High Dimension, Low Sample Size Data Analysis

Jeongyoun Ahn

A dissertation submitted to the faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Statistics and Operations Research.

Chapel Hill
2006

Approved by

Advisor: Dr. J. S. Marron

Reader: Dr. Keith E. Muller

Reader: Dr. Hao Zhang

Reader: Dr. Yufeng Liu

Reader: Dr. Haipeng Shen

ABSTRACT

JEONGYOUN AHN: High Dimension, Low Sample Size Data Analysis
(Under the direction of Dr. J. S. Marron)

This dissertation consists of three research topics regarding High Dimension, Low Sample Size (HDLSS) data analysis. The first topic is a study of the sample covariance matrix of a data set with extremely large dimensionality, but with relatively small sample size. Especially the asymptotic behavior of eigenvalues and eigenvectors of the sample covariance matrix is the focus of our study. Assuming that the true population covariance matrix of the data is not too far from identity matrix (i.e., spherical in the Gaussian case), we show that the sample eigenvalues and eigenvectors tend to behave as if the true structure of the data is indeed from identity covariance. Based on this, the asymptotic geometric representation of HDLSS data is extended to a wide range of underlying distributions. The representation essentially states that data vectors form a regular simplex in the data space with the number of vertices equal to the sample size.

The second part of the dissertation studies a discriminant direction vector, which is only interesting in HDLSS settings. This direction is characterized by the property that it projects all the data vectors, which are generated from two classes, to two distinct values, one for each class. It will be seen that this Maximal Data Piling (MDP) direction lies within the hyperplane generated by all the data vectors, while it is orthogonal to the hyperplanes generated by each class. It has the largest distance between piling sites among all the possible piling direction vectors and also maximizes the amount of piling. The formula of MDP is equivalent to the Fisher's linear discrimination when the dimension is less than the sample size. As a classification method, MDP is heuristically desirable when the data are well approximated by the HDLSS geometric representation.

The third topic relates to kernel methods in statistical learning, especially the kernel based classification problem. Taking the case of the Gaussian kernel function for the support vector machines and mean difference methods, we propose a novel approach to select the bandwidth parameter in kernel functions. The derivation is based on the fact that the bandwidth parameter in a kernel function determines the geometry of the high dimensional kernel embedded feature space. Compared with cross-validation and other tuning criteria from the literature, our approach is demonstrated to be robust to the sampling variation, while maintaining comparable classification power and low computing cost, in real and simulated data examples.

ACKNOWLEDGMENTS

This dissertation has been made possible by many people who have supported me. First of all I am very grateful to my advisor Steve Marron who guided this work with many fruitful, insightful discussions. He gave me the chance to participate in several interesting research groups and their research projects.

I would like to express my sincere thanks to Helen Hao Zhang, Keith E. Muller and Yufeng Liu for the opportunities of great collaborations and also their support in the dissertation work. The research assistantship with Harry Hurd in UNC hospitals and Cystic Fibrosis center was an invaluable experience working outside the department.

Finally, I wish to thank my husband Cheolwoo Park and my parents for their support, understanding and love.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Introduction

Data sets with more variables (i.e., attributes or entries in the data vector) than observations are now important in many fields. For example, in genetics a typical microarray gene expression data set has the number of genes ranging from thousands to tens of thousands, while the number of tissue samples (i.e., observations) is less than several hundreds (see for example Golub *et al.* (1999) and Furey *et al.* (2000)). Data from medical imaging, and from text recognition also often have a much larger dimension d than the sample size n. The term "High Dimension Low Sample Size (HDLSS)" will be used for this type of situation in this dissertation. Other terms such as "large p small n" (from the usage of p for the number of variables), appear in other references such as Bickel and Levina (2004). We view these as synonyms. As the technologies for collecting information are improving, it is quite likely that one will face HDLSS data more frequently in future applications.

The analysis of these types of data has become a serious challenge to the statistical community. Many statistical methods for multivariate analysis, which can be found in textbooks such as Muirhead (1982) and Anderson (2003), seldom work for HDLSS data. In fact, these classical methods rarely provide meaningful results in situations with high dimensional data in practice, even with the sample size somewhat larger than the dimensionality. One of the main reasons for their failures is because there are not enough data to estimate the underlying covariance structure properly. A key element in the classical multivariate analysis is to sphere the data by multiplying the data vector by the squared root inverse of the sample covariance matrix, which cannot be done when the matrix is singular, as is the case for HDLSS data. Bickel and Levina (2004) observed that sometimes simply assuming independence of variables, sometimes colled the "naïve Bayes" approach, can yield better performance than attempting to estimate the whole covariance. Chapter 2 and 3 discuss this topic in detail.

Several methods have been developed which confront the HDLSS challenge. Methods that reduce a high dimension to a lower, more manageable dimension have been developed over a number of years. For example, the dimension reduction techniques based on principal component analysis (PCA) find a set of new variables explaining most of the variability in the data. The number of these principal components can be as large as $n - 1$, but usually one expects it to be much smaller than n or d, in practice. Feature selection or variable selection methods find a subset of variables that are most relevant to the objective of analysis. In supervised learning problems such as regression or classification, a popular solution to the model unidentifiability problem caused by the singularity of the covariance, is to put a restriction on the model space. For example, a smoothing constraint on coefficients can be added to the objective function (Hastie and Tibshirani, 2004).

Most methods discussed above were originally developed under the assumption of $d < n$. They have been applied to the HDLSS cases because it is plausible and seems appropriate. However, they do not consider the characteristics of HDLSS data which are quite different from typical non-HDLSS data. Especially HDLSS data are known to have some surprising and often counter-intuitive geometrical structure (Hall *et al.*, 2005). For example, mean zero Gaussian random samples in a very high dimensional space are hardly present near the population mean. Not only that, they also tend to be farther away from the origin as the dimension increases, which appears paradoxical since the Gaussian density is largest near the origin. More detailed discussion of this topic can be found in Chapter 2.

Even though challenging, the visualization of HDLSS data is helpful for understanding of the geometrical structure. Many methods have been proposed such as parallel coordinates, which uses a curve connecting the d values to display a d-length vector and then overlay n such curves to show the entire data set (Inselberg, 1985). Another popular method is to project the data onto one, two, or three dimensional subspaces and look at the projected data (Buja *et al.*, 1996). Especially the so-called draftsman's plot (Tukey and Tukey, 1981) has been proposed as a useful display of low dimensional projections. While these methods can be used to show high dimensional data, it is still possible that they may miss some important high dimensional structures.

This dissertation takes geometrical approaches to several statistical problems regarding HDLSS data analysis. Each of the problems is introduced along with a literature review and discussed in detail in separate chapters.

In Chapter 2, a non-classical type of asymptotics of the sample covariance matrix from HDLSS data is studied. Some of the results imply a shortcoming of PCA applied in HDLSS settings. Also, the asymptotical geometric representation of HDLSS data, which was first introduced in Hall *et al.* (2005), is established under much milder assumptions than the original work.

The binary discrimination problem in HDLSS settings is considered in Chapter 3. In particular a phenomenon called "data piling" (Marron *et al.*, 2005) is explored. A direction vector in the data space that maximizes data piling is the main topic of the chapter and is discussed with a careful mathematical treatment and geometrical characterization. Also this direction is compared with some other popular discriminant directions as a linear classification method.

In Chapter 4, a bandwidth parameter selection problem in the kernel based classification is considered. We treat a nonlinear classification as a linear one in the kernel embedded feature space and propose to choose the bandwidth that makes this linear classification task "easier". The geometry of the embedded data in the infinite dimensional feature space is taken into account to develop a criterion for a measure of "easiness". The proposed method is shown to be more robust to sampling variation than the classical cross-validation approach. The Gaussian Radial Basis Function kernel is used to demonstrate how the proposed method compare with the support vector machine and the mean difference methods as classification methods.

CHAPTER 2

HDLSS Asymptotics: Sample Covariance Matrices and the Geometric Representation

2.1 Introduction

Asymptotic studies regarding HDLSS data are of increasing interest to many researchers. While the classical asymptotic analysis deals with the sample size increasing with fixed dimensionality, an important type of asymptotics, which is more relevant for HDLSS data, studies the case where the dimension d increases. Sample size n can grow with d at the same rate or it can be fixed. We will use notations such as n-, (d, n), and d-asymptotics to denote these three different kinds of asymptotics: The traditional large sample asymptotics will be denoted by n-asymptotics, the asymptotics for both simultaneously increasing dimension and sample size will be denoted by (d, n)-asymptotics, and finally the asymptotics for increasing dimension with a fixed sample size will be denoted by d-asymptotics.

Among various subjects that could be studied these types of asymptotics, particularly there has been substantial interest in the HDLSS properties of the sample covariance matrix. One of the reasons is because many multivariate statistical analysis methods inevitably use the sample covariance matrix, such as by transforming the data vectors by multiplying by the root inverse of the sample covariance matrix. Most of the HDLSS studies on the covariance matrix have had a focus on the asymptotic properties when both dimension and sample size go to infinity together, i.e., the (d, n)-asymptotic properties. Also it is usually assumed that d and n increase at the same rate, i.e., the ratio d/n goes to some constant $\gamma \in (0, \infty)$. Section 2.2 will be an overview of some of the (d, n)-asymptotic results on the sample covariance matrix. See Fujikoshi (2004) for a useful survey of some other (d, n)-asymptotic results on multivariate analysis, such as MANOVA and discriminant functions.

Unlike the aforementioned works, Hall *et al.* (2005) let only d tend to infinity, i.e., take a d-asymptotic approach. They examine the geometry of HDLSS data and showed that under some conditions on the underlying distribution, pairwise distances between the data vectors become a constant as d grows with a fixed n. This geometric representation essentially means that the randomness of data, with an extremely high dimensionality and small sample size, only lies in random rotations of a regular n-simplex in $\mathcal{R}^d$. (See Section 2.5 for details.) In their paper, they applied this to the binary classification problem and obtained some insights about asymptotic behaviors of some popular discrimination methods such as the support vector machines (Section 3.1.3) and the distance weighted discrimination (Section 3.1.4).

The original work of Hall *et al.* (2005) requires the variables to be "nearly independent" and the condition they imposed views the data entries (variables) as a time series. This assumption has some evident shortcomings. First, it is somewhat too strict because it is common to have a severe collinearity among variables. Second, the condition also depends on the order of the data entries, which can be arbitrary.

In this chapter, d-asymptotic properties of the sample covariance matrix are studied in Section 2.3. From this result, it is shown that the geometric representation in Hall *et al.* (2005) can be extended to a much more general condition in Section 2.5. The new condition is on the population eigenvalues and controls the departure from sphericity by restricting the relative size of dominating eigenvalues.

In the (d, n)-asymptotic limit, it is known that eigenvalues of the sample covariance matrix behave as if the underlying covariance were identity, which is the so-called "phase transition" phenomenon. The sufficient condition for this is that the underlying structure of the data is not far from spherical in the sense that non-unit eigenvalues are not much larger than one (Section 2.2.4). Paul (2005) shows that sample eigenvectors also undergo the same phenomenon. It turns out that this also happens in the d-asymptotic case in Section 2.3.

The phase transition phenomenon makes the PCA with HDLSS data often unreliable (Johnstone and Lu, 2004; Muller *et al.*, 2005). In Section 2.4, an extremely non-spherical population model is presented, for which the phenomenon no longer takes place. The departure from sphericity in this model is so large that the first principal component direction vector converges to the population analog as d grows with a fixed n. However even in this case the first sample eigenvalue still fails to converge to its population counterpart. This model can be considered as an extreme case of the "spiked population model" (Section 2.2.4).

2.2 (d, n)-Asymptotics of Sample Covariance Matrices

Assume a $d \times n$ data matrix $\mathbf{X}$ is from the multivariate Gaussian distribution with mean zero and covariance $\mathbf{\Sigma}$, i.e., each d-length column vector, $\mathbf{x}_j$, $j = 1, \cdots, n$, are independently and identically distributed as $\mathcal{N}_d(\mathbf{0}, \mathbf{\Sigma})$. Denote the sample covariance matrix by $\mathbf{S}$:

$$\mathbf{S} = \frac{1}{n}\mathbf{X}\mathbf{X}^{\mathrm{T}}. \tag{2.1}$$

Note that the sample mean is not subtracted from the data matrix in (2.1). $\mathbf{S}$ is easier to handle than the usual sample covariance matrix with mean subtracted in this setting. Also it is the maximum likelihood estimator of $\mathbf{\Sigma}$. In the following subsections, various topics involving $\mathbf{S}$ when the dimension/sample size ratio d/n approaches some constant $\gamma \in (0, \infty)$ as d increases, are reviewed. During most of those discussions, even though the entries of the matrix $\mathbf{X}$ depends on both d and n, we suppress the indices for the sake of simplicity of notation.

2.2.1 Hypothesis Tests for the Population Covariance Matrix

In this section, two common hypothesis testing problems, for the sample covariance matrix from the Gaussian distribution, are considered.

The first testing problem is whether the underlying distribution is spherical. The testing hypotheses of the sphericity test are

$$H_0 : \mathbf{\Sigma} = \sigma^2\mathbf{I} \quad \mathrm{v}s. \quad H_1 : \mathbf{\Sigma} \neq \sigma^2\mathbf{I}, \tag{2.2}$$

where σ is not specified. The likelihood ratio criterion by Mauchly (1940) is

$$R_1 = \frac{|\mathbf{A}|^{\frac{1}{2}n}}{(\mathrm{tr}(\mathbf{A})/d)^{\frac{1}{2}dn}},$$

where $\mathbf{A} = n\mathbf{S}$. Note that this criterion is degenerate when S is singular, i.e., when $d > n$.

The locally most powerful test for (2.2) is based on the statistic (John, 1971)

$$U = \frac{1}{d}\mathrm{tr}\left[\left(\frac{\mathbf{S}}{\mathrm{tr}(\mathbf{S})/d} - \mathbf{I}\right)^2\right], \tag{2.3}$$

and U is well-defined even when $d > n$.

The second hypothesis testing problem is to test whether the underlying covariance is equal to a given positive definite matrix $\mathbf{\Sigma}_0$. This is equivalent to testing

$$H_0 : \mathbf{\Sigma} = \mathbf{I} \quad vs. \quad H_1 : \mathbf{\Sigma} \neq \mathbf{I},$$

by multiplying the data by $\mathbf{\Sigma}_0^{-1/2}$. The likelihood ratio criterion (Anderson, 2003, Chapter 10) is

$$R_2 = \left(\frac{e}{n}\right)^{\frac{1}{2}dn} |\mathbf{A}|^{\frac{1}{2}n} e^{-\frac{1}{2}\mathrm{tr}(\mathbf{A})},$$

which is also degenerate when $d > n$. A non-degenerate testing criterion (Nagao, 1973), well defined even if the dimensionality exceeds the sample size, is

$$V = \frac{1}{d}\mathrm{tr}\left[(\mathbf{S} - \mathbf{I})^2\right],$$

which can be derived in a similar way as U for the test of $\mathbf{\Sigma} = \mathbf{I}$.

Ledoit and Wolf (2002) studied asymptotic behaviors of the tests based on U and V, as d and n go to infinity together with the ratio d/n converging to a limit $\gamma \in (0, \infty)$. They showed that the sphericity test based on the test statistic U remains consistent as $d/n \to \gamma$, in the sense that the (d, n)-asymptotic limiting distribution of U is the same as its n-asymptotic limiting distribution. However, it no longer holds for the second type of test based on V.

Ledoit and Wolf (2002) modified V and introduced a new test statistic W as follows:

$$W = \frac{1}{d}\mathrm{tr}\left[(\mathbf{S} - \mathbf{I})^2\right] - \frac{d}{n}\left(\frac{\mathrm{tr}(\mathbf{S})}{d}\right)^2 + \frac{d}{n}.$$

W has the same n-asymptotic properties as V and shows better (d, n)-asymptotic behaviors, i.e., is not only consistent in the n-asymptotic sense, but also its n-asymptotic properties remain valid in the limit of $d/n \to \gamma$. Birke and Dette (2003) investigated (d, n)-asymptotic properties of U, V, and W when the ratio d/n goes to either zero or infinity.

2.2.2 Spectral Distribution of Sample Covariance Matrices

Let $\ell_1 \geqslant \cdots \geqslant \ell_d$ be the eigenvalues of a sample covariance matrix $\mathbf{S}$ from the spherical Gaussian distribution, i.e., from $\mathcal{N}_d(\mathbf{0}, \mathbf{I})$. The empirical distribution of these sample eigenvalues is called the empirical spectral distribution of $\mathbf{S}$, and defined by

$$F_d(x) = \frac{1}{d} \times \text{number of elements in } \{i : \ell_i \leqslant x\}.$$

The limiting spectral distribution, i.e., the limit of F_d, if d/n goes to $\gamma \in (0, 1]$ as $d \to \infty$, was first obtained by Marčenko and Pastur (1967). F_d converges to the Marčenko-Pastur distribution F, defined with the density function

$$f(x) = F'(x) = \begin{cases} (2\pi\gamma x)^{-1}\sqrt{(x-a)(b-x)} & a < x < b \\ 0 & \text{otherwise,} \end{cases} \tag{2.4}$$

where $a = (1-\sqrt{\gamma})^2$ and $b = (1+\sqrt{\gamma})^2$. When $\gamma > 1$, this distribution has an additional Dirac measure at $x = 0$ of mass $1 - \frac{1}{c}$.

The survey paper by Bai (1999) has a comprehensive review on the spectral distribution, for both the real and the complex cases. Under some conditions on moments, Bai *et al.* (2003) found out that the order of the convergence in (2.4) is $O_p(n^{-2/5})$ when γ is not close to 1 in the sense that $\gamma < 1 - O(n^{-1/8})$, and $O_p(n^{-1/8})$ when γ is close to 1 in the sense that $\gamma > 1 - O(n^{-1/8})$. The model with $\mathbf{\Sigma}$ that has a few non-unit eigenvalues is called the "spiked population model" (Section 2.2.4). It is known (Silverstein and Bai, 1995) that the Marčhenko-Pastur limit still holds for this case too, with possibly different support depending on the model.

