classification. The former refers to the situation where the output vector consists of continuous value and the latter refers to the case where the output vector denotes the discrete class label of each sample. This book introduces the state-of-the-art supervised learning algorithm, and support vector machines (SVMs), as well as its applications, in chemistry and biotechnology. A huge amount of data, such as vibration spectra, drug activity, OMICS, data from analytical instruments, and microarray experiment-based gene expression profiles have recently been generated in chemistry and biotechnology. How to mine useful information from such data using SVMs is the main concern of this book. In this chapter, the background and key elements of SVM together with some applications in chemistry and biotechnology are briefly described.

1.2 Background

SVMs, developed by Vapnik and his coworkers in the field of computer science, are supervised machine learning algorithms for data mining and knowledge discovery. They stem from the framework of statistical learning theory or Vapnik–Chervonenkis (VC) theory and were originally developed for the pattern recognition problem. To date, VC theory is the most successful tool for accurately describing the capacity of a learned model and further telling us how to ensure the generalization performance for future samples by controlling model capacity. The theory mainly concerns the consistency of a learning process the rate of convergence of a learning process, how to control the generalization performance of a learning process, and how to construct learning algorithms. VC dimension and the structural risk minimization (SRM) principle are the two most important elements of VC theory. VC dimension is a measure of the capacity of a set of functions and the SRM principle can ensure that the learned model can generalize well.

Historically, all the necessary elements that form the theory and algorithm of SVMs have been known since the early 1970s. But it took about 25 years before the concept of SVMs was developed and the spirit of SVMs was systematically elucidated in a formal way in the two fundamental monographs: *Statistical Learning Theory* and *The Nature of Statistical Learning Theory* [1, 2]. As pointed out in the two books, in contrast to traditional learning methods where dimension reduction is performed in order to control the generalization performance of the model, the SVMs dramatically increase the dimensionality of the data and then build an optimal separating hyperplane in the high-dimensional feature space relying on the so-called margin maximizing technique. It's surprising but expected that very excellent performance is observed when SVMs are applied to practical problems such as handwriting recognition.

The distinctive features and excellent empirical performance greatly accelerate the expansion of the SVM idea and further lead to their application in a wide variety of fields, such as credit rating analysis [3], text classification [4], spectral calibration [5,6], QSAR/QSPR [7,8], drug design [9], cancer classification [10], protein structure and function prediction [11,12] and metabolomics [13]. All these contribute to both the theoretic and experimental development of SVM and make it a very active area.

Considering the outstanding performances of SVMs, one may be eager to discover what on earth makes SVMs so powerful. To this end, it's first recommended to have a basic mastering of the necessary mathematics that formulate the theory of SVM, which include, but are not limited to, maximal interval linear classifier, kernel functions, kernel matrix, feature spaces, optimization theory (linear or quadratic programming), dual representations, and so on. Now let's go through these key points step by step.

此为试读,需要完整PDF请访问: www.ertongbook.com

1.2.1 *Maximal interval linear classifier*

In classification, the simplest model is the linear classifier, which was mainly developed in the precomputer age of statistical and machine learning. Even in the current era with rapid computer development, there are still enough reasons to employ the linear model to perform our studies because it is easy for us to understand the latent input/output relationship and to make some further interpretations or statistical inferences based on the established linear model.

Linear regression is the simplest way to build a linear classifier. Typically we are given a dataset of m samples collected into a matrix $\boldsymbol{X}$ of size $m \times p$, where p denotes the number of variables. The class label vector is $\boldsymbol{y}$ with its element "1" standing for the positive class or "–1" standing for the negative class in the binary classification setting. The linear regression model has the following formula:

$$\boldsymbol{y} = \boldsymbol{X}\beta + \boldsymbol{e} \tag{1.1}$$

where β is the unknown regression coefficient vector and $\boldsymbol{e}$ denotes the systematic error. By minimizing the squared error loss function which is in vector form defined as

$$RSS(\beta) = (\boldsymbol{y} - \boldsymbol{X}\beta)^T(\boldsymbol{y} - \boldsymbol{X}\beta) \tag{1.2}$$

where *RSS* stands for residual sum of squares, the least squares solution to the linear model can be easily computed as

$$\hat{\beta} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y} \tag{1.3}$$

Then one can make predictions using the following fitted model.