The density curves f and the distribution function F for $\gamma = .1, .5$, and 1, are sketched in Figure 2.1. Note that when γ is close to 1, i.e., $d \approx n$, the eigenvalues have more spread, with the largest approaching 4 and the smallest approaching 0. This density function is more

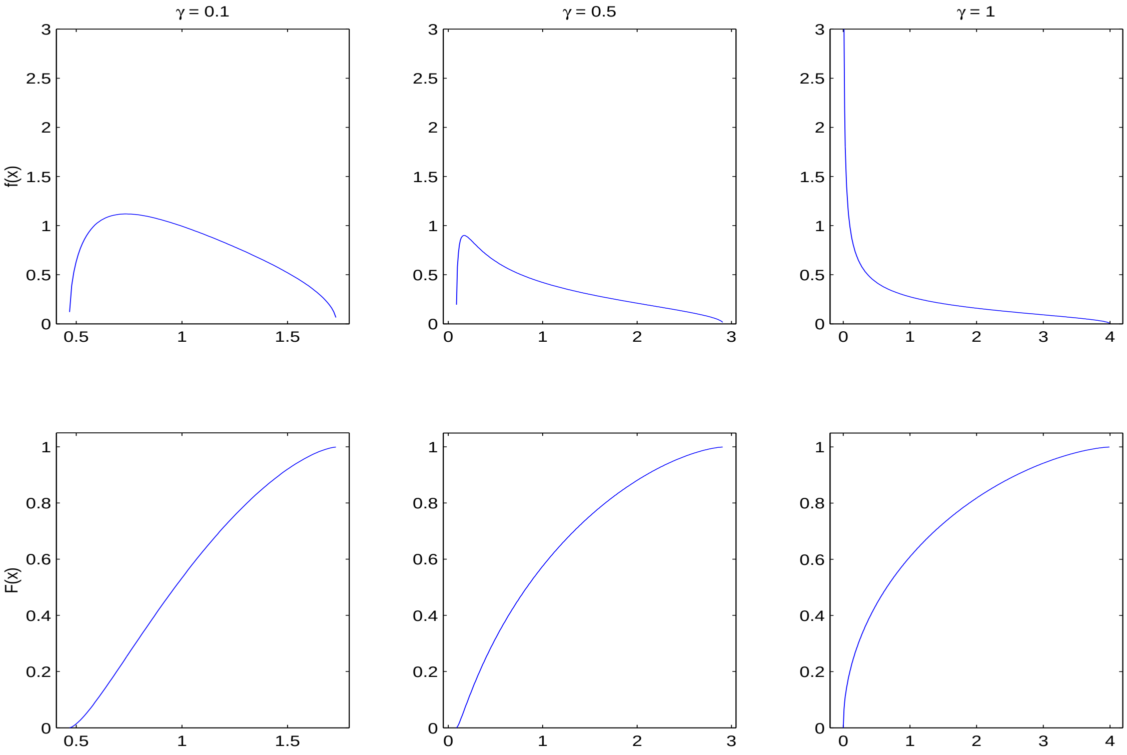

Figure 2.1: *The density and distribution functions of the Marčenko-Pastur distribution.*

skewed than the densities for smaller γ's, with more mass around zero.

Note that this limiting spectral distribution is useful in obtaining the asymptotic behavior of a function of the form

$$\begin{aligned} T_n &\equiv \frac{1}{d}\{\phi(\ell_1) + \cdots + \phi(\ell_d)\} \\ &= \int_0^\infty \phi(x) dF_d(x), \end{aligned}$$

since it converges to

$$\int_0^\infty \phi(x) dF(x),$$

as $d/n \to \gamma$.

2.2.3 Sample Eigenvalues from the Spherical Distribution

While the results in the previous section are about the whole bulk of the sample eigenvalues, in this and the following sections we consider asymptotic properties of each eigenvalue. The identity covariance matrix is considered here and the non-identity case is considered in the next section.

Geman (1980) showed that, for a spherical Gaussian distribution, the largest sample eigenvalue converges to the upper edge of the support of F in (2.4):

$$\ell_1 \to (1+\sqrt{\gamma})^2 \quad \text{almost surely.} \tag{2.5}$$

An analogous result for the smallest eigenvalue holds (Silverstein, 1985; Bai and Yin, 1993):

$$\ell_{\min\{d,n\}} \to (1-\sqrt{\gamma})^2 \quad \text{almost surely,} \tag{2.6}$$

i.e., it converges to the lower edge of the support. Yin *et al.* (1988) generalized these results to non-Gaussian cases under the assumption of a finite fourth moment. Note that these results do not provide any information on the variability, or the distribution of the largest or the smallest eigenvalues.

For the Gaussian case, the limiting distribution of the largest sample eigenvalue ℓ_1 is derived by Johnstone (2001). If we define the center and scaling constants as follows:

$$\mu_{nd} = (\sqrt{n-1}+\sqrt{d})^2, \tag{2.7}$$

$$\sigma_{nd} = (\sqrt{n-1}+\sqrt{d})\left(\frac{1}{\sqrt{n-1}}+\frac{1}{\sqrt{d}}\right)^{1/3}, \tag{2.8}$$

then as $d/n \to \gamma \in (0,1]$,

$$\frac{n\ell_1-\mu_{nd}}{\sigma_{nd}} \xrightarrow{D} W_1 \sim G_1. \tag{2.9}$$

The limiting distribution function G_1 is defined by

$$G_1(s) = \exp\left[-\frac{1}{2}\int_s^\infty \left\{q(x)+(x-s)q^2(x)\right\} dx\right], \quad s \in R,$$

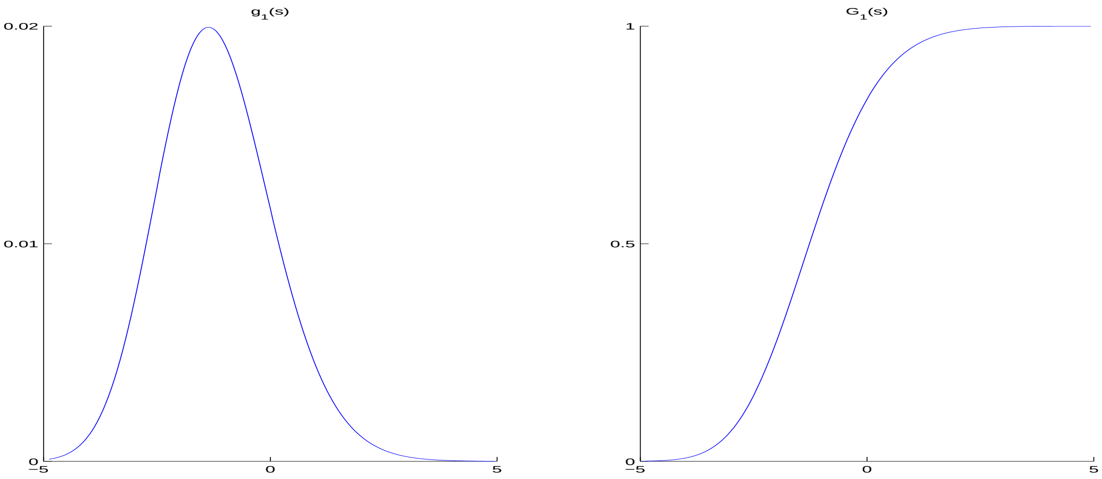

Figure 2.2: *The density and distribution function of the Tracy-Widom distribution* G_1.

where $q(x)$ solves the Painlevè II differential equation

$$\begin{aligned} q''(x) &= xq(x) + 2q^3(x), \\ q(x) &\sim \mathrm{Ai}(x) \quad \text{as} \quad x \to \infty, \end{aligned}$$

and $\mathrm{Ai}(x)$ denotes the Airy function (Deift, 1999). Tracy and Widom (1996) first found this distribution as the limiting law of the largest eigenvalue of an n by n Gaussian symmetric matrix. Note that the mean constant (2.7) divided by n gives $(\sqrt{1-n^{-1}} + \sqrt{d/n})^2$, which is about the same as the limit (2.5) except for a slight adjustment to the quality of approximation for small n.

The distribution G has been numerically evaluated (Prähofer and Spohn, 2003) and plotted in Figure 2.2. It is asymmetric, has mean $\doteq -1.21$ and standard deviation $\doteq 1.27$ and decays like $\exp(-\frac{1}{24}|s|^3)$ at the left tail and $\exp(-\frac{2}{3}s^{3/2})$ at the right. Comparing percentiles of the empirical distribution of the largest sample eigenvalue and its limiting distribution, Johnstone (2001) argued that this approximation is reasonable for both n and d as small as 5.

Principal Component Analysis (PCA) is often performed after each variable has been standardized, because variables are often not on the same scale. Equivalently the sample correlation matrix is used instead of the covariance matrix. In this case, the result in (2.9) does not directly apply for the largest eigenvalue. Johnstone (2001) suggested to synthesize a data matrix

$\widetilde{\mathbf{X}} = [\tilde{\mathbf{x}}_1, \cdots, \tilde{\mathbf{x}}_p]^{\mathrm{T}}$ where $\tilde{\mathbf{x}}_j = r_j \frac{\mathbf{x}_j}{\|\mathbf{x}_j\|}$ and r_j is iid $\frac{1}{n}\chi^2_n$. Now the above approximation holds for the largest eigenvalue of $\frac{1}{n}\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}^{\mathrm{T}}$.

2.2.4 Sample Eigenvalues from the Spiked Population Model

The assumption of identity covariance is sometimes unrealistic since there are many cases where a few sample eigenvalues are separated from the rest of the eigenvalues. Such examples include speech recognition (Johnstone, 2001; Buja *et al.*, 1995), wireless communication (Telatar, 1999), and statistical learning (Hoyle and Rattray, 2004). In this section the spiked population model (Johnstone, 2001), where all but finitely many eigenvalues of the population covariance matrix are one, is considered as the underlying distribution.

Paul (2005) and Baik and Silverstein (2004) independently derived the almost sure limit of the largest sample eigenvalue from the spiked population model. The former work assumes the real Gaussian model, derives the asymptotic distribution of the largest eigenvalue, and also examines asymptotic properties of the corresponding eigenvector. Baik and Silverstein (2004) have a focus on the almost sure limits of the largest and the smallest eigenvalues in real/complex non-Gaussian cases.

Assume that the population distribution is Gaussian with mean zero and diagonal covariance matrix

$$\mathbf{\Sigma} = diag(\lambda_1, \cdots, \lambda_1, \lambda_2, \cdots, \lambda_2, \cdots, \lambda_M, \cdots, \lambda_M, 1, \cdots, 1),$$

where $\lambda_1 > \cdots > \lambda_M > 0$ with multiplicity $k_1, \cdots, k_M$, respectively. Set $k_0 = 0$. Thus if $r = k_1 + \cdots k_M$, then $d - r$ number of eigenvalues are one's. Let us assume that $\ell_1 \geqslant \cdots \geqslant \ell_d \geqslant 0$ are the sample eigenvalues and the ratio d/n converges to γ as $d, n \to \infty$. Baik and Silverstein (2004) showed the following. Note that all the convergence results below are almost sure convergence.

(1) Case $0 < \gamma < 1$.

Let M_0 be the number of j's such that $\lambda_j > 1 + \sqrt{\gamma}$, and let $M - M_1$ be the number of

j's such that $\lambda_j < 1 - \sqrt{\gamma}$. Then, for each $1 \leqslant j \leqslant M_0$,

$$\ell_{k_1+\cdots+k_{j-1}+i} \to \lambda_j \left(1 + \frac{\gamma}{\lambda_j - 1}\right), \quad 1 \leqslant i \leqslant k_j. \tag{2.10}$$

For each $M_1 + 1 \leqslant j \leqslant M$,

$$\ell_{d-r+k_1\cdots+k_{j-1}+i} \to \lambda_j \left(1 + \frac{\gamma}{\lambda_j - 1}\right), \quad 1 \leqslant i \leqslant k_j. \tag{2.11}$$

Also

$$\begin{aligned} \ell_{k_1+\cdots k_{M_0}+1} &\to (1+\sqrt{\gamma})^2, \text{ and} \qquad (2.12) \\ \ell_{d-r+k_1+\cdots+k_{M_1}} &\to (1-\sqrt{\gamma})^2. \qquad (2.13) \end{aligned}$$

Note that (2.10) is for the largest $k_1 + \cdots + k_{M_0}$ eigenvalues and (2.11) is for the smallest $k_{M_1+1} + \cdots + k_M$ eigenvalues. The results (2.10) and (2.12) are proved also by Paul (2005).

(2) Case $\gamma > 1$.

Let M_0 be the number of j's such that $\lambda_j > 1 + \sqrt{\gamma}$. Then, for each $1 \leqslant j \leqslant M_0$,

$$\ell_{k_1+\cdots+k_{j-1}+i} \to \lambda_j \left(1 + \frac{\gamma}{\lambda_j - 1}\right), \quad 1 \leqslant i \leqslant k_j.$$

Also

$$\begin{aligned} \ell_{k_1+\cdots k_{M_0}+1} &\to (1+\sqrt{\gamma})^2, \\ \ell_n &\to (1-\sqrt{\gamma})^2, \text{ and} \\ \ell_{n+1} = \cdots = \ell_d &= 0. \end{aligned}$$

(3) Case $\gamma = 1$.

Let M_0 be the number of j's such that $\lambda_j > 2$. Then for each $1 \leqslant j \leqslant M_0$,

$$\ell_{k_1+\cdots+k_{j-1}+i} \to \lambda_j \left(1 + \frac{\gamma}{\lambda_j - 1}\right), \quad 1 \leqslant i \leqslant k_j.$$

Also

$$
\begin{aligned}
\ell_{k_1+\cdots k_{M_0}+1} &\rightarrow 4, \text{ and} \\
\ell_{\min\{n,d\}} &\rightarrow 0.
\end{aligned}
$$

The overlapping results of Baik and Silverstein (2004) and Paul (2005) say that if the population eigenvalue λ_j is larger than $1+\sqrt{\gamma}$ then the corresponding sample eigenvalue ℓ_j converges to $\lambda_j\left(1+\frac{\gamma}{\lambda_j-1}\right)$ almost surely, and if λ_j is less than $1+\sqrt{\gamma}$ then ℓ_j converges to $(1+\sqrt{\gamma})^2$.

Note that the limits in (2.12) and (2.13) are the same as the limits in (2.5) and(2.6). In other words, the limit of the largest (or smallest) eigenvalue from a non-identity underlying covariance is the same as the limit from the identity covariance. This "phase transition" phenomenon means that if the deviance from sphericity is not strong enough, in the sense that the largest true eigenvalue is less than $1+\sqrt{\gamma}$ or the smallest one is bigger than $1-\sqrt{\gamma}$, then the sample eigenvalue behaves as if it is from the identity covariance.

Now let us consider asymptotic distributions of sample eigenvalues. The (d,n)-asymptotic normality of the largest sample eigenvalue is shown by Paul (2005), in the case of a severe spiked Gaussian population model: If $\lambda_j > 1+\sqrt{\gamma}$ with multiplicity one and $\frac{d}{n}-\gamma = o(n^{-1/2})$, then

$$\sqrt{n}(\ell_j - \mu_j) \xrightarrow{D} N(0,\sigma_j^2),$$

where $\mu_j = \lambda_j\left(1+\frac{\gamma}{\lambda_j-1}\right)$ and $\sigma_j^2 = 2\lambda_j^2\left(1-\frac{\gamma}{(\lambda_j-1)^2}\right)$.

The phase transition phenomenon also happens in sample eigenvectors (Paul, 2005). Suppose the j-th population eigenvector $\mathbf{v}_j$, $j=1,\cdots,d$ is $d\times 1$ vector with 1 in the j-th coordinate and zeros elsewhere and let $\mathbf{e}_j$ be the corresponding sample eigenvector. Then the following results hold when $d/n \rightarrow \gamma \in (0,1)$:

(a) If $\lambda_j \leqslant 1+\sqrt{\gamma}$,

$$< \mathbf{e}_j, \mathbf{v}_j > \xrightarrow{a.s.} 0 \quad \text{as} \quad n,d\rightarrow\infty.$$

(b) If $\lambda_j > 1 + \sqrt{\gamma}$ and of multiplicity one, then

$$|<\mathbf{e}_j, \mathbf{v}_j>| \xrightarrow{a.s.} \sqrt{\left(1 - \frac{\gamma}{(\lambda_j - 1)^2}\right) / \left(1 + \frac{\gamma}{\lambda_j - 1}\right)} \quad \text{as} \quad n, d \to \infty.$$

The inconsistency of principal component analysis in a high dimensional setting has been observed (Johnstone and Lu, 2004). The above result in (a) proves a stronger version of the inconsistency for the mildly spiked population model.

2.3 d-Asymptotic Properties of Sample Covariance Matrices

In this section we examine the asymptotic properties of the sample covariance matrix when only the dimension d tends to infinity while the sample size n is fixed, since the order of magnitude of d is much larger than n in many real data sets.

Suppose we have a $d \times n$ $(d > n)$ data matrix

$$\mathbf{X} \equiv [\mathbf{x}_1, \cdots, \mathbf{x}_n],$$

where $\mathbf{x}_j = (x_{1j}, \cdots, x_{dj})^{\mathrm{T}}$ are iid from a d-multivariate distribution with mean zero and non-negative definite covariance matrix $\boldsymbol{\Sigma}$. Note that $[,]$ is for the horizontal concatenation of vectors/matrices. The eigenvalue decomposition of $\boldsymbol{\Sigma}$ is $\boldsymbol{\Sigma} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^{\mathrm{T}}$, where $\boldsymbol{\Lambda}$ is a diagonal matrix of eigenvalues $\lambda_1 \geqslant \cdots \geqslant \lambda_d > 0$ and $\mathbf{V}$ is the matrix of corresponding eigenvectors. A "factor matrix", which is essentially the square root of $\boldsymbol{\Sigma}$, is defined as $\mathbf{F} \equiv \mathbf{V}\boldsymbol{\Lambda}^{1/2}$, which gives $\boldsymbol{\Sigma} = \mathbf{F}\mathbf{F}^{\mathrm{T}}$. Using the factor matrix $\mathbf{F}$, we can write $\mathbf{X} = \mathbf{F}\mathbf{Z}$ where

$$\mathbf{Z} = \boldsymbol{\Lambda}^{-\frac{1}{2}}\mathbf{V}^{\mathrm{T}}\mathbf{X}$$

is a $d \times n$ random data matrix from d-multivariate distribution with identity covariance matrix. Note that if $\mathbf{X}$ is from the multivariate Gaussian distribution, the elements of $\mathbf{Z}$ are independent standard univariate normal variables.

Using the factor matrix $\mathbf{F}$, the sample covariance matrix $\mathbf{S}$ is decomposed as

$$\mathbf{S} \equiv \frac{1}{n}\mathbf{X}\mathbf{X}^{\mathrm{T}} = \frac{1}{n}\mathbf{F}\mathbf{Z}\mathbf{Z}^{\mathrm{T}}\mathbf{F}^{\mathrm{T}}.$$

A "dual" approach switches the columns and rows of the data matrix, replacing $\mathbf{X}$ by $\mathbf{X}^{\mathrm{T}}$. The $n \times n$ dual sample covariance matrix

$$\mathbf{S}_D \equiv \frac{1}{n}\mathbf{X}^{\mathrm{T}}\mathbf{X}.$$

Note that $\mathbf{S}_D$ has the same eigenvalues as $\mathbf{S}$. If we write $\mathbf{X}$ as $\mathbf{FZ}$ and use the fact that $\mathbf{V}^{\mathrm{T}}\mathbf{V}$ is the identity matrix,

$$\begin{aligned} n\mathbf{S}_D &= (\mathbf{Z}^{\mathrm{T}}\mathbf{F}^{\mathrm{T}})(\mathbf{FZ}) \\ &= \mathbf{Z}^{\mathrm{T}}\boldsymbol{\Lambda}\mathbf{Z} \\ &= \sum_{i=1}^{d} \lambda_i \mathbf{W}_i, \end{aligned} \tag{2.14}$$

where the $n \times n$ matrix $\mathbf{W}_i = \mathbf{z}_i^{\mathrm{T}}\mathbf{z}_i$ and $\mathbf{z}_i$'s, $i = 1, \cdots, d$, are row vectors of the matrix $\mathbf{Z}$. If $\mathbf{X}$ is Gaussian, $\mathbf{W}_i$'s are independently from Wishart distribution $\mathcal{W}_n(1, \mathbf{I}_n)$.

Remark. If a matrix $\mathbf{A}$ has the Wishart distribution $\mathcal{W}_m(\nu, \mathbf{G})$, then $\mathbf{A} = \sum_{i=1}^{\nu} \mathbf{z}_i\mathbf{z}_i^{\mathrm{T}}$, where $\mathbf{z}_1, \cdots, \mathbf{z}_\nu$ are random vectors with an iid m-dimensional multivariate Gaussian distribution with mean 0 and covariance matrix $\mathbf{G}$. See Muirhead (1982) or Anderson (2003) for more details on this distribution.

The main theorem in this chapter states that under mild conditions on the population eigenvalues, $\mathbf{S}_D$ becomes a scaled identity matrix for very large d with a fixed n. Thus all the eigenvalues of $\mathbf{S}_D$ (also those of $\mathbf{S}$) approximately are the same. In a sense, extreme HDLSS data behave as if the underlying distribution were spherical. This means that the phase transition phenomenon, observed for (d, n)-asymptotics in Section 2.2.4 also happens for d-asymptotics.

The assumption for this theorem can be conveniently characterized by a well-known measure of sphericity (Muller *et al.*, 2005)

$$\begin{aligned} \epsilon &\equiv \frac{\mathrm{tr}^2(\boldsymbol{\Sigma})}{d\,\mathrm{tr}(\boldsymbol{\Sigma}^2)} \\ &= \frac{\left(\sum_{j=1}^{d} \lambda_j\right)^2}{d\sum_{i=1}^{d} \lambda_j^2}. \end{aligned} \tag{2.15}$$

The empirical version of (2.15)

$$\hat{\epsilon} = \frac{\mathrm{tr}^2(\mathbf{S})}{d\,\mathrm{tr}(\mathbf{S}^2)},$$

is a locally most powerful invariant test statistic of sphericity of multivariate Gaussian distributions (John, 1972). Note that test statistics (2.3) in Section 2.2.1 can be expressed as a function of $\hat{\epsilon}$:

$$U = \frac{1-\hat{\epsilon}}{\hat{\epsilon}}.$$

Also note that these inequalities always hold:

$$\frac{1}{d} \leqslant \epsilon \leqslant 1.$$

Note that perfect sphericity of the distribution occurs only when $\epsilon = 1$. Our key assumption concerns the other end of the ϵ spectrum, in particular, we need $\epsilon \gg \frac{1}{d}$ for large d, in the sense that $\epsilon^{-1} = o(d)$. In other words, the underlying distribution needs to be *not too close* to the most singular case, where only the first eigenvalue is nonzero.

Theorem 2.3.1. *For a fixed n, consider a sequence of $d \times n$ random matrices $\mathbf{X}_1, \cdots, \mathbf{X}_d, \cdots$ from multivariate distributions with dimension d, with zero means and covariance matrices $\mathbf{\Sigma}_1, \cdots, \mathbf{\Sigma}_d, \cdots$. Let $\lambda_{1,d} > \cdots > \lambda_{d,d}$ be the eigenvalues of the covariance matrix $\mathbf{\Sigma}_d$, and $\mathbf{S}_{D,d}$ be the corresponding dual sample covariance matrix. Suppose the eigenvalues of $\mathbf{\Sigma}_d$ are sufficiently diffused, in the sense that*

$$\frac{1}{d\epsilon} = \frac{\sum_{j=1}^{d} \lambda_{j,d}^2}{\left(\sum_{j=1}^{d} \lambda_{j,d}\right)^2} \to 0 \quad as \quad d \to \infty. \tag{2.16}$$

Then the sample eigenvalues behave as if they are those of the identity covariance in the sense that

$$\frac{\mathbf{S}_{D,d}}{K_d} \to \mathbf{I}_n \quad as \quad d \to \infty,$$

where $K_d = \frac{1}{n}\sum_{j=1}^{d}\lambda_{j,d}$.

Proof. By (2.14), any diagonal element of $n\mathbf{S}_{D,d}$ can be expressed as $\sum_{j=1}^{d}\lambda_{j,d}Z_j^2$ where the Z_j's are iid from univariate distribution with unit variance. Define the relative eigenvalues $\widetilde{\lambda}_{j,d}$ as

$$\widetilde{\lambda}_{j,d} \equiv \frac{\lambda_{j,d}}{\sum_{j=1}^{d}\lambda_{j,d}}.$$

Then by Chebyshev's inequality and the assumption (2.16), for any $\tau > 0$,

$$\begin{aligned}\Pr\left[\left|\sum_{j=1}^{d}\widetilde{\lambda}_{j,d}Z_j^2 - 1\right| > \tau\right] &\leqslant \frac{\operatorname{var}\left(\sum_{j=1}^{d}\widetilde{\lambda}_{j,d}Z_j^2\right)}{\tau^2} \\ &= \frac{2\sum_{j=1}^{d}\widetilde{\lambda}_{j,d}^2}{\tau^2} \to 0 \quad \text{as} \quad d\to\infty.\end{aligned}$$

Thus a diagonal element $\sum_{j=1}^{d}\widetilde{\lambda}_{j,d}Z_j^2$ converges to 1 almost surely.

The off-diagonal elements of $n\mathbf{S}_{D,d}$ can be expressed as $\sum_{j=1}^{d}\lambda_{j,d}Z_jZ_{j'}$ where the Z_j's and the $Z_{j'}$ are independent.

$$\begin{aligned}\Pr\left[\left|\sum_{j=1}^{d}\widetilde{\lambda}_{j,d}Z_jZ_{j'}\right| > \tau\right] &\leqslant \frac{\operatorname{var}\left(\sum_{j=1}^{d}\widetilde{\lambda}_{j,d}Z_jZ_{j'}\right)}{\tau^2} \\ &= \frac{\sum_{j=1}^{d}\widetilde{\lambda}_{j,d}^2}{\tau^2} \to 0 \quad \text{as} \quad d\to\infty.\end{aligned}$$

Thus an off-diagonal element converges to 0 almost surely. □

The condition (2.16) holds for quite general settings, including the following.

(a) Constant: $\lambda_{1,d} = \cdots = \lambda_{d,d} = C$, where C is a constant. This is a spherical case where all the eigenvalues are the same. Since,

$$\frac{\sum_{j=1}^{d}\lambda_{j,d}^2}{\left(\sum_{j=1}^{d}\lambda_{j,d}\right)^2} = \frac{dC^2}{(dC)^2} = \frac{1}{d} \to 0 \quad \text{as} \quad d\to\infty.$$

(b) Fixed Block, Small: $\lambda_{1,d} = \cdots = \lambda_{k,d} = C_1 d^{\alpha}$, $\lambda_{k+1,d} = \cdots = \lambda_{d,d} = C_2$, where

$k < d, \alpha < 1,\ C_1, C_2 > 0$. The first k eigenvalues are moderately larger than the rest. Since,

$$\frac{\sum_{j=1}^{d} \lambda_{j,d}^2}{\left(\sum_{j=1}^{d} \lambda_{j,d}\right)^2} = \frac{kC_1^2 d^{2\alpha} + (d-k)C_2^2}{(kC_1 d^{\alpha} + (d-k)C_2)^2} = \frac{O(d \vee d^{2\alpha})}{O(d^2)} \to 0 \quad \text{as} \quad d \to \infty.$$

(c) Polynomial: $\lambda_{j,d} = j^{-\beta},\ j = 1, \cdots, d,\ \forall \beta > 0$. The eigenvalues decrease in polynomial order. Since,

$$\frac{\sum_{j=1}^{d} \lambda_{j,d}^2}{\left(\sum_{j=1}^{d} \lambda_{j,d}\right)^2} = \frac{\sum_{j=1}^{d} j^{-2\beta}}{(\sum_{j=1}^{d} j^{-\beta})^2} = \frac{O(d^{-2\beta+1})}{O(d^{-2\beta+2})} \to 0 \quad \text{as} \quad d \to \infty.$$

The cases where the condition (2.16) fails include the following.

(d) Fixed Block, Large: $\lambda_{1,d} = \cdots = \lambda_{k,d} = C_1 d^{\alpha},\ \lambda_{k+1,d} = \cdots = \lambda_{d,d} = C_2$, where $k < d, \alpha \geqslant 1,\ C_1, C_2 > 0$. The first k eigenvalues are greatly larger than the rest. Since,

$$\frac{\sum_{j=1}^{d} \lambda_{j,d}^2}{\left(\sum_{j=1}^{d} \lambda_{j,d}\right)^2} = \frac{kc_1^2 d^{2\alpha} + (d-k)C_2^2}{(kC_1 d^{\alpha} + (d-k)C_2)^2} \to C_1 \quad \text{as} \quad d \to \infty.$$

(e) Exponential: $\lambda_{j,d} = \gamma^j,\ j = 1, \cdots, d,\ \forall\ 0 < \gamma < 1$. The eigenvalues decrease exponentially. Since,

$$\frac{\sum_{j=1}^{d} \lambda_{j,d}^2}{\left(\sum_{j=1}^{d} \lambda_{j,d}\right)^2} = \frac{(1-\gamma)^2(1-\gamma^{2d})}{(1-\gamma^2)(1-\gamma^d)^2} \to \frac{1-\gamma}{1+\gamma} \quad \text{as} \quad d \to \infty.$$

(f) Finite Support: $\lambda_{j,d} = C,\ j = 1, \cdots, k,\ \lambda_{k+1,d} = \cdots = \lambda_{d,d} = 0,\ k < d,\ C > 0$. Only the first k eigenvalues are nonzero. Since,

$$\frac{\sum_{j=1}^{d} \lambda_{j,d}^2}{\left(\sum_{j=1}^{d} \lambda_{j,d}\right)^2} = \frac{1}{k}.$$

Example (f) has a singular covariance structure, and (d) and (e) are examples that become nearly singular as the dimension tends to infinity.

2.4 An Extremely Spiked Population Model

For high dimensional data, PCA often fails to estimate the true directions and the variances of principal components, due to the phase transition phenomenon as explained earlier. The condition (2.16) characterizes an underlying structure of the data, which makes the PCA fail in providing reasonable estimates. While this condition is quite mild, there are some distribution models that do not satisfy (2.16). In this section we consider such an extreme case of the spiked population model, in order to see if the PCA can work well under this model. Specifically, we will look at whether the first sample eigenvalue and eigenvector can estimate their population analogs.

We assume that the extremely spiked population models considered here are a sequence of Gaussian distributions, i.e., $\mathcal{N}_d(\mathbf{0}, \mathbf{\Sigma}_d)$, $d = 1, 2, \cdots$. The diagonal covariance matrix $\mathbf{\Sigma}_d$ has a dominating first eigenvalue, i.e.,

$$\mathbf{\Sigma}_d \equiv \mathbf{\Lambda}_d \equiv \mathrm{diag}(d^\alpha, 1, \cdots, 1), \tag{2.17}$$

where $\alpha > 1$. Also the eigenvectors of $\mathbf{\Sigma}_d$ are assumed to be d-dimensional unit vectors. Thus the first eigenvalue and eigenvector for this model are d^α and $(1, 0, \cdots, 0)^{\mathrm{T}}$, respectively. In a factor analysis context, this model corresponds to single common factor model with uniqueness diminishing as $d \to \infty$. It also can be seen as a special case of example (d) in Section 2.3. Note that we already looked at the case where $0 < \alpha < 1$ in the example (b) in that section.

2.4.1 The First Sample Eigenvalue

Let $\ell_{1,d} > \cdots > \ell_{n,d}$ be nonzero eigenvalues of the sample covariance matrix $\mathbf{S}$ (or $\mathbf{S}_D$). By (2.14), the dual sample covariance matrix

$$\begin{aligned}
\mathbf{S}_D &= \frac{1}{n}\sum_{j=1}^{d} \lambda_j \mathbf{W}_j \\
&= \frac{1}{n}\left\{ d^\alpha \mathbf{W}_1 + \sum_{j=2}^{d} \mathbf{W}_j \right\},
\end{aligned}$$

where $\mathbf{W}_j$'s are iid $\mathcal{W}_n(1, \mathbf{I}_n)$. If we define

$$\begin{aligned} \mathbf{U} &\equiv \mathbf{W}_1, \\ \mathbf{V} &\equiv \sum_{j=2}^{d} \mathbf{W}_j, \end{aligned}$$

then $\mathbf{U} \sim \mathcal{W}_n(1, \mathbf{I}_n)$ and $\mathbf{V} \sim \mathcal{W}_n(d-1, \mathbf{I}_n)$, independently. Here the subscript $_{,d}$ is being omitted for the sake of simplicity in notation. Dividing $\mathbf{S}_D$ by d^α gives

$$\frac{1}{d^\alpha}\mathbf{S}_D = \frac{1}{n}\mathbf{U} + \frac{1}{nd^\alpha}\mathbf{V}. \tag{2.18}$$

$\mathbf{U}$ can be expressed as the outer product of a random vector from the $\mathcal{N}_n(\mathbf{0}, \mathbf{I}_n)$ distribution with itself. Thus it has rank one and the only nonzero eigenvalue is the inner product of that random vector with itself, which is a χ^2_n random variable. Also as d becomes large, $\mathbf{V}$ approximates $(d-1)\mathbf{I}_n$. Hence if $\alpha > 1$, (2.18) converges to $\frac{1}{n}\mathbf{U}$ as $d \to \infty$, since the second term $\frac{1}{nd^\alpha}\mathbf{V}$ tends to 0.

The first sample eigenvalue $\ell_{1,d}$ is approximately distributed as $\frac{1}{n}d^\alpha\chi^2_n$ when d is large. This means for fixed sample size n, the first sample eigenvalue is the product of the true eigenvalue and a random quantity depending on the specific realization of data. However, with a reasonably large n, $\frac{1}{n}\chi^2_n$ is expected to be close to 1 by the law of large numbers, which makes the first sample eigenvalue close to the population counterpart. Note that the other eigenvalues converge to 0 as d increases since $\mathbf{U}$ has rank one.