$$\hat{y} = \boldsymbol{X}\hat{\beta} \tag{1.4}$$

Given a new sample $\boldsymbol{x}_{\text{new}}$ which has not been seen by the fitted model shown in Equation (1.4), let's see how to predict its class. If the computed $\hat{y}_{new}$ by Equation (1.4) is positive, then one can say that $\boldsymbol{x}_{\text{new}}$ should belong to the positive class "1". On the contrary, if $\hat{y}_{new}$ is a minus number, the class of $\boldsymbol{x}_{\text{new}}$ should be predicted as "–1". The prediction rule can be summarized as

$$\textit{Predicted class label} = \begin{cases} 1, & \textit{if } \hat{y}_{new} > 0 \\ -1, & \textit{if } \hat{y}_{new} > 0 \end{cases} \tag{1.5}$$

Let's see an example. We first simulate a two-dimensional dataset of 22 samples, with 15 samples belonging to the positive class (circle marker) and the remaining 7 samples to the negative class (diamond marker). The data are shown in Figure 1.1A. Then a linear regression model is fitted to the data and the resulting linear classifier is also given in Figure 1.1A as a solid line. Apparently in this linearly separable case, all the samples are correctly classified by the constructed linear classifier. The careful reader will find that the classifier is very close to the positive class and may further claim that this classifier is not reliable because the positive sample is easily misclassified as a negative sample if it is contaminated by noise. In other words, this classifier is unfair to the positive samples and somewhat "dangerous." Therefore, it seems necessary for us to develop a "safer" strategy based on which one can establish a "safer" or more reliable classification rule. Next we address this "safer" kind of classifier: a linear classifier with maximal interval.

Figure 1.1B shows the same data as shown in Figure 1.1A. Note that there are altogether three parallel lines. Two dashed lines are located on the boundaries of the two classes of samples and the solid line is in the middle of the two dashed lines. Further suppose that the line in the middle is a candidate classifier. With these assumptions, we can now define the interval of the candidate classifier as the distance between the two dashed lines. Intuitively, this definition has a clear geometrical explanation and is very easy to understand. Naturally, the one whose interval achieves the maximum is defined as the linear classifier with maximal interval. By the way, it should be mentioned here that the interval of a

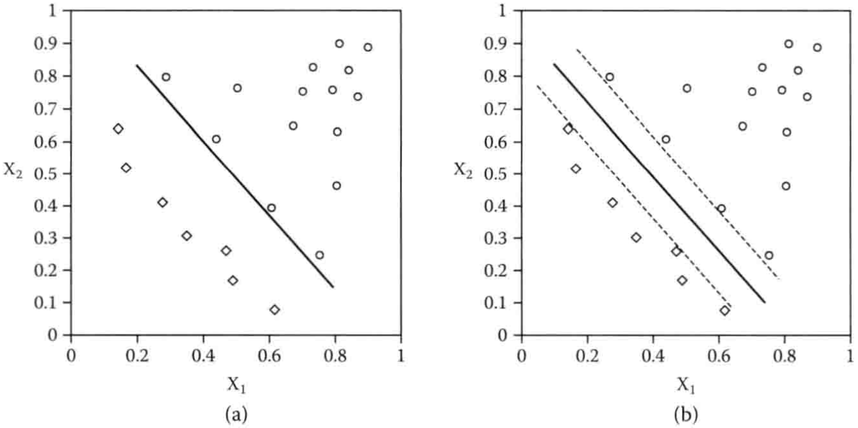


Figure 1.1 Illustration of the linear classifier built by using linear regression (Plot A) and the linear classifier with maximal interval (Plot B).

classifier can also be termed *margin*. In the SVM research, we prefer to use the term *margin* instead of interval.

Compared to the linear classifier in Figure 1.1A, one can easily find that the linear classifier with maximal interval should be the best choice in that it has the larger capacity to tolerate the noise or error. Intuitionally, it is the safest one. Of course, it is necessary for us to know how the maximal interval classifier can be computed because it is very important for developing SVMs. We can perhaps call it the predecessor of SVMs. However, we do not show the computational procedure here. Readers can refer directly to Chapter 2 for detailed information on both mathematics and computations.

Let's reconsider the example in Figure 1.1. In this case, the simulated data are linearly separable and the margin (interval) between classes can be easily defined in geometrics. However, the reader may ask questions such as, "Does the notion of maximal interval linear classifier still work if the data are linearly inseparable? If it does, where is the linear classifier and what is the margin?" It is not possible for us to find such a linear classifier in the exterior, let alone the one with maximal interval. However, it is shown by taking a simple but enlightening example where the linear classifier does exist in the so-called feature space produced by kernel functions: another key element for developing SVM.