2.4.2 The First Sample Eigenvector

Consider the eigenvalue decomposition of $\mathbf{S} = \mathbf{G}\mathbf{L}\mathbf{G}^{\mathrm{T}}$, where

$$\mathbf{G} = \{g_{ij} : i, j = 1, \cdots, d\}$$

is the matrix of corresponding eigenvectors, $\mathbf{e}_j = (g_{1j}, \cdots, g_{dj})^{\mathrm{T}}, \quad j = 1, \cdots, d$, are the eigenvectors, and

$$\mathbf{L} = \mathrm{diag}(\ell_{1,d}, \cdots, \ell_{n,d}, 0, \cdots, 0).$$

Now writing $\mathbf{\Lambda}_d$ in (2.17) as $\mathbf{\Lambda}$ to simplify the notation, define a standardized version of $\mathbf{S}$ as

$$\begin{aligned}\widetilde{\mathbf{S}} &\equiv \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{S}\mathbf{\Lambda}^{-\frac{1}{2}} \\ &= \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{G}\mathbf{L}\mathbf{G}^{\mathrm{T}}\Lambda^{-\frac{1}{2}}. \end{aligned} \tag{2.19}$$

Using the fact that

$$\begin{aligned}\mathbf{S} &= \frac{1}{n}\mathbf{F}\mathbf{Z}\mathbf{Z}^{\mathrm{T}}\mathbf{F}^{\mathrm{T}} \\ &= \frac{1}{n}\mathbf{\Lambda}^{\frac{1}{2}}\mathbf{Z}\mathbf{Z}^{\mathrm{T}}\mathbf{\Lambda}^{\frac{1}{2}},\end{aligned}$$

we have

$$\begin{aligned}\widetilde{\mathbf{S}} &= \frac{1}{n}\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{\Lambda}^{\frac{1}{2}}\mathbf{Z}\mathbf{Z}^{\mathrm{T}}\mathbf{\Lambda}^{\frac{1}{2}}\mathbf{\Lambda}^{-\frac{1}{2}} \\ &= \frac{1}{n}\mathbf{Z}\mathbf{Z}^{\mathrm{T}} \\ &\sim \frac{1}{n}\mathcal{W}_d(n, \mathbf{I}_d),\end{aligned} \tag{2.20}$$

where $\mathbf{Z}$ is a $d \times n$ data matrix whose elements are iid from the standard univariate Gaussian distribution. Note that $\widetilde{\mathbf{S}}$ is independent of the underlying covariance matrix $\mathbf{\Sigma}_d$ and its diagonal elements are independently distributed as $\frac{1}{n}\chi_n^2$. Also by (2.19), the i-th diagonal entry of $\widetilde{\mathbf{S}}$ is

$$\tilde{s}_{ii} = \frac{g_{i1}^2\ell_{1,d} + \cdots + g_{in}^2\ell_{n,d} + 0 + \cdots + 0}{\lambda_{i,d}}.$$

If we plug in the underlying eigenvalues, then

$$\begin{aligned}\tilde{s}_{11} &= \frac{g_{11}^2\ell_{1,d} + \cdots + g_{1n}^2\ell_{n,d}}{d^\alpha}, \\ \tilde{s}_{jj} &= g_{j1}^2\ell_{1,d} + \cdots + g_{jn}^2\ell_{n,d}, \quad j = 2, \cdots, d.\end{aligned}$$

From the result of Section 2.4.1, for a large d,

$$\tilde{s}_{11} \approx \frac{[\lim_{d\to\infty} g_{11}^2]d^\alpha\chi_n^2}{nd^\alpha},$$

$$\tilde{s}_{jj} \approx \frac{[\lim_{d\to\infty} g_{j1}^2] d^\alpha \chi_n^2}{n}, \quad j = 2, \cdots, d.$$

Since by (2.20) $\tilde{s}_{jj} \sim \frac{1}{n}\chi_n^2$, $j = 1, \cdots, d$,

$$\begin{aligned} g_{11}^2 &\to 1 \quad \text{as} \quad d \to \infty, \\ g_{j1}^2 &\to 0 \quad \text{as} \quad d \to \infty, \quad j = 2, \cdots, d. \end{aligned}$$

Thus, under the identifiability restriction that the first component of the eigenvector must be positive, $\mathbf{e}_1 \to (1, 0, \cdots, 0)^{\mathrm{T}}$, i.e., the first sample eigenvector converges to the population counterpart as d increases.

2.5 HDLSS Geometric Representation

Understanding the geometric structure of HDLSS data is a challenging task due to the limitation of visualizing the data. In fact it is known that they have quite different geometry from low dimensional data. In their (d, n)- asymptotic study on simplices in high dimensional space, Donoho and Tanner (2005) found out that the convex hull of n Gaussian data vectors in $\mathcal{R}^d$ “looks like a simplex” as the ratio d/n converges to $\gamma \in (0, 1)$, in the sense that all points are on the boundary of the convex hull. Furthermore, depending on the value of γ, all the pairwise line segments, triangles, quadrangles, etc., are also on the boundary of the convex hull. In this section we focus on the geometry of HDLSS data using the d-asymptotic approach, letting only d tend to infinity, while fixing n.

Suppose $\mathbf{z}_d = (z_1, \cdots, z_d)^{\mathrm{T}}$ is a d-dimensional random vector from the Gaussian distribution with mean zero and identity covariance matrix. Since the sum of squared entries of $\mathbf{z}_d$ has Chi-square distribution with degrees of freedom d, it can be shown that

$$\|\mathbf{z}_d\| = d^{\frac{1}{2}} + O_p(1).$$

This gives a sense that the data vector lie on the surface of an expanding sphere. If there are two independent vectors from same distribution, $\mathbf{z}_{1,d}$ and $\mathbf{z}_{2,d}$, then the distance between these two is

$$\|\mathbf{z}_{1,d} - \mathbf{z}_{2,d}\| = (2d)^{\frac{1}{2}} + O_p(1). \tag{2.21}$$

Note that data vectors tend to have a deterministic distance apart. Also they are approximately perpendicular because the angle between them

$$\operatorname{ang}(\mathbf{z}_{1,d}, \mathbf{z}_{2,d}) = \frac{1}{2}\pi + O_p(d^{-\frac{1}{2}}). \tag{2.22}$$

Both equations (4.7) and (2.22) hold for n random vectors $\mathbf{z}_{1,d}, \cdots, \mathbf{z}_{n,d}$. This implies that all pairwise distances are approximately equal and all pairwise angles are approximately perpendicular.

Hall *et al.* (2005) extended the above argument to the non-Gaussian case. Suppose $\mathbf{x}_d = (x_1, \cdots, x_d)^{\mathrm{T}}$ is a random vector from a d-dimensional multivariate distribution. Assume the following:

(1) The fourth moments of the entries of the data vectors are uniformly bounded.

(2) For a constant σ^2,

$$\frac{1}{d}\sum_{j=1}^{d} \operatorname{var}(x_j) \to \sigma^2.$$

(3) Viewed as a time series, $x_1, \cdots, x_d, \cdots$ is ρ-mixing for functions that are dominated by quadratics. That is, for $i, j = 1, \cdots, d$ with $|i - j| \geqslant r$,

$$\sup_{|i-j|\geqslant r} |E(x_i x_j)| \leqslant \rho(r) \to 0 \quad \text{as} \quad r \to \infty. \tag{2.23}$$

If $\mathbf{x}_{1,d}, \cdots, \mathbf{x}_{n,d}$ are random vectors from the distribution satisfying the conditions above, then the distance between $\mathbf{x}_{i,d}$ and $\mathbf{x}_{j,d}$, $i \neq j$, is approximately $(2\sigma^2 d)^{\frac{1}{2}}$, in the sense that

$$d^{-\frac{1}{2}}\|\mathbf{x}_{i,d} - \mathbf{x}_{j,d}\| \to (2\sigma^2)^{\frac{1}{2}}, \quad \text{in probability.}$$

Thus after scaling by $d^{-\frac{1}{2}}$, the data vectors $\mathbf{x}_{i,d}$ are asymptotically located at the vertices of a regular n-simplex where all the edges are of length $(2\sigma^2)^{\frac{1}{2}}$.

Hall *et al.* (2005) applied the above result to the two sample case in the context of binary classification. They also obtained some insights about limiting behaviors of some popular discrimination methods such as the support vector machines (see for example Cristianini and Shawe-Taylor (2000)) and the distance weighted discrimination (Marron *et al.*, 2005).

The condition (3) requires entries in the data vector (variables) to be nearly independent. If the entries are far from each other in their locations in the data vector, the correlation between them should diminish. This condition is somewhat too strict because it is common to have a severe collinearity among variables and the condition also depends on the order of the data entries, which can be arbitrary. In this section we establish the same HDLSS geometric representation using Theorem 2.3.1 and show that our assumption (2.16) is more general than that of Hall *et al.* (2005).

Let $\mathbf{x}_{j,d} = (x_{1j}, \cdots, x_{dj})^{\mathrm{T}}$, $j = 1, \cdots, n$, be j-th column of the data matrix $\mathbf{X}$, i.e., j-th sample, from the d-dimensional multivariate distribution with mean zero and covariance matrix $\mathbf{\Sigma}_d$. Suppose the eigenvalues of $\mathbf{\Sigma}_d$ satisfy the condition (2.16). The squared distance between $\mathbf{x}_{i,d}$ and $\mathbf{x}_{j,d}$ is

$$\begin{aligned} \|\mathbf{x}_{i,d} - \mathbf{x}_{j,d}\|^2 &= \sum_{k=1}^{d} (x_{ki} - x_{kj})^2 \\ &= \sum_{i=1}^{d} x_{ki}^2 + \sum_{i=1}^{d} x_{kj}^2 - 2\sum_{i=1}^{d} x_{ki} x_{kj}. \end{aligned} \tag{2.24}$$

Note that the first two terms in (2.24) are the i-th and j-th diagonal entries of $n\mathbf{S}_D$ respectively. Thus for a sufficiently large d, both terms become close to $\sum_{j=1}^{d} \lambda_j$ by Theorem 2.3.1. Also since the third term is the (i, j)-th entry of $n\mathbf{S}_D$, it diminishes to zero as d grows. Thus, for sufficiently large d, the pairwise distances become approximately

$$\|\mathbf{x}_{i,d} - \mathbf{x}_{j,d}\| \approx \left\{ 2 \sum_{j=1}^{d} \lambda_j \right\}^{\frac{1}{2}}.$$

Now let us compare the ρ-mixing condition with (2.16). If (2.23) is satisfied, for $i, j =$

$1, \cdots, d$,

$$E[x_i x_j] \to 0 \quad \text{as} \quad |i-j| \to \infty. \tag{2.25}$$

Note that

$$\begin{aligned}\sum_{k=1}^{d} \lambda_{k,d} &= \sum_{k=1}^{d} E[x_k^2], \quad \text{and} \\ \sum_{k=1}^{d} \lambda_{k,d}^2 &= \operatorname{tr}(\mathbf{\Sigma}_d^2) = \sum_{i,j=1}^{d} E[x_i x_j]^2.\end{aligned}$$

Then the left hand side of (2.16) becomes

$$\frac{\sum_{j=1}^{d} \lambda_{j,d}^2}{\left(\sum_{j=1}^{d} \lambda_{j,d}\right)^2} = \frac{\sum_{i,j=1}^{d} E[x_i x_j]^2}{\sum_{i,j=1}^{d} E[x_i^2]E[x_j^2]} = \frac{o(d^2)}{O(d^2)} \to 0 \quad \text{as} \quad d \to \infty,$$

if all the moments above are bounded. This equation implies that our condition (2.16) is at least as mild as their mixing condition. Note that by a random permutation of the data entries it is easy to make an example that satisfies (2.16) but not the mixing condition. Hence our condition is strictly milder than their ρ-mixing condition.

CHAPTER 3

The Direction of Maximal Data Piling in High Dimensional Spaces

3.1 Introduction to Linear Discrimination

Suppose we are given two classes of objects. We are then faced with a new object with unknown class information, which we are to assign to one of the two classes. The given objects with the known classes are called *training data*. This binary discrimination problem can be formulated as follows. Assume we are given the training data

$$(\mathbf{x}_1, t_1), \cdots, (\mathbf{x}_n, t_n) \in \mathcal{X} \times \{\pm 1\}.$$

Here, $\mathbf{x}_i$ are called inputs or patterns, the class variable t_i are sometimes called targets or labels, and $\mathcal{X}$ is called the feature space where the $\mathbf{x}_i$ are taken from.

It is commonly assumed that $\mathcal{X}$ is d-dimensional Euclidean space $\mathcal{R}^d$ and $\mathbf{x}_i$ are independently and identically distributed random vectors on $\mathcal{R}^d$. We call each dimension of the feature space a feature or a variable. Whenever it is easier to convey a particular idea, we will use $\mathbf{x}$ for the input vectors with Class $+1$ and $\mathbf{y}$ for Class -1.

A discriminating function, or a classifier $f(\mathbf{x})$ is a real valued function of a new data vector $\mathbf{x} \in \mathcal{R}^d$. We assign $\mathbf{x}$ according to the sign of $f(\mathbf{x})$, i.e., if $f(\mathbf{x}) \geqslant 0$ (< 0) then we classify it to the Class $+1$ (-1). The process of obtaining such classifiers is called a classification rule. A possible measure to evaluate a classifier $f(\mathbf{x})$ is the training error

$$\widehat{\text{error}} = \frac{\sum_{i=1}^{n} 1(\text{sign}(f(\mathbf{x}_i)) \neq t_i)}{n},$$

where $1(\cdot)$ is the indicator function. That is, the relative frequency of the training data points which have a discrepancy between the label predicted by $f(\mathbf{x})$ and its actual label. However, we want our classifier to work well eventually for future observations as well as the observations at hand, i.e., to have a good generalizability (Cristianini and Shawe-Taylor, 2000). Thus it is always desirable that we use the misclassification error evaluated from a separate test data set, i.e., the data independently generated from the same distribution as the training data. The misclassification error from the test data can be used to approximate the classification error, which is the expected relative frequency over all the possible test data sets. When the test data set is not available, the V-fold cross validation (Burman, 1989; Wahba *et al.*, 2000) can approximate the test error. See Chapter 4 for more discussion of cross-validation.

Duda *et al.* (2000) gives broad reviews on various discrimination methods, also covering some recent topics in the context of machine learning. Standard textbooks on multivariate data analysis such as Mardia (1980) and Kachigan (1991) have discussions on traditional approaches for discriminant analysis, such as Fisher's linear discrimination. For recently developed classification algorithms such as CART, Hastie *et al.* (2001), Schölkopf and Smola (2001), and Shawe-Taylor and Cristianini (2004) give good overviews of the many references that are available.

We call a classifier linear/nonlinear if the discriminating function $f(\mathbf{x})$ is a linear/nonlinear function of $\mathbf{x}$. In this chapter only linear classifiers are considered. In the classification problem in a HDLSS setting, a linear classifier often gives better performance than most standard nonlinear classifier in many applications (Hastie *et al.*, 2001, Chapter 5), even though the nonlinear classifier rules are generally more flexible (Chapter 4). We can explain this by the HDLSS asymptotics in Section 2.5, since the data vectors approximately form two simplices for each class, where a linear classifier can be a heuristically reasonable choice.

The classification boundary generated by a linear classifier is a linear separating hyperplane between the two classes. A linear classifier in $\mathcal{R}^d$ can be expressed

$$f(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\mathbf{x} + b, \tag{3.1}$$

where the d-column vector $\mathbf{w}$ is the normal vector of the separating hyperplane and the intercept

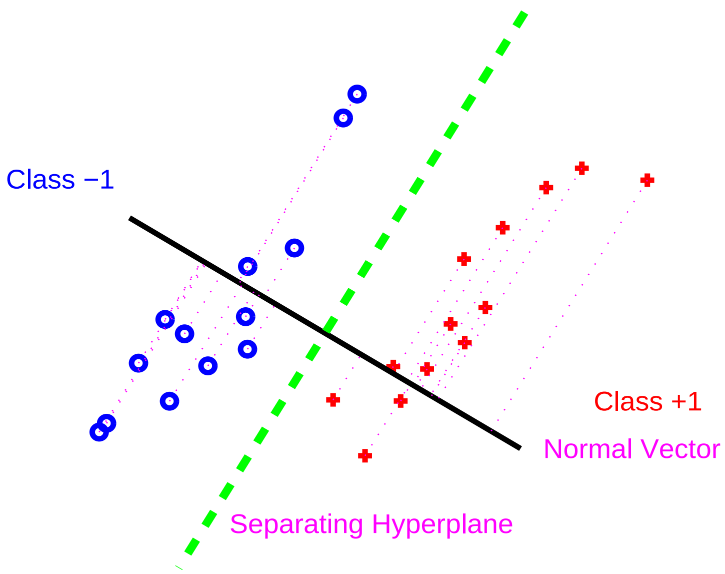

Figure 3.1: *Separable toy data in $\mathcal{R}^2$, with a separating hyperplane, shown as the dashed line, and projections (dotted lines) onto the normal vector (solid line).*

b is called the bias or the shift parameter. Figure 3.1 shows a toy data set in $\mathcal{R}^2$ with a separating hyperplane. It also shows the projections of the data points onto the normal vector of the hyperplane, which will be discussed in Section 3.2. In the following subsections, we will look at some basic and some widely used linear classification methods.

3.1.1 Mean Difference

The Mean Difference (MD) method comes from a very simple idea which uses the sample means as representatives for each class. It is the optimal method when the underlying distributions of the two classes are Gaussian and spherical, and differ only in their means.

The normal direction vector of the mean difference method $\mathbf{w}$ is proportional to the difference vector between the sample means of each class. If we normalize the direction vector,

$$\mathbf{w} \equiv \frac{\bar{\mathbf{x}} - \bar{\mathbf{y}}}{\|\bar{\mathbf{x}} - \bar{\mathbf{y}}\|},$$

where $\bar{\mathbf{x}}$ and $\bar{\mathbf{y}}$ are the sample mean vectors of the training data from each class. Note that the probability of $\mathbf{w}$ being degenerate is zero due to the continuous distribution assumption. The classifier bisects the difference vector $\bar{\mathbf{x}} - \bar{\mathbf{y}}$, thus the threshold b is determined as

$$b = -\mathbf{w}^{\mathrm{T}} \left(\frac{\bar{\mathbf{x}} + \bar{\mathbf{y}}}{2} \right). \tag{3.2}$$

The MD method is also called the nearest centroid method since it classifies a new data vector $\mathbf{x}$ to the class whose centroid (sample mean) is closer to it. Schölkopf and Smola (2001, Chapter 1) gives an overview on this method in the context of machine learning. A recent work with an application to a microarray gene expression analysis can be found in Tibshirani *et al.* (2003). They identified the subsets of genes that best characterize each class and then apply the MD method, called the nearest shrunken centroid method.

3.1.2 Fisher's Linear Discrimination

The Fisher's Linear Discrimination (FLD), also known as linear discriminant analysis, uses the sample covariance structure as well as the sample means. Let $\mathbf{X}_{d \times n_1}$ and $\mathbf{Y}_{d \times n_2}$ be the training data matrices of Class $+1$ and Class -1, respectively. Let $n = n_1 + n_2$ be the total number of training samples. The direction vector of FLD is

$$\mathbf{w} \equiv \frac{\widehat{\boldsymbol{\Sigma}}^{-}(\bar{\mathbf{x}} - \bar{\mathbf{y}})}{\|\widehat{\boldsymbol{\Sigma}}^{-}(\bar{\mathbf{x}} - \bar{\mathbf{y}})\|}, \tag{3.3}$$

where $\mathbf{A}^{-}$ is the Moore-Penrose generalized inverse of the matrix $\mathbf{A}$ and $\widehat{\boldsymbol{\Sigma}}$ is the pooled sample covariance matrix, i.e.,

$$\widehat{\boldsymbol{\Sigma}} = \frac{1}{n-2} \left\{ (\mathbf{X} - \bar{\mathbf{X}})(\mathbf{X} - \bar{\mathbf{X}})^{\mathrm{T}} + (\mathbf{Y} - \bar{\mathbf{Y}})(\mathbf{Y} - \bar{\mathbf{Y}})^{\mathrm{T}} \right\}. \tag{3.4}$$

Here $\bar{\mathbf{X}} = \bar{\mathbf{x}} \mathbf{1}_{n_1}^{\mathrm{T}}$ and $\bar{\mathbf{Y}} = \bar{\mathbf{y}} \mathbf{1}_{n_2}^{\mathrm{T}}$, where $\mathbf{1}_n$ is a n-column vector of ones. The threshold b is determined the same way as in the mean difference method (3.2). Note that the Moore-Penrose generalized inverse operation is equivalent to the regular matrix inverse when the matrix is nonsingular. This can happen for $d \leqslant n - 2$. See Hastie *et al.* (2001, Chapter 5 and 12) for a detailed discussion and some extensions of the FLD.

Note that the Naïve Bayes (NB) discrimination method uses only the diagonal elements of

$\widehat{\mathbf{\Sigma}}$ (Duda *et al.*, 2000, p.62), i.e. using only the variance estimates of the variables. Essentially this method assumes the independence of variables, ignoring the covariance structure between them. Bickel and Levina (2004) showed that in certain HDLSS situations, ignoring covariance structure can give a better asymptotic classification performance than trying to estimate the whole covariance matrix (Section 3.4.3).

3.1.3 Support Vector Machine

While the previous two methods, the MD and FLD, are classical methods, the methods which will be presented in this and the following sections, are relatively recently developed. Let us first assume that the training data set at hand is linearly separable, i.e., there is a linear classifier that can have zero training error. Consider the convex hulls of training data vectors of Class +1 and Class −1, respectively. Define the margin as the minimum distance between these two convex hulls, which can be viewed as the shortest distance between the classes. The Support Vector Machine (SVM) method (Vapnik, 1998; Cristianini and Shawe-Taylor, 2000; Burges, 1998) seeks a separating hyperplane (3.1) that maximizes this margin between the classes.

Let a linear classifier, $f(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\mathbf{x} + b$, be such that $f(\mathbf{x}) \geqslant 1 (\leqslant -1)$ if $\mathbf{x}$ is inside or on the convex hull of Class $+1$ (-1). For this classifier, the margin is $2/\|\mathbf{w}\|$. To obtain the hyperplane maximizing this, one solves the following optimization problem

$$\begin{aligned} &\underset{\mathbf{w},b}{\text{minimize}} \quad \frac{1}{2}\|\mathbf{w}\|^2 \\ &\text{subject to} \quad t_i \cdot (\mathbf{w}^{\mathrm{T}}\mathbf{x}_i + b) \geqslant 1, \quad i = 1, \cdots, n. \end{aligned} \tag{3.5}$$

In many real-world problems, the training data may not be linearly separable, thus the classifier from the above argument may not be plausible. Furthermore, it is common to have noise in the training data set, in the sense that some portion of the data are mislabeled or have measurement errors. Thus it is natural to use a classifier that allows some violations on the margin boundary, which is called the soft margin classifier. The slack variables ξ are introduced to allow the margin constraints (3.5) to be violated. Also the sum of ξ's, the amount of violations, is limited by a constant. Now the problem becomes

$$\begin{aligned} \underset{\mathbf{w},b,\boldsymbol{\xi}}{\text{minimize}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 \\ \text{subject to} \quad & t_i(\mathbf{w}^{\mathrm{T}}\mathbf{x}_i + b) \geqslant 1 - \xi_i, \\ & \xi_i \geqslant 0, \quad \sum_{i=1}^{n} \xi_i \leqslant \text{constant}, \ \ \forall i, \end{aligned} \tag{3.6}$$

which is equivalent to

$$\begin{aligned} \underset{\mathbf{w},b,\boldsymbol{\xi}}{\text{minimize}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i \\ \text{subject to} \quad & t_i(\mathbf{w}^{\mathrm{T}}\mathbf{x}_i + b) \geqslant 1 - \xi_i, \\ & \xi_i \geqslant 0, \ \ \forall i, \end{aligned} \tag{3.7}$$

where $C > 0$ replaces the constant in the previous condition (3.6). C is called the penalty parameter and needs to be chosen. A larger value of C allows less violations thus forces the classifier to have smaller training errors, while a smaller value of C has the opposite effect. This constrained optimization problem in (3.7) can be dealt with by a Lagrangian. The primal Lagrangian is

$$L_P = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i \sum_{i=1}^{n} \alpha_i \left[t_i(\mathbf{w}^{\mathrm{T}}\mathbf{x}_i + b) - (1 - \xi_i)\right] - \sum_{i=1}^{n} \mu_i \xi_i, \tag{3.8}$$

where $\alpha_i \geqslant 0$, $\mu_i \geqslant 0$, $i = 1, \cdots, n$, are the Lagrange multipliers. We minimize L_P with respect to the normal vector $\mathbf{w}$, the intercept b, and the slack variable $\boldsymbol{\xi}$, while maximizing it with respect to $\alpha_i \geqslant 0$ and $\mu_i \geqslant 0$.

The following conditions are obtained by differentiating (3.8):

$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i t_i \mathbf{x}_i \ , \tag{3.9}$$

$$0 = \sum_{i=1}^{n} \alpha_i t_i, \tag{3.10}$$

$$\alpha_i = C - \mu_i, \ \forall i. \tag{3.11}$$

If we substitute (3.9)-(3.11) into (3.8), then the dual Lagrangian is:

$$L_D = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j t_i t_j \mathbf{x}_i^{\mathrm{T}} \mathbf{x}_j \ . \tag{3.12}$$

We maximize L_D subject to $0 \leqslant \alpha_i \leqslant C$ and $\sum_{i=1}^{n} \alpha_i t_i = 0$. The Karush-Kuhn-Tucker conditions for this problem are

$$\begin{aligned} \alpha_i \left[t_i(\mathbf{w}^{\mathrm{T}}\mathbf{x}_i + b) - (1 - \xi_i) \right] &= 0, \\ \mu_i \xi_i &= 0, \\ t_i(\mathbf{w}^{\mathrm{T}}\mathbf{x}_i + b) - (1 - \xi_i) &\geqslant 0, \ \forall i. \end{aligned}$$

With these conditions, the solution to this convex optimization problem is uniquely defined. Note that plugging (3.9) into (3.1), the separating hyperplane becomes

$$f(\mathbf{x}) = \sum_{i=1}^{n} \hat{\alpha}_i t_i \mathbf{x}_i^{\mathrm{T}} \mathbf{x} + \hat{b}, \tag{3.13}$$

where $\hat{\alpha}_i$ and $\hat{b}$ are from solving the above optimization problem. The data vectors with nonzero corresponding α are called the support vectors. The SVM direction vector $\mathbf{w}$ is represented solely by support vectors as seen in (3.9) and consequently the SVM classifier (3.13) only depends on them.

While SVM is one of the most celebrated discrimination methods, there have been many studies to improve the method. For example, Bradley and Mangasarian (1954), Zhu *et al.* (2003) and Zhang *et al.* (2005a) minimize different norms on wv to select important features, and Shen *et al.* (2003) proposed a robust version of SVM.

3.1.4 Distance Weighted Discrimination

The Distance Weighted Discrimination (DWD) method (Marron *et al.*, 2005) is a recently developed classification method that aims primarily at HDLSS data. For a recent application of DWD on the microarray gene expression analysis, see Benito *et al.* (2004).

Given a separating hyperplane of a linear classifier $f(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\mathbf{x} + b$, denote the distance to the hyperplane from the data point $\mathbf{x}_i$ by $\bar{r}_i$, i.e.,

$$\bar{r}_i = t_i(\mathbf{w}^{\mathrm{T}}\mathbf{x}_i + b).$$

The DWD method finds the separating hyperplane that minimizes the sum of the inverse distances, i.e., $\sum 1/\bar{r}_i$. In this way, the data points that are close to the hyperplane have large influence on the decision boundary, and those that are far away have little impact.

Introducing the slack variable $\boldsymbol{\xi}$ for good generalizability, the new distance, whose sum of inverse is to be minimized is

$$r_i = \bar{r}_i + \xi_i = t_i(\mathbf{w}^{\mathrm{T}}\mathbf{x}_i + b) + \xi_i.$$

Thus the DWD optimization problem is

$$\begin{array}{ll} \underset{\mathbf{w},b,\boldsymbol{\xi}}{\text{minimize}} & \sum_{i=1}^{n}(r_i^{-1} + C\xi_i) \\ \text{subject to} & r_i \geqslant 0,\ \xi_i \geqslant 0, \quad \forall i, \end{array}$$

where $C > 0$ is the penalty parameter. This problem can be formulated as a second-order cone programming (SOCP) problem and solved by software packages such as SDPT3 (for Matlab), which is web-available at: http://www.math.nus.edu.sg/∼mattohkc/sdpt3.html.

3.2 The MDP Direction Vector

This section introduces the main topic of this chapter, the Maximal Data Piling direction vector, and demonstrates the vector with a toy data example.

3.2.1 Introduction to MDP

Suppose we have a HDLSS data set with sample size n in d-dimensional Euclidean space $\mathcal{R}^d$ and the underlying distribution of the data is continuous. Let us consider one dimensional projections of the data vectors. There are directions such that all the data vectors project to exactly one point. Since the data vectors generate an n-dimensional subspace, the projections onto any direction in the $(d - n)$ dimensional orthogonal subspace are all zeros. Also, the projections onto any direction vector orthogonal to the $(n-1)$ dimensional hyperplane generated by the data, (i.e., the affine set of the data vectors) are possibly non-zero constants.

Consider the case where we have two data sets from two different continuous distributions in $\mathcal{R}^d$ and the combined sample size n is less than d, i.e., we have a HDLSS binary discrimination problem (Section 3.1). We will show that there exists a direction vector with a property that when the data are projected on that direction, they fall completely on just two points, one for each class. We call this direction the *Maximal Data Piling* (MDP) direction.

In Section 3.3, the formula for the MDP direction vector is characterized as the product of the generalized inverse of the global sample covariance matrix and the mean difference vector. Here, the global sample covariance matrix is obtained by using the global sample mean calculated from the whole data instead of two sample means from each class. In Section 3.3.1, we show that the MDP direction is uniquely determined within the subspace generated by the data, and furthermore, it lies within the hyperplane generated by all the training samples, yet is orthogonal to both of the hyperplanes generated by the separate training samples from each class. Note the formula of MDP only replaces the pooled sample covariance matrix by the global one in the FLD formula. It turns out that these two direction vectors are exactly the same for non-HDLSS data in Section 3.3.2.

In Section 3.4, the classification performance of the MDP is discussed. It is compared with SVM, with a simulated toy example and a microarray gene expression data set in Sections 3.4.1 and 3.4.2. A systematic comparative study to other simple directions in linear classification such as MD, FLD, and NB is done by a simulation in Section 3.4.3. In this simulation we consider a broad range of dimensions with a fixed sample size in a spherical Gaussian setting. It is observed that the MDP shows much better performance than FLD in a very high dimensional space, even slightly better than the NB. The performance of the MDP compared to that of other methods is discussed in detail. In particular, an unexpected behavior of the MDP and FLD, when the dimension is close to the sample size, is discussed.

In Section 3.4.3, the relationship between the MDP direction and the asymptotics of the geometric representation of HDLSS data in Section 2.5 is also discussed. This geometric representation provides the explanation of the good behavior of the MDP for very high dimensional space, along with insights on the other directions for classification.

3.2.2 A Toy Data Example

As it is difficult to visualize HDLSS data, projecting the data vectors to low dimensional spaces, especially to a one dimensional vector or to a two dimensional plane can provide a limited, but useful way to look at the data set. Projections on three direction vectors of interest for a simulated HDLSS data set are shown in Figure 3.2. This toy data set has the dimension $d = 4000$ and sample size $n = 60$, generated from the multivariate Gaussian distribution with identity covariance matrix, with 30 observations from mean $(0.1, \cdots, 0.1)^{\mathrm{T}}$, shown with '+' in the figure, and the other 30 from mean $(-0.1, \cdots, -0.1)^{\mathrm{T}}$, shown with 'o'. The three projection directions are the first coordinate direction (X_1), i.e., $(1, 0, \cdots, 0)^{\mathrm{T}}$, the MDP direction, and the first principal component direction (PC1). Note that PC1 is the direction with the largest variation in the whole data set. With the sample size increasing, the PC1 direction converges to $\frac{1}{\sqrt{d}}(1, \cdots, 1)^{\mathrm{T}}$, which is the optimal direction to discriminate the two classes.

Projections onto each of these three directions are shown in the three diagonal panels. In each diagonal panel, a "jitter plot" (Tukey and Tukey, 1990) is displayed for each class: The projected data are shown with random vertical coordinates for visual separation of the data. Also kernel density estimation curves are drawn to show how the projected values are distributed. The off-diagonal panels show the projections onto the planes spanned by each pair of directions, which are shown with two solid lines in the panel. This type of display is called "draftsman's view".

Let us denote the panel in the r-th row and c-th column by "$[r, c]$". For example, [1,2] is the (top, center) off-diagonal panel showing the projected data on the plane spanned by the X_1 direction and the MDP direction. The first diagonal panel [1,1] of Figure 3.2 indicates that the difference between the two classes is not discernible in the X_1 direction. The projected data on the MDP direction shown in [2,2] are piled up completely at two points, one for each class. The panel [3,3] shows nice separation of the two classes by projecting the data onto the PC1 direction. The panels [1,2] and [2,1] show that the X_1 direction is almost orthogonal to MDP, while [1,3] and [3,1] show it is also nearly orthogonal to PC1, even though the angle with MDP is a little more perpendicular than with PC1 (see [2,1] and [3,1]). The projections to the

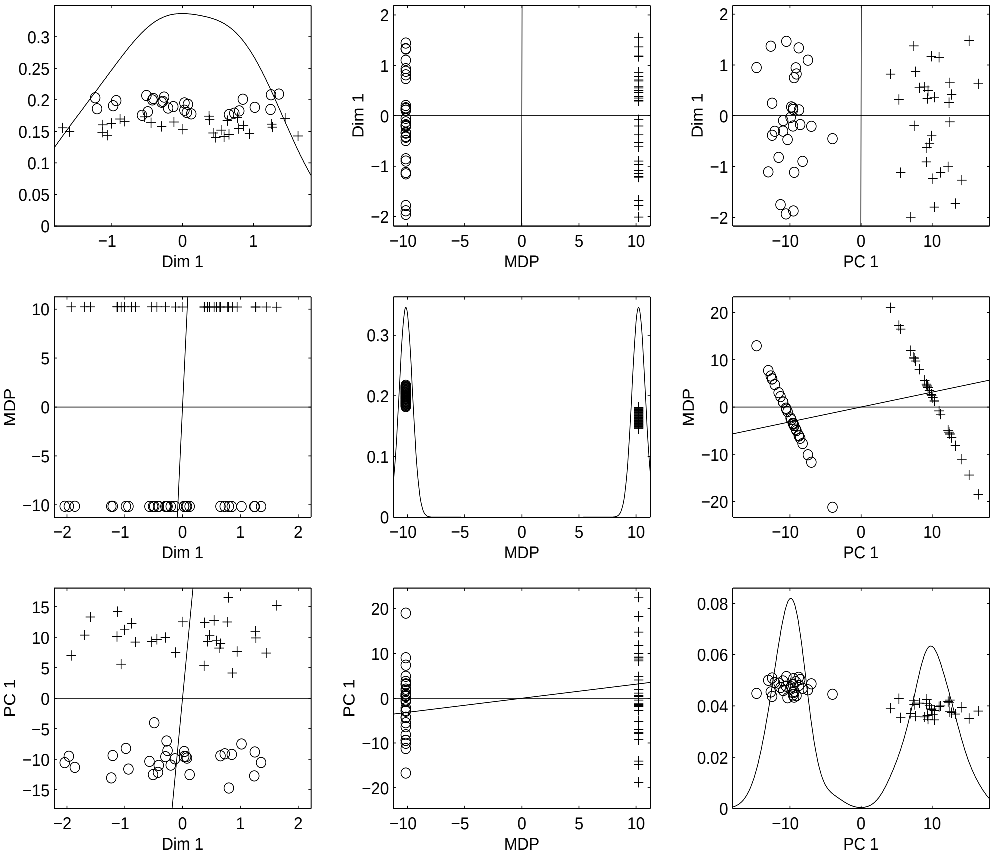

Figure 3.2: *Visualization of a toy HDLSS data based on 1 and 2 dimensional projections ($d = 4000$, $n = 60$). The three projection directions, used in the diagonal panels, are the first coordinate direction, the MDP direction, and the first principal component direction. The off-diagonal panels show the projected data on the plane generated by the respective two directions.*

plane of MDP and PC1 ([2,3] and [3,2]) show that these two directions are much closer to each other, which means the geometric representation of HDLSS data in Chapter 2 applies well for this toy data set.

3.3 Mathematics of the MDP Direction

In this section we express the MDP direction vector in a closed form and show that it is uniquely determined within the subspace generated by the data. Also, we show that it is equivalent to the FLD direction vector when the dimension d is less than or equal to the sample size minus two, i.e., when $d \leqslant n-2$.