1.2.2 Kernel functions and kernel matrix

To intuitively understand the notion of a kernel function, let's first see another dataset shown in Figure 1.2. The data in each class are distributed in two circular regions. Each sample belongs to either the positive class (plus marker) or negative class (asterisk marker) and they cannot be separated well using a linear classifier in the 2-D space (Figure 1.2A). One way to solve this problem is to construct very complicated nonlinear models, for example, ANN. However, it should be noticed that the adjustment of tuning parameters of such kinds of model is usually a time-consuming and tedious task. Moreover, the learned nonlinear discriminating function is most often so flexible that it is difficult for one to ensure its generalization performance. However, the other feasible and effective solution is simply to increase the dimension of the data.

In this case, let's increase the dimension of each sample by one. For the *i*th sample $\mathbf{x}_i = [x_{i1}, x_{i2}]$, the value of the third dimension can simply be calculated as $x_{i3} = x_{i1}^2 + x_{i2}^2$. Thus, in the 3-D space in Figure 1.2B, the *i*th sample can be denoted by $\mathbf{x}_i = [x_{i1}, x_{i2}, x_{i3}]$. Indeed, this operation realizes a nonlinear mapping of the original data from the input space (original low-dimensional variable space before increasing dimensions) into feature space (higher-dimensional space after increasing dimensions). This operation, usually called feature mapping, is primarily an implicit

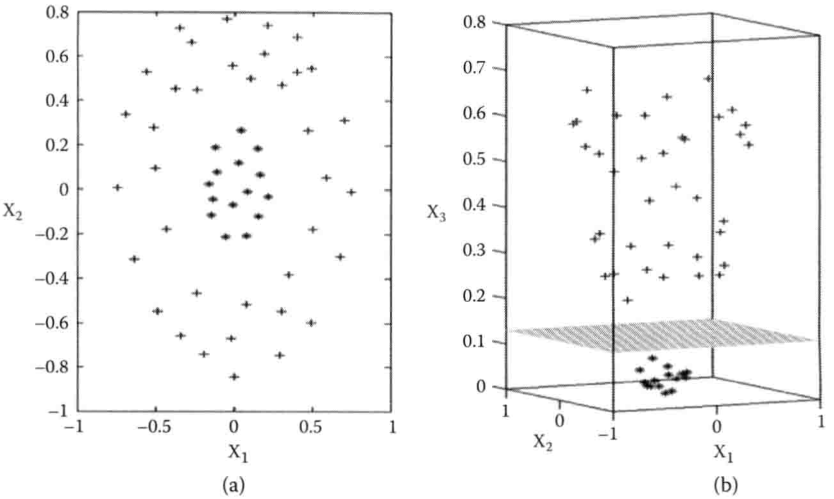


Figure 1.2 The two plots ((A) and (B)) show that the two classes of linearly inseparable samples (plus and asterisk) in a two-dimensional space can be separated linearly without errors when the third dimension is added to each sample. It should be mentioned that the operation of increasing data dimension can be easily and explicitly implemented by using the kernel function.

characteristic of the kernel function. Obviously, it can be seen that the linearly inseparable samples in 2-D space can be separated by a linear hyperplane without any errors.

But we face some problems. For instance: how could we efficiently choose the function to compute the additional dimension? Can we ensure the dimension-increased data are linearly separable? Do we have to know explicitly the function for adding dimensions? We illustrate that the kernel function does provide a smart solution to this problem. It serves as a dimension-increasing technique and further transforms the linearly inseparable data into linearly separable data in the feature space. More interesting, with the help of kernel functions, it's not even necessary for us to know the mathematical form of the functions for adding dimension. However, kernel function details are not delineated in this chapter. But a brief introduction is necessary. Please consult Chapters 2 and 3 for both theoretical and computational details for kernel functions.