Let $\mathbf{X} = (\mathbf{x}_1, \cdots, \mathbf{x}_{n_1})$ be the matrix of the training data from Class $+1$, where the $\mathbf{x}_i$'s are random column vectors from a continuous probability distribution in $\mathcal{R}^d$, and define $\mathbf{Y} = (\mathbf{y}_1, \cdots, \mathbf{y}_{n_2})$ for Class -1 in a similar way. Let $n = n_1 + n_2$. Define the $d \times n$ combined data matrix $\mathbf{Z}$ as $[\mathbf{X}, \mathbf{Y}]$, the horizontal concatenation of the matrices $\mathbf{X}$ and $\mathbf{Y}$, using Matlab-like notation. Denote the global sample mean vector of the combined data by $\bar{\mathbf{z}}$. Then the sample covariance matrix of $\mathbf{Z}$ is

$$\widetilde{\mathbf{\Sigma}} \equiv \frac{1}{n-1}\{(\mathbf{Z} - \bar{\mathbf{Z}})(\mathbf{Z} - \bar{\mathbf{Z}})^{\mathrm{T}}\}, \tag{3.14}$$

where $\bar{\mathbf{Z}} = \bar{\mathbf{z}}\mathbf{1}_n^{\mathrm{T}}$. We will call the matrix $\widetilde{\mathbf{\Sigma}}$ the global sample covariance matrix.

The MDP direction vector is defined as

$$\mathbf{v}_{\mathrm{MDP}} \equiv \frac{\tilde{\mathbf{\Sigma}}^{-}(\bar{\mathbf{x}} - \bar{\mathbf{y}})}{\|\tilde{\mathbf{\Sigma}}^{-}(\bar{\mathbf{x}} - \bar{\mathbf{y}})\|}, \tag{3.15}$$

where $\mathbf{A}^{-}$ is the Moore-Penrose generalized inverse of a matrix $\mathbf{A}$. The MDP direction vector exists if the data vectors are linearly independent, which is satisfied with probability one when the underlying distribution of the data is continuous in $\mathcal{R}^d$.

3.3.1 Characterization of the MDP Direction Vector in the Data Space

In this section we characterize the geometry of the MDP direction vector within the data space. The data vectors in the matrix $\mathbf{Z}$, i.e. the columns of $\mathbf{Z}$, generate an n-dimensional subspace in $\mathcal{R}^d$ and this subspace is expressed as

$$\mathcal{S}_{\mathbf{Z}} = \{\mathbf{Z}\mathbf{w} : \mathbf{w} \in \mathcal{R}^n\}. \tag{3.16}$$

In other words, $\mathcal{S}_{\mathbf{Z}}$ is the set of all linear combinations of the data vectors. If we let $\widetilde{\mathcal{H}}_{\mathbf{Z}}$ be the hyperplane generated by $\mathbf{Z}$, then we can write

$$\widetilde{\mathcal{H}}_{\mathbf{Z}} = \{\mathbf{Z}\mathbf{w} : \mathbf{w}^{\mathrm{T}}\mathbf{1}_n = 1, \mathbf{w} \in \mathcal{R}^n\}. \tag{3.17}$$

Note that $\widetilde{\mathcal{H}}_{\mathbf{Z}}$ is a set of linear combinations of data points of which the sum of the coefficients is one. The parallel subspace can be found by shifting the hyperplane so that it goes through the origin. A natural shift is via the point in $\widetilde{\mathcal{H}}_{\mathbf{Z}}$ that is closest to the origin which is calculated in the following lemma.

Lemma 3.3.1. *Let $\mathbf{v}_{\mathbf{Z}}$ be the point in $\widetilde{\mathcal{H}}_{\mathbf{Z}}$ that is nearest to the origin. Then,*

$$\mathbf{v}_{\mathbb{Z}} = \frac{\mathbf{Z}(\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{1}_n}{\mathbf{1}_n^T(\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{1}_n}.$$

Proof. Since $\mathbf{v}_{\mathbf{Z}}$ is on the hyperplane $\widetilde{\mathcal{H}}_{\mathbf{Z}}$, it can be expressed in the form $\mathbf{v}_{\mathbf{Z}} = \mathbf{Z}\mathbf{w}$, where $\mathbf{w}^{\mathrm{T}}\mathbf{1}_n = 1, \mathbf{w} \in \mathcal{R}^n$ by (3.17). The squared distance from the origin to $\mathbf{v}_{\mathbf{Z}}$ is $\mathbf{w}^{\mathrm{T}}\mathbf{Z}^{\mathrm{T}}\mathbf{Z}\mathbf{w}$ and we need to find $\mathbf{w}$ that minimizes this distance. The Lagrangian (Cristianini and Shawe-Taylor, 2000, Chapter 5) of this minimization problem is

$$L(\mathbf{w}) = \frac{1}{2}\mathbf{w}^{\mathrm{T}}\mathbf{Z}^{\mathrm{T}}\mathbf{Z}\mathbf{w} - \alpha(\mathbf{1}_n^{\mathrm{T}}\mathbf{w} - 1),$$

where $\alpha \geqslant 0$ is the Lagrangian multiplier. From $\partial L(\mathbf{w})/\partial \mathbf{w} = 0$,

$$\mathbf{w} \;\; = \;\; \alpha(\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}.$$

From $\mathbf{w}^{\mathrm{T}}\mathbf{1}_n = 1$,

$$\alpha \;=\; \frac{1}{\mathbf{1}_n^{\mathrm{T}}(\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}\mathbf{1}_n}.$$

Thus,

$$\mathbf{v}_{\mathbf{Z}} = \frac{\mathbf{Z}(\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}\mathbf{1}_n}{\mathbf{1}_n^{\mathrm{T}}(\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}\mathbf{1}_n}.$$

□

Now let us shift the hyperplane $\widetilde{\mathcal{H}}_{\mathbf{Z}}$ so that it contains the origin and call the new shifted hyperplane $\mathcal{H}_{\mathbf{Z}}$. Because $\mathcal{H}_{\mathbf{Z}} = \widetilde{\mathcal{H}}_{\mathbf{Z}} - \mathbf{v}_{\mathbf{Z}}$,

$$\begin{aligned} \mathcal{H}_{\mathbf{Z}} &= \left\{ \mathbf{Z}\mathbf{w} : \mathbf{w}^* = \mathbf{w} - \frac{(\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}\mathbf{1}_n}{\mathbf{1}_n^{\mathrm{T}}(\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}\mathbf{1}_n}, \mathbf{w}^{\mathrm{T}}\mathbf{1}_n = 1, \mathbf{w} \in \mathcal{R}^n \right\} \\ &= \left\{ \mathbf{Z}\mathbf{w}^* : \mathbf{w}^{*\mathrm{T}}\mathbf{1}_n = 0, \mathbf{w}^* \in \mathcal{R}^n \right\}. \end{aligned} \tag{3.18}$$

Note that $\mathcal{H}_{\mathbf{Z}}$ is a subspace of $\mathcal{R}^d$ with dimension $n-1$ and we can decompose $\mathcal{S}_{\mathbf{Z}}$ into an orthogonal sum of $\mathcal{H}_{\mathbf{Z}}$ and $\{\mathbf{v}_{\mathbf{Z}}\}$, i.e., $\mathcal{S}_{\mathbf{Z}} = \mathcal{H}_{\mathbf{Z}} \oplus \{\mathbf{v}_{\mathbb{Z}}\}$. In the same fashion we can define subspaces parallel to the hyperplanes of $\mathbf{X}$ and $\mathbf{Y}$, call them $\mathcal{H}_{\mathbf{X}}$ and $\mathcal{H}_{\mathbf{Y}}$, respectively. They have the following expressions:

$$\begin{aligned} \mathcal{H}_{\mathbf{X}} &= \left\{ \mathbf{X}\mathbf{w}_1^* : \mathbf{w}_1^* = \mathbf{w}_1 - \frac{(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{1}_{n_1}}{\mathbf{1}_{n_1}^{\mathrm{T}}(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{1}_{n_1}}, \mathbf{w}_1^{\mathrm{T}}\mathbf{1}_{n_1} = 1, \mathbf{w}_1 \in \mathcal{R}^{n_1} \right\} \\ &= \left\{ \mathbf{X}\mathbf{w}_1^* : \mathbf{w}_1^{*\mathrm{T}}\mathbf{1}_{n_1} = 0, \mathbf{w}_1^* \in \mathcal{R}^{n_1} \right\}, \end{aligned} \tag{3.19}$$

$$\begin{aligned} \mathcal{H}_{\mathbf{Y}} &= \left\{ \mathbf{Y}\mathbf{w}_2^* : \mathbf{w}_2^* = \mathbf{w}_2 - \frac{(\mathbf{Y}^{\mathrm{T}}\mathbf{Y})^{-1}\mathbf{1}_{n_2}}{\mathbf{1}_{n_2}^{\mathrm{T}}(\mathbf{Y}^{\mathrm{T}}\mathbf{Y})^{-1}\mathbf{1}_{n_2}}, \mathbf{w}_2^{\mathrm{T}}\mathbf{1}_{n_2} = 1, \mathbf{w}_2 \in \mathcal{R}^{n_2} \right\} \\ &= \left\{ \mathbf{Y}\mathbf{w}_2^* : \mathbf{w}_2^{*\mathrm{T}}\mathbf{1}_{n_2} = 0, \mathbf{w}_2^* \in \mathcal{R}^{n_2} \right\}. \end{aligned} \tag{3.20}$$

We can show these two subspaces of $\mathbf{X}$ and $\mathbf{Y}$ are actually the subspace of $\mathcal{H}_{\mathbf{Z}}$ in the following lemma.

Lemma 3.3.2. *Let $\mathcal{H}_{\mathbf{Z}}$, $\mathcal{H}_{\mathbf{X}}$, and $\mathcal{H}_{\mathbf{Y}}$ be as defined in (3.18), (3.19), and (3.20), respectively. Then both $\mathcal{H}_{\mathbf{X}}$ and $\mathcal{H}_{\mathbf{Y}}$ are subspaces of $\mathcal{H}_{\mathbf{Z}}$.*

Proof. This can be shown by setting the $\mathbf{w}^*$ in (3.18) to $\mathbf{w}^* = [\mathbf{w}_1^*; \mathbf{0}_{n_2}]$ for $\mathcal{H}_{\mathbf{X}}$ and $\mathbf{w}^* = [\mathbf{0}_{n_1}; \mathbf{w}_2^*]$ for $\mathcal{H}_{\mathbf{Y}}$. Here ";" denotes the vertical concatenation of two vectors and $\mathbf{0}_n$ is the n-vector of zeros. □

The following theorem specifies where the MDP direction vector $\mathbf{v}_{\text{MDP}}$ (as defined in (3.15)) lies within the subspace generated by the data.

Theorem 3.3.3. *The MDP direction vector $\mathbf{v}_{MDP}$ is a member of $\mathcal{H}_{\mathbf{Z}}$ and orthogonal to the subspaces $\mathcal{H}_{\mathbf{X}}$ and $\mathcal{H}_{\mathbf{Y}}$. i.e.,*

$$\mathcal{H}_{\mathbf{Z}} = \{\mathcal{H}_{\mathbf{X}} + \mathcal{H}_{\mathbf{Y}}\} \oplus \{\mathbf{v}_{MDP}\}. \tag{3.21}$$

Note that this theorem also implies that the MDP direction vector is uniquely determined within the subspace generated by the data: The sum of dimensions in the right hand side of (3.21) is equal to the dimension of $\mathcal{H}_{\mathbf{Z}}$, which is $n-1$.

Figure 3.3 illustrates geometric relationships among $\mathcal{H}_{\mathbf{X}}$, $\mathcal{H}_{\mathbf{Y}}$, and $\mathbf{v}_{\text{MDP}}$ when each class has two data points, i.e., $n_1 = n_2 = 2$. Note that the subspaces $\mathcal{H}_{\mathbf{X}}$ and $\mathcal{H}_{\mathbf{Y}}$ are actually one-dimensional straight lines and they are not necessarily orthogonal to each other. The hyperplanes $\widetilde{\mathcal{H}}_{\mathbf{X}}$ and $\widetilde{\mathcal{H}}_{\mathbf{Y}}$ are shifted to $\mathcal{H}_{\mathbf{X}}$ and $\mathcal{H}_{\mathbf{Y}}$ so they meet each other at the origin. Theorem 3.3.3 states that $\mathbf{v}_{\text{MDP}}$ is orthogonal to the subspace generated by $\mathcal{H}_{\mathbf{X}}$ and $\mathcal{H}_{\mathbf{Y}}$, shown as the gray plane in Figure 3.3.

To prove Theorem 3.3.3 we need the following lemma:

Lemma 3.3.4. *Let $\mathbf{A}$ be a $d \times m (m < d)$ matrix with rank$(\mathbf{A}) = m-1$ such that the sum of each row is zero. Then*

$$\mathbf{A}^{-}\mathbf{A} = \mathbf{I}_m - \frac{1}{m}\mathbf{J}_m,$$

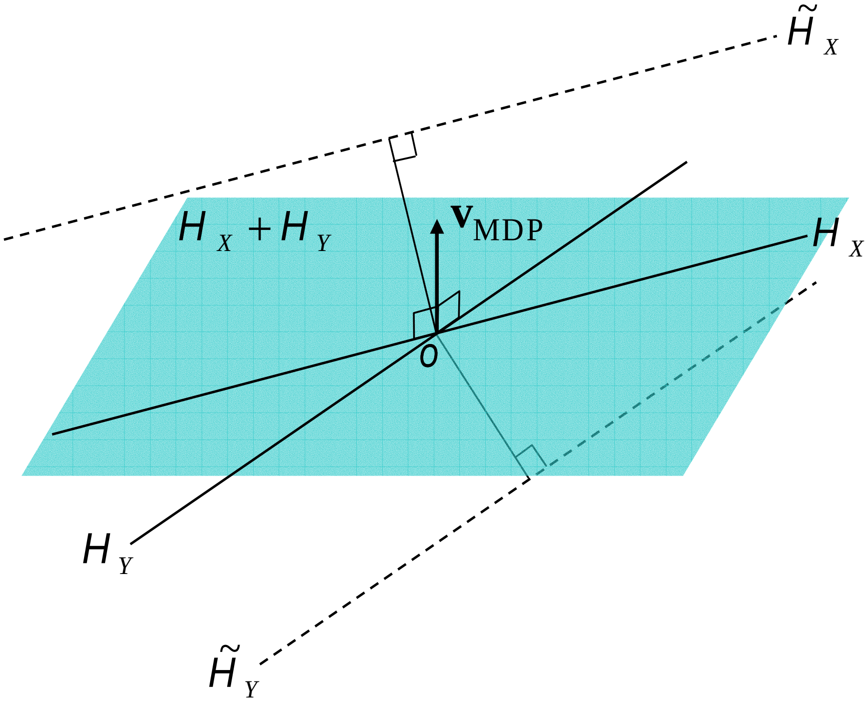

Figure 3.3: *The illustration of* $\mathcal{H}_{\mathbf{X}}$, $\mathcal{H}_{\mathbf{Y}}$, *and* $\mathbf{v}_{MDP}$ *when* $n_1 = 2$, *and* $n_2 = 2$.

where $\mathbf{J}_m = \mathbf{1}_m \mathbf{1}_m^T$.

Proof. Consider the singular value decomposition of

$$\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^{\mathrm{T}},$$

where

$$\begin{aligned}
\mathbf{U}_{(d\times m)} &= (\mathbf{u}_1, \cdots, \mathbf{u}_m), \\
\mathbf{\Lambda}_{(m\times m)} &= \mathrm{diag}(\lambda_1, \cdots, \lambda_{m-1}, 0), \quad \text{and} \\
\mathbf{V}_{(m\times m)} &= (\mathbf{v}_1, \cdots, \mathbf{v}_m).
\end{aligned}$$

Here $\mathbf{U}, \mathbf{\Lambda}$, and $\mathbf{V}$ are generic notations for singular value decomposition in this chapter. Note that the columns of $\mathbf{U}$ and $\mathbf{V}$ form an orthonormal basis in $\mathcal{R}^d$ and $\mathcal{R}^m$, respectively, and $\mathbf{V}$ is obtained from the eigenvalue decomposition of $\mathbf{A}^{\mathrm{T}}\mathbf{A} = \mathbf{V}\mathbf{L}^2\mathbf{V}^{\mathrm{T}}$. Since the columns of the eigenvector matrix $\mathbf{V}$ span the row space of $\mathbf{A}$, the sum of the coefficients of $\mathbf{v}_i$, $(i = 1, \cdots, m-1)$ is zero and the last column $\mathbf{v}_m = (m^{-\frac{1}{2}}, \cdots, m^{-\frac{1}{2}})^{\mathrm{T}}$ to satisfy orthogonality condition on the columns.

Now since

$$\mathbf{A}^{-} = (\mathbf{v}_1, \cdots, \mathbf{v}_{m-1}) \begin{pmatrix} \lambda_1^{-1} & \cdots & 0 \\ 0 & \ddots & 0 \\ 0 & \cdots & \lambda_{m-1}^{-1} \end{pmatrix} \begin{pmatrix} \mathbf{u}_1^{\mathrm{T}} \\ \vdots \\ \mathbf{u}_{m-1}^{\mathrm{T}} \end{pmatrix},$$

we have

$$\mathbf{A}^{-}\mathbf{A} = (\mathbf{v}_1, \cdots, \mathbf{v}_{m-1}) \begin{pmatrix} \mathbf{v}_1^{\mathrm{T}} \\ \vdots \\ \mathbf{v}_{m-1}^{\mathrm{T}} \end{pmatrix}$$

by the orthogonality of the matrix $\mathbf{U}$. It follows from the fact $\mathbf{V}\mathbf{V}^{\mathrm{T}} = \mathbf{I}_m$, that

$$\begin{aligned} \mathbf{A}^{-}\mathbf{A} &= \mathbf{v}_1\mathbf{v}_1^{\mathrm{T}} + \cdots + \mathbf{v}_{m-1}\mathbf{v}_{m-1}^{\mathrm{T}} \\ &= \mathbf{I}_m - \mathbf{v}_m\mathbf{v}_m^{\mathrm{T}} \\ &= \mathbf{I}_m - \frac{1}{m}\mathbf{J}_m. \end{aligned}$$

□

Proof of Theorem 3.3.3. First we show that $\mathbf{v}_{\mathrm{MDP}}$ is in $\mathcal{H}_{\mathbf{Z}}$. Note that each member of $\mathcal{H}_Z$ is a linear combination of data vectors where the sum of the coefficients is zero. Since

$$(\mathbf{Z} - \bar{\mathbf{Z}}) = \mathbf{Z}\left(\mathbf{I} - \frac{1}{n}\mathbf{J}_n\right),$$

and

$$(\bar{\mathbf{x}} - \bar{\mathbf{y}}) = \mathbf{Z}\left(\frac{1}{n_1}, \cdots, \frac{1}{n_1}, -\frac{1}{n_2}, \cdots, -\frac{1}{n_2}\right)^{\mathrm{T}},$$

all the columns of $\mathbf{Z} - \bar{\mathbf{Z}}$ and $\bar{\mathbf{x}} - \bar{\mathbf{y}}$ are in $\mathcal{H}_{\mathbf{Z}}$.

Using the notations $\mathbf{U}, \mathbf{\Lambda}$ and $\mathbf{V}$ in a generic way, the singular value decomposition of $\mathbf{Z}-\bar{\mathbf{Z}}$ is

$$\mathbf{Z}-\bar{\mathbf{Z}} = \mathbf{U}_{d\times(n-1)}\mathbf{\Lambda}_{(n-1)\times(n-1)}\mathbf{V}^{\mathrm{T}}_{(n-1)\times(n-1)}.$$

Then the eigenvalue decomposition of

$$\begin{aligned}(\mathbf{Z}-\bar{\mathbf{Z}})(\mathbf{Z}-\bar{\mathbf{Z}})^{\mathrm{T}} &\equiv \mathbf{U}\mathbf{\Lambda}\mathbf{V}^{\mathrm{T}}\mathbf{V}\mathbf{\Lambda}\mathbf{U}^{\mathrm{T}} \\ &= \mathbf{U}\mathbf{\Lambda}^2\mathbf{U}^{\mathrm{T}} \\ &= \sum_{j=1}^{n-1}\lambda_j^2\mathbf{u}_j\mathbf{u}_j^{\mathrm{T}},\end{aligned}$$

where $\mathbf{u}_1, \cdots, \mathbf{u}_{n-1}$ are columns of $\mathbf{U}$ and are orthogonal to each other. Since $\mathbf{u}_1, \cdots, \mathbf{u}_{n-1}$ are the bases of the column space of $\mathbf{Z}-\bar{\mathbf{Z}}$, which is $\mathcal{H}_{\mathbf{Z}}$, the mean difference vector $\bar{\mathbf{x}}-\bar{\mathbf{y}}$ can be expressed as the linear combination of $\mathbf{u}_j$'s. That is, for some constants $a_1, \cdots, a_{n-1}$,

$$\bar{\mathbf{x}}-\bar{\mathbf{y}} = \sum_{k=1}^{n-1} a_k\mathbf{u}_k.$$

Then, since the MDP direction is proportional to

$$\begin{aligned}\left\{(\mathbf{Z}-\bar{\mathbf{Z}})(\mathbf{Z}-\bar{\mathbf{Z}})^{\mathrm{T}}\right\}^{-}(\bar{\mathbf{x}}-\bar{\mathbf{y}}) &= \sum_{j=1}^{n-1}\lambda_j^{-2}\mathbf{u}_j\mathbf{u}_j^{\mathrm{T}}\left(\sum_{k=1}^{n-1}a_k\mathbf{u}_k\right) \\ &= \sum_{j=1}^{n-1}\lambda_j^{-2}a_j\mathbf{u}_j,\end{aligned}$$

the MDP belongs to $\mathcal{H}_{\mathbf{Z}}$.

For the second part of the theorem, let $\mathbf{w}$ be $(\mathbf{w}_1^*; \mathbf{w}_2^*)$, where $\mathbf{w}_1^*$ and $\mathbf{w}_2^*$ are as defined in (3.19) and (3.20), respectively. Let $\widetilde{\mathbf{Z}} = \mathbf{Z}-\bar{\mathbf{Z}}$. To show that $\{\mathbf{v}_{\mathrm{MDP}}\} \perp \mathcal{H}_{\mathbf{X}}$ and $\{\mathbf{v}_{\mathrm{MDP}}\} \perp \mathcal{H}_{\mathbf{Y}}$, it suffices to show that the inner product of $\mathbf{Z}\ \mathbf{w}$ and $\mathbf{v}_{\mathrm{MDP}}$ is zero.

Note that it suffices to show that the inner product of $\mathbf{Z}\mathbf{w}$ and $\mathbf{v}_{\mathrm{MDP}}$ is zero.

$$\begin{aligned}<\mathbf{Z}\mathbf{w}, \mathbf{v}_{\mathrm{MDP}}> &= \mathbf{w}^{\mathrm{T}}\mathbf{Z}^{\mathrm{T}}\mathbf{v}_{\mathrm{MDP}} \\ &\propto \mathbf{w}^{\mathrm{T}}\mathbf{Z}^{\mathrm{T}}(\widetilde{\mathbf{Z}}\widetilde{\mathbf{Z}}^{\mathrm{T}})^{-}(\bar{\mathbf{x}}-\bar{\mathbf{y}})\end{aligned}$$

$$\begin{aligned} &\propto \mathbf{w}^{\mathrm{T}}\mathbf{Z}^{\mathrm{T}}(\widetilde{\mathbf{Z}}\widetilde{\mathbf{Z}}^{\mathrm{T}})^{-}\mathbf{Z}(\mathbf{1}_{n_1}; -\mathbf{1}_{n_2}) \\ &= \mathbf{w}^{\mathrm{T}}\widetilde{\mathbf{Z}}^{\mathrm{T}}(\widetilde{\mathbf{Z}}\widetilde{\mathbf{Z}}^{\mathrm{T}})^{-}\widetilde{\mathbf{Z}}(\mathbf{1}_{n_1}; -\mathbf{1}_{n_2}). \end{aligned}$$

By Searle (1982, p.222),

$$\widetilde{\mathbf{Z}}^{\mathrm{T}}(\widetilde{\mathbf{Z}}\widetilde{\mathbf{Z}}^{\mathrm{T}})^{-}\widetilde{\mathbf{Z}} = \widetilde{\mathbf{Z}}^{-}\widetilde{\mathbf{Z}}.$$

Now by Lemma 3.3.4,

$$\begin{aligned} < \mathbf{Zw}, \mathbf{v}_{\mathrm{MDP}} > &= \mathbf{w}^{\mathrm{T}}\widetilde{\mathbf{Z}}^{-}\widetilde{\mathbf{Z}}(\mathbf{1}_{n_1}; -\mathbf{1}_{n_2}) \\ &= \mathbf{w}^{\mathrm{T}}(\mathbf{I}_n - \frac{1}{n}\mathbf{J}_n)(\mathbf{1}_{n_1}; -\mathbf{1}_{n_2}) \\ &= 0. \end{aligned}$$

□

3.3.2 Relationship to Fisher's Linear Discrimination

The FLD direction vector, defined in (3.3), has almost the same formula as MDP except it uses the pooled sample covariance matrix (3.4). The following theorem says the two directions are actually equivalent in non-HDLSS settings.

Theorem 3.3.5. *Let $\mathbf{v}_{MDP}$ and $\mathbf{v}_{FLD}$ be as defined in (3.15) and (3.3), respectively. If $d < n-1$ and the data matrix $\mathbf{Z}$ is of full rank, then $\mathbf{v}_{MDP} = \mathbf{v}_{FLD}$.*

To prove this theorem we need the following lemma, which is a slight variation of exercise 5.16 in Searle (1982).

Lemma 3.3.6. *Let $\mathbf{A}$ and $\mathbf{B}$ be matrices with the same number of rows and let c be a constant. As long as the following inverse matrices make sense,*

$$(\mathbf{B} + c\mathbf{A}\mathbf{A}^{T})^{-1}\mathbf{A} = \mathbf{B}^{-1}\mathbf{A}(\mathbf{I} + c\mathbf{A}^{T}\mathbf{B}^{-1}\mathbf{A})^{-1}.$$

Proof.

$$\begin{aligned}
(\mathbf{B}+c\mathbf{A}\mathbf{A}^{\mathrm{T}})\mathbf{B}^{-1}\mathbf{A}(\mathbf{I}+c\mathbf{A}^{\mathrm{T}}\mathbf{B}^{-1}\mathbf{A})^{-1} &= (\mathbf{I}+c\mathbf{A}\mathbf{A}^{\mathrm{T}}\mathbf{B}^{-1})\mathbf{A}(\mathbf{I}+c\mathbf{A}^{\mathrm{T}}\mathbf{B}^{-1}\mathbf{A})^{-1}\\
&= (\mathbf{A}+c\mathbf{A}\mathbf{A}^{\mathrm{T}}\mathbf{B}^{-1}\mathbf{A})(\mathbf{I}+c\mathbf{A}^{\mathrm{T}}\mathbf{B}^{-1}\mathbf{A})^{-1}\\
&= \mathbf{A}(\mathbf{I}+c\mathbf{A}^{\mathrm{T}}\mathbf{B}^{-1}\mathbf{A})(\mathbf{I}+c\mathbf{A}^{\mathrm{T}}\mathbf{B}^{-1}\mathbf{A})^{-1}\\
&= \mathbf{A}.
\end{aligned}$$

□

Proof of Theorem 3.3.5. Note that under the assumption of $d < n-1$, both the global sample covariance matrix $\widetilde{\mathbf{\Sigma}}$ and the pooled sample covariance matrix $\widehat{\mathbf{\Sigma}}$ are nonsingular so that the Moore-Penrose generalized inverse matrices are actually the regular inverse matrices. Let p and q be the proportions of samples with Class $+1$ and -1, respectively, i.e. $p = n_1/n$ and $q = n_2/n$. Note that the centered version of $\mathbf{Z}$,

$$\begin{aligned}
\mathbf{Z}-\bar{\mathbf{Z}} &= (\mathbf{X},\mathbf{Y})-(p\bar{\mathbf{X}}+q\bar{\mathbf{Y}})\\
&= (\mathbf{X},\mathbf{Y})-(\bar{\mathbf{X}},\bar{\mathbf{Y}})+\left(q(\bar{\mathbf{x}}-\bar{\mathbf{y}})\mathbf{1}_{n_1}^{\mathrm{T}},-p(\bar{\mathbf{x}}-\bar{\mathbf{y}})\mathbf{1}_{n_2}^{\mathrm{T}}\right)\\
&= (\mathbf{X}-\bar{\mathbf{X}},\mathbf{Y}-\bar{\mathbf{Y}})+\left(q(\bar{\mathbf{x}}-\bar{\mathbf{y}})\mathbf{1}_{n_1}^{\mathrm{T}},-p(\bar{\mathbf{x}}-\bar{\mathbf{y}})\mathbf{1}_{n_2}^{\mathrm{T}}\right).
\end{aligned}$$

Thus,

$$\begin{aligned}
(n-1)\widetilde{\mathbf{\Sigma}} &= (\mathbf{Z}-\bar{\mathbf{Z}})(\mathbf{Z}-\bar{\mathbf{Z}})^{\mathrm{T}}\\
&= (\mathbf{X}-\bar{\mathbf{X}})(\mathbf{X}-\bar{\mathbf{X}})^{\mathrm{T}}+(\mathbf{Y}-\bar{\mathbf{Y}})(\mathbf{Y}-\bar{\mathbf{Y}})^{\mathrm{T}}\\
&\qquad +n_1q^2(\bar{\mathbf{x}}-\bar{\mathbf{y}})(\bar{\mathbf{x}}-\bar{\mathbf{y}})^{\mathrm{T}}+n_2p^2(\bar{\mathbf{x}}-\bar{\mathbf{y}})(\bar{\mathbf{x}}-\bar{\mathbf{y}})^{\mathrm{T}}\\
&= (n-2)\widehat{\mathbf{\Sigma}}+(n_1q^2+n_2p^2)(\bar{\mathbf{x}}-\bar{\mathbf{y}})(\bar{\mathbf{x}}-\bar{\mathbf{y}})^{\mathrm{T}},
\end{aligned}$$

since the cross-product term is zero.

Therefore,

$$\widetilde{\mathbf{\Sigma}} = \frac{n-2}{n-1}\widehat{\mathbf{\Sigma}}+\frac{(n_1q^2+n_2p^2)(\bar{\mathbf{x}}-\bar{\mathbf{y}})(\bar{\mathbf{x}}-\bar{\mathbf{y}})^{\mathrm{T}}}{n-1}.$$

Now if we apply Lemma 3.3.6 with $\mathbf{B} = \frac{n-2}{n-1}\widehat{\mathbf{\Sigma}}$, $\mathbf{A} = \bar{\mathbf{x}} - \bar{\mathbf{y}}$, and $c = \frac{n_1 q^2 + n_2 p^2}{n-1}$, then

$$\begin{aligned}
\widetilde{\mathbf{\Sigma}}^{-1}(\bar{\mathbf{x}} - \bar{\mathbf{y}}) &= \left(\frac{n-2}{n-1}\widehat{\mathbf{\Sigma}}\right)^{-1}(\bar{\mathbf{x}} - \bar{\mathbf{y}})\left(\mathbf{I} + \frac{n_1 q^2 + n_2 p^2}{n-1}(\bar{\mathbf{x}} - \bar{\mathbf{y}})^{\mathrm{T}}\left(\frac{n-2}{n-1}\widehat{\mathbf{\Sigma}}\right)^{-1}(\bar{\mathbf{x}} - \bar{\mathbf{y}})\right)^{-1} \\
&= \frac{\widehat{\mathbf{\Sigma}}^{-1}(\bar{\mathbf{x}} - \bar{\mathbf{y}})}{\frac{n-2}{n-1} + \frac{n_1 q^2 + n_2 p^2}{n-1}(\bar{\mathbf{x}} - \bar{\mathbf{y}})^{\mathrm{T}}\widehat{\mathbf{\Sigma}}^{-1}(\bar{\mathbf{x}} - \bar{\mathbf{y}})} \\
&= \frac{\widehat{\mathbf{\Sigma}}^{-1}(\bar{\mathbf{x}} - \bar{\mathbf{y}})}{\text{constant}}.
\end{aligned}$$

Thus $\mathbf{v}_{\text{MDP}} = \mathbf{v}_{\text{FLD}}$ after standardization. □

Because $\mathbf{v}_{\text{MDP}}$ and $\mathbf{v}_{\text{FLD}}$ are based on different covariance matrices, Theorem 3.3.5 may be viewed as surprising. For deeper geometrical understanding, consider a simple example in $\mathcal{R}^2$ and see how it can be seen in terms of the underlying distributions. Let X and Y be random variables from two bivariate normal distributions with different means, but with the same covariance matrix. That is,

$$X \sim \mathcal{N}_2\left(\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right), \tag{3.22}$$

$$Y \sim \mathcal{N}_2\left(\begin{pmatrix} -\mu_1 \\ -\mu_2 \end{pmatrix}, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right), \tag{3.23}$$

where μ_1 and μ_2 are real numbers and $-1 < \rho < 1$. Note that by a shift of the data, this mean structure is quite general. The common underlying covariance structure of X and Y, whose estimator is the pooled sample covariance matrix $\widehat{\mathbf{\Sigma}}$, is

$$\mathbf{\Sigma}_p \equiv \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}.$$

Let Z be the random variable whose distribution is the mixture of the two distributions (3.22) and (3.23) with equal probabilities. We can write

$$\begin{aligned} Z &= \begin{pmatrix} Z_1 \\ Z_2 \end{pmatrix} \\ &= B \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} + U, \end{aligned}$$

where

$$B = \begin{cases} +1 & \text{w.p. } 1/2 \\ -1 & \text{w.p. } 1/2, \end{cases}$$

$$U = \begin{pmatrix} U_1 \\ U_2 \end{pmatrix} \sim \mathcal{N}_2 \left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix} \right),$$

and B and U are independent.

Then, Z has mean $\mathbf{0}$ and the variance of Z_i is equal to the variance of $\mu_i B + U_i$, which is $1 + \mu_i^2,\ i = 1, 2$. The covariance of Z_1 and Z_2 is equal to the covariance of $\mu_1 B + U_1$ and $\mu_2 B + U_2$, which is $\rho + \mu_1 \mu_2$. Thus, the covariance matrix of Z, whose estimator is the global sample covariance matrix $\widetilde{\mathbf{\Sigma}}$, is

$$\mathbf{\Sigma}_g \equiv \begin{pmatrix} 1 + \mu_1^2 & \rho + \mu_1 \mu_2 \\ \rho + \mu_1 \mu_2 & 1 + \mu_2^2 \end{pmatrix}.$$

Note that $\mathbf{\Sigma}_g$ can be expressed as a sum of $\mathbf{\Sigma}_p$ and the outer product of the mean difference vector $(2\mu_1, 2\mu_2)^{\mathrm{T}}$ multiplied by a constant:

$$\mathbf{\Sigma}_g = \mathbf{\Sigma}_p + \frac{1}{4} \begin{pmatrix} 2\mu_1 \\ 2\mu_2 \end{pmatrix} \begin{pmatrix} 2\mu_1 & 2\mu_2 \end{pmatrix}.$$

The true version of the MDP direction vector can be defined as the product of $\mathbf{\Sigma}_g^{-1}$ and the mean difference vector, $(2\mu_1, 2\mu_2)^{\mathrm{T}}$:

$$\begin{aligned}
\text{True } \mathbf{v}_{\mathrm{MDP}} &= \Sigma_g^{-1} \times \begin{pmatrix} 2\mu_1 \\ 2\mu_2 \end{pmatrix} \\
&\propto \begin{pmatrix} 1+\mu_2^2 & -\rho-\mu_1\mu_2 \\ -\rho-\mu_1\mu_2 & 1+\mu_1^2 \end{pmatrix} \times \begin{pmatrix} 2\mu_1 \\ 2\mu_2 \end{pmatrix} \qquad (3.24) \\
&\propto \begin{pmatrix} \mu_1-\rho\mu_2 \\ \mu_2-\rho\mu_1 \end{pmatrix}. \qquad (3.25)
\end{aligned}$$

Similarly the true version of the FLD direction vector is the product of $\mathbf{\Sigma}_p^{-1}$ and $(2\mu_1, 2\mu_2)^{\mathrm{T}}$:

$$\begin{aligned}
\text{True } \mathbf{v}_{\mathrm{FLD}} &= \mathbf{\Sigma}_p^{-1} \times \begin{pmatrix} 2\mu_1 \\ 2\mu_2 \end{pmatrix} \\
&\propto \begin{pmatrix} 1 & -\rho \\ -\rho & 1 \end{pmatrix} \times \begin{pmatrix} 2\mu_1 \\ 2\mu_2 \end{pmatrix} \\
&\propto \begin{pmatrix} \mu_1-\rho\mu_2 \\ \mu_2-\rho\mu_1 \end{pmatrix}. \qquad (3.26)
\end{aligned}$$

From (3.25) and (3.26), we can see the resulting direction vectors are actually equivalent.