Briefly, a kernel function is primarily a symmetric mathematical function that has the general form shown in Equation (1.6),

$$\boldsymbol{K}(\boldsymbol{X}_i, \boldsymbol{X}_j) = <\varphi(\boldsymbol{X}_i), \varphi(\boldsymbol{X}_j)>, i, j = 1,2,3...m \tag{1.6}$$

where $<\cdot>$ denotes the inner product and $\varphi(\cdot)$ is a set of mapping functions that can project the original samples into a high-dimensional feature space. That is to say, $\varphi(\boldsymbol{X}_i)$ is a vector of higher dimension than $\boldsymbol{X}_i$. Furthermore, for each pair of samples, one can compute an inner product using Equation (1.6). All the inner products are now collected into a matrix **K** with elements

$$\mathbf{K}_{ij} = K(\boldsymbol{X}_i, \boldsymbol{X}_j) \tag{1.7}$$

This matrix **K** is the so-called kernel matrix, which without any exception is a key point of all the kernel-based algorithms. From this perspective, SVMs are just a special case of kernel methods. As is known, the inner product is a measurement of the similarity between two samples. In this sense, each element of the kernel matrix reflects the similarity between the samples in the feature space produced by $\varphi(\cdot)$.

So far, we have at least a basic understanding of kernel functions and the associated kernel matrix. Now it is time for us to determine what properties of a function $K(\boldsymbol{X}_i,\boldsymbol{X}_j)$ are necessary to ensure that it is a kernel function for some feature space. According to Mercer's theorem, assume that $\boldsymbol{X}$ is a finite input space with a symmetric function on $\boldsymbol{X}$. It becomes a kernel function if and only if the resulting kernel matrix **K** is positive semidefinitive, that is, without negative eigenvalues.

1.2.3 Optimization theory

Optimization theory is very important for SVMs because the computation of the SVM model can be converted to find the solution of a corresponding optimization problem. In the area of optimization, the most frequent cases we are confronted with may be those of constrained optimization. A special constrained case is convex optimization where the feasible solution of the optimization problem is convex, meaning that any connecting line between two points of the feasible region still falls in the region. There are many freely available programs written in C++, MATLAB®, or R for solving convex problem. But how are SVMs related to the optimization problem?

SVMs work mainly in three steps: (1) using a given kernel function to transform the original data into the feature space; (2) mathematically defining a general linearly separating hyperplane associated with a margin in the feature space, and further establishing a convex optimization problem with maximizing the margin as the objective function; and (3) finding the solution to the convex problem. The solution is just the linear classifier in the feature space with maximal margin, which is the so-called SVMs classifier. It is also called the optimal separating hyperplane (OSH).

As we know, the Lagrange multiplier method is a famous technique for solving optimization problems. In the case of SVMs, the problem of maximizing the margin can also be solved by this method. Without loss of generality, we assume that the derived optimization problem with Lagrange multipliers $\boldsymbol{\alpha}$ introduced is

$$\text{Maximize: } f(w,\boldsymbol{\alpha}) \tag{1.8}$$

Here w denotes the primal variables of a SVM model. By deriving (1.8) against w and setting it to zero, one can derive an equivalent optimization problem with w eliminated. That is,

$$\text{Maximize: } g(\boldsymbol{\alpha}) \tag{1.9}$$

In (1.8) and (1.9), $\boldsymbol{\alpha}$ is called the dual variable. Formula (1.8) is called the primal problem, and Formula (1.9) is called the dual problem corresponding to (1.8). This characteristic is the so-called duality. Only the general notion is given here. In Chapter 2, we show this in great detail.

The Lagrangian treatment of convex optimization problems results in an alternative dual representation, which in most cases turns out to be much cheaper to compute than the primal problem. The reason for this is that the dual problem is just a convex problem with simpler constraints and can be solved easily using quadratic programming (QP). By the way, dual strategies play a central role in kernel methods, such as kernelized Fisher discriminant analysis (KFDA) [14–17] and kernelized partial least squares (KPLS) [18,19], and so on. Support vector machines are special kernel methods that possess some distinguished properties, such as margin maximization and sparsity, which are discussed later.

1.3 Elements of support vector machines

In the last section, we introduced the foundations of support vector machines. Here, we string them together, trying to give an overall picture of SVMs. Figure 1.3 shows the basic procedures for computing a SVM model. Summing up, a SVM model is the mathematical solution of a convex optimization problem whose objective is to maximize the margin of the linear classifier in the feature space produced by the user-chosen kernel function. It should be emphasized here, according to William S. Noble [20], that the basic idea behind the SVM classifier can be explained without ever reading an equation. To understand the essence of the SVM classifier, one only needs to grasp four concepts: kernel function, feature space, separating hyperplane, and optimization problem. Once again, it

此为试读,需要完整PDF请访问: www.ertongbook.com