One possible interpretation is that the effect of the mean difference vector on the global covariance matrix $\mathbf{\Sigma}_g$ is negated when we take the inverse (3.24), which cancels out when the mean difference vector is multiplied by $\mathbf{\Sigma}_g^{-1}$ to obtain the direction vector. Note that this is only true for the non-HDLSS case.

3.4 MDP as a Linear Classifier

3.4.1 Relationship to the Support Vector Machine

In many applications of SVM in HDLSS settings, we observe that a large portion of the data lie on the margin boundaries. Thus, if we project the data points onto the normal vector of the SVM hyperplane, many of the projections are identical, which is what we call *data piling*. Marron *et al.* (2005) initially developed the notion of data piling. The MDP can be thought as a linear classifier that maximizes data piling, in the sense that all of the data vectors pile onto exactly two distinct values.

The term "Maximal" has two implications: The MDP maximizes the amount of data piling, forcing all data points to pile up at the margin boundaries. Also it has the largest distance between the two piling sites among all the possible piling directions. Note that any linear combination of the vector orthogonal to the affine set of the data and the MDP direction vector also has complete data piling, but has a smaller margin than MDP.

Here we use a toy example to illustrate data piling for the SVM and the MDP. The toy data vectors are generated from a spherical, unit variance Gaussian distribution with dimension 50, and mean 0, except that the first coordinate has mean $+5.2$ for Class $+1$ and -5.2 for Class -1. The toy data set is of size 40, out of which 20 belong to each class. Note that since the underlying distribution is known, the theoretically optimal Bayes rule is the hyperplane with $(1, 0, \cdots, 0)^{\mathrm{T}}$ as its normal direction vector.

Figure 3.4 has the same structure as Figure 3.2 except the three projection directions are now the Bayes rule, MDP, and the SVM directions. The projections onto the theoretical Bayes direction have a nice Gaussian mixture appearance in [1,1]. The SVM shows some partial data piling in [3,3], i.e., for each class, the projection values closest to 0 are taken on by about 10 data points. The MDP, as expected, has complete data piling in [2,2]. From [2,2] and [3,3], it can be seen that the MDP has smaller margin (the distance between the piling sites) than SVM. The MDP direction tends to have a smaller margin than the SVM because it needs to adapt to all the noise in the data to obtain the complete data piling. Note that as discussed above, the MDP maximizes the margin keeping the complete data piling, however, the SVM maximizes it regardless of data piling. A smaller margin usually yields a bigger classification error bound as shown in Vapnik (1998, Chapter 10). Thus for this example, we can expect that the MDP has worse generalizability than the SVM.

The angles between the lines in the off-diagonal panels [2,1] and [3,1] also give some insights into generalizability. The Bayes direction has a bigger angle with MDP (60.3°) than with SVM (22.7°). Since the data are separated only in the Bayes direction, the angle between the Bayes direction and the MDP ([2,1]) or SVM ([3,1]) determines the discrimination performance. In particular, performance is better for smaller angles, which again implies the superior

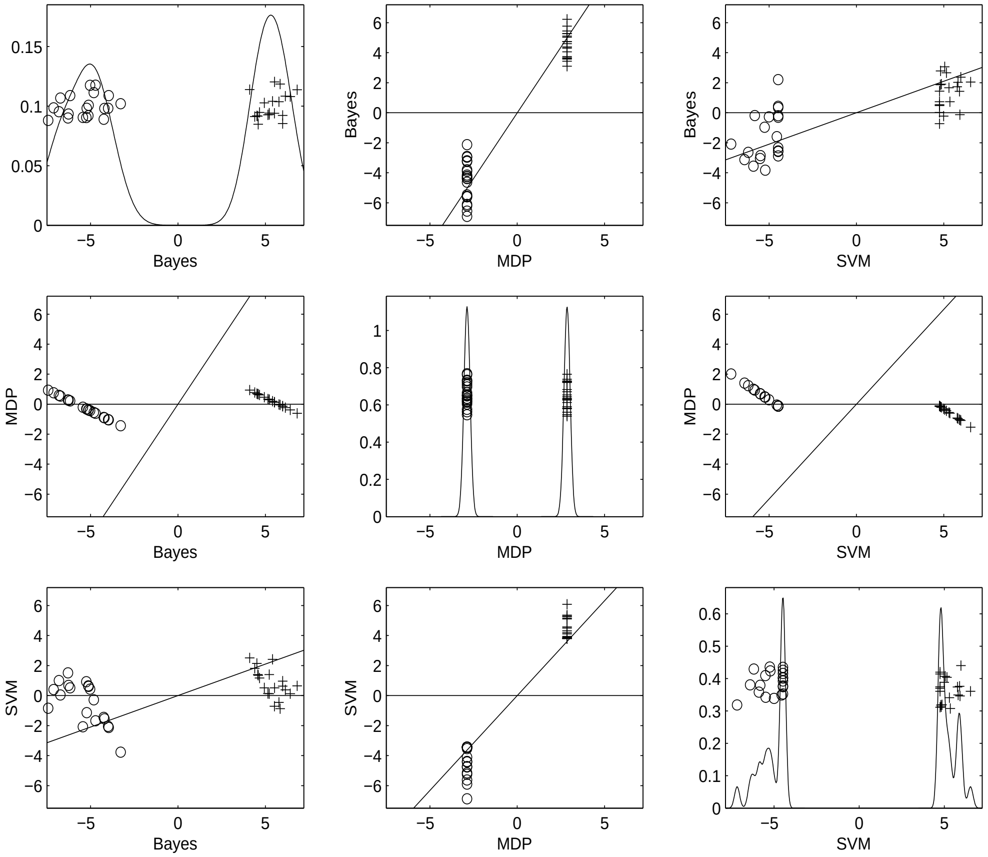

Figure 3.4: *Illustration of data piling for the SVM and the MDP direction using a toy data example ($d = 50$, $n = 40$). Here the three projection directions are the theoretically optimal Bayes rule, the MDP, and the SVM direction. Shows none, complete, and moderate data piling respectively.*

performance of the SVM in this example.

3.4.2 Gene Expression Data Example

In this section a real data example is presented to demonstrate data piling. The data are microarray gene expressions from the UNC breast cancer data base. See Perou *et al.* (2000) for the details regarding this data set. As for many other microarray data sets, this is a HDLSS data set, with 5,705 genes and 105 breast cancer patients, of whom 71 survived and 34 died. Here we consider a linear classification problem using mortality as the target variable and apply the SVM, DWD, MDP, and MD. The tuning parameter $C = 1000$ is used for the SVM and DWD. The projected value of the global sample mean is used as the threshold for the MDP.

Figure 3.5 shows a draftsman's view of the projection plots of the MDP, SVM, DWD, and MD direction vectors. In each panel, the circles represent the women that survived and the pluses are for the group that died. The panel [1,1] shows that the projections of each class onto the MDP direction vector pile up completely at two points, one for each class, and the distance between them is 9.00. [2,2] shows substantial data piling also for the SVM direction vector. The projections of the group that died pile up completely near 6 and most of the group that survived pile up around -3, with the distance between the piling sites about 9.09. The projection onto the DWD direction vector which is shown in [3,3], on the other hand, shows no piling at all, and the distance between the two peaks is a little more than 10. The MD has the biggest distance (about 21) between the peaks of the projections in [4,4], however, it also has huge overlap.

The 2-d projections on the off-diagonal panels highlight relationships between these directions. In particular, the MDP and SVM directions are quite similar to each other in this example (e.g. the angle between them is small in [1,2]). The only difference between substantial data piling (SVM) and complete data piling (MDP) is a slight rotation. The DWD direction is rather close to both MDP and SVM, having relatively small angles with them in [1,3] and [2,3]. The fact that the MD direction is very different than the others is reflected in the much larger relative angles.

The ten-fold cross validation error of the four methods are 0.50 (MDP), 0.28 (SVM), 0.29 (DWD), and 0.36 (MD). For DWD and SVM, the penalty parameter C is tuned by cross

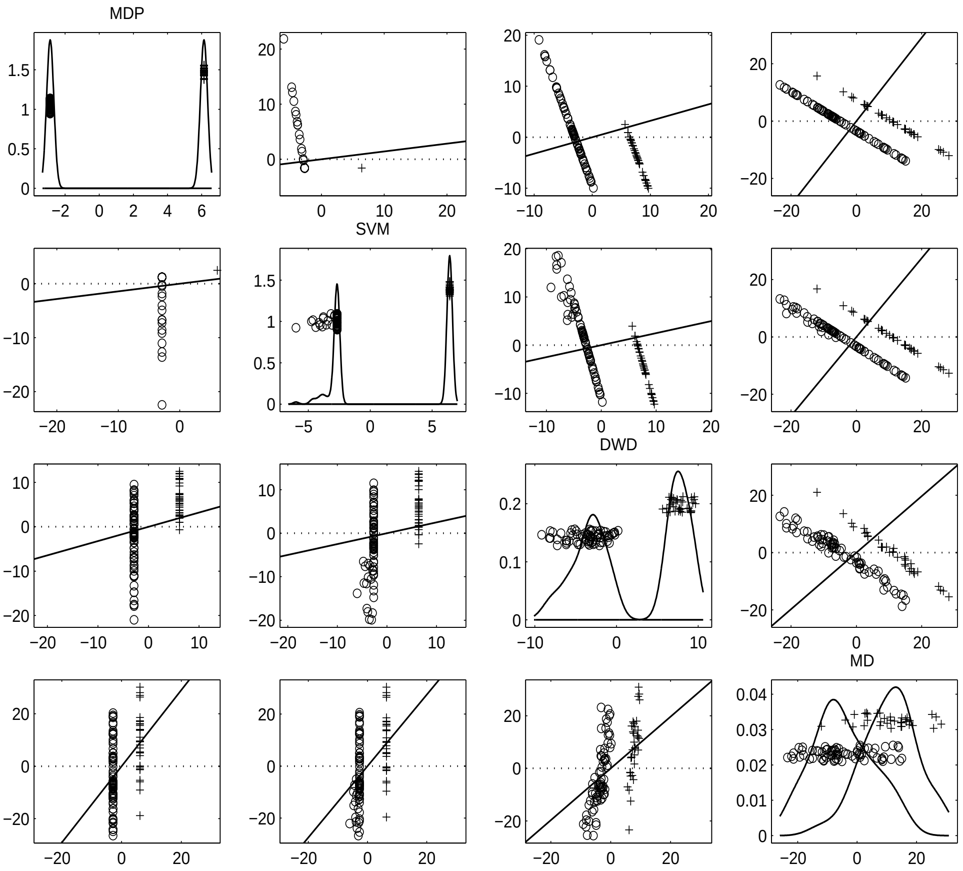

Figure 3.5: *Projections of microarray data onto MDP, SVM, DWD, and MD direction vectors, with 1-d projections onto each direction vectors on the diagonal panels and 2-d projections onto the plane generated by each pair of the direction vectors on the off-diagonal panels*

validation within each training data set. Consistent with the above discussions, MDP has a substantially worse error rate. MD also shows a poor performance. The SVM shows slightly better performance than DWD. Hall *et al.* (2005) pointed out a need for improvement of DWD with unequal sample sizes of each class, which may be the reason for the inferior performance of DWD to SVM. Since these data have a large difference between the sample sizes for each class, an improved version of DWD that takes care of this problem is recommended, as proposed in Zhang *et al.* (2005b).

For this particular data set, the accuracy of the labels is also an issue. There are likely to be some mislabeled samples in the sense that the patients who are about to die were categorized as "survived" when their gene expression may be more closely connected with the patients who died. A sophisticated approach to this issue, using survival analysis ideas, is taken in Johnson *et al.* (2005).

3.4.3 A HDLSS Comparative Simulation Study

In this section, the MDP is compared with other simple classification methods over a wide range of dimensionality. Each class has a d-dimensional multivariate Gaussian distribution with identity covariance. The two classes only differ in their means: Class $+1$ has mean $\boldsymbol{\mu} = (0.1, \cdots, 0.1)^{\mathrm{T}}$ and Class -1 has mean $-\boldsymbol{\mu} = (-0.1, \cdots, -0.1)^{\mathrm{T}}$. We generate $n = 50$ training data vectors, 25 for each class, with $d = 5, 10, 50, 100$, and 1000, repeating 100 times at each setting. Later, for a more detailed analysis of the behavior in the critical region where $d \approx n$, 1000 repetitions were made at the finer grid $d = 41, 42, \cdots, 60$. The classification methods considered here are MD, FLD, MDP, NB, as well as the theoretically optimal Bayes rule, whose direction vector is $(1, \cdots, 1)^{\mathrm{T}}/\sqrt{d}$.

To evaluate the performances of the discrimination methods, an analytic misclassification error is calculated as follows. Suppose $\mathbf{v}$ is the direction vector of a separating hyperplane. The projection of a random vector from Class $+1(-1)$ onto $\mathbf{v}$ is distributed as $\mathcal{N}(\mathbf{v}^{\mathrm{T}}\boldsymbol{\mu}(-\boldsymbol{\mu}), 1)$. To simplify the argument, we use 0 as the threshold of the classifiers. In other words, a new sample $\mathbf{x}$ is classified as Class $+1(-1)$ if the projected value $\mathbf{v}^{\mathrm{T}}\mathbf{x} > 0$ (< 0). Instead of generating independent test data to evaluate the misclassification error, we calculate the theoretical error analytically since we know the underlying distributions. It is given by $\Phi(-\mathbf{v}^{\mathrm{T}}\boldsymbol{\mu})$, where Φ is

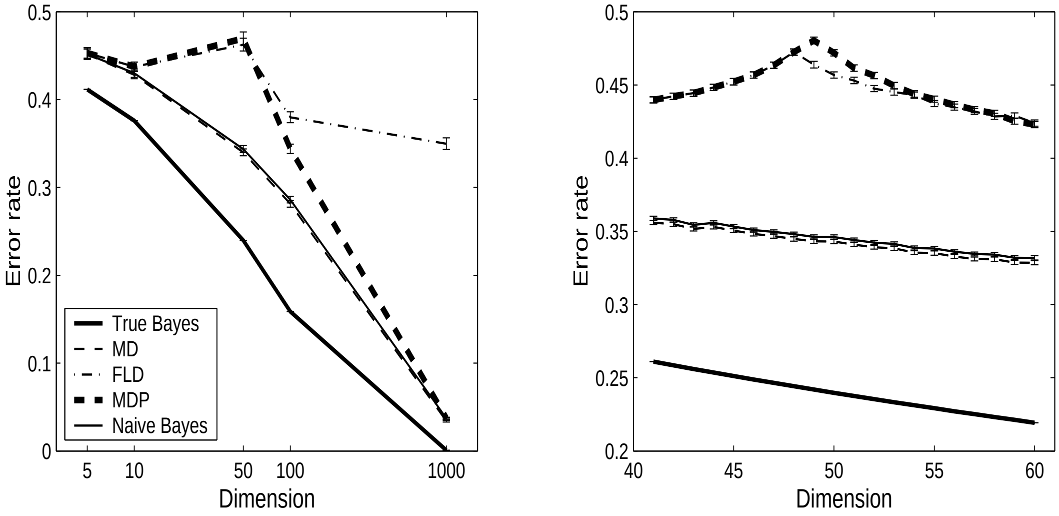

Figure 3.6: *Misclassification error rate shown with error bars for the Bayes, MD, FLD, MDP, and Naive Bayes method from the simulation with* $n = 50$*. Left hand panels are for* $d = (5, 10, 50, 100, 1000)$ *with 100 repetitions and right hand panels are for* $d = (41, \cdots, 60)$ *with 1000 repetitions.*

the standard normal cumulative distribution function.

Figure 3.6 shows the analytic misclassification rates with error bars, reflecting the Monte Carlo variation, for each method over a range of dimensions. The right hand panel shows the results for a finer range of d around n. We can see that MD and NB show nearly identical error rates in all cases. Also they both perform better than any other methods in most of the cases except the theoretically optimal Bayes rule . (The MDP behaves slightly better than NB when $d = 1000$.) Note that for this underlying structure of the data, MD is the empirically optimal Bayes rule, thus it is not surprising for MD to perform the best among all methods. The good HDLSS performance of NB was theoretically justified by Bickel and Levina (2004), where they compare the asymptotic properties of FLD and NB as both d and n tend to infinity. In their work, the inferior performance of FLD in a very high dimensional space is also discussed.

Also in this particular setting, the classification task becomes somewhat less challenging as the dimension increases because a larger d means the two classes are farther away from each other. This is part of the reason that all the methods show generally better performances for higher dimensions in the left hand panel of Figure 3.6.

The good performance of MDP when $d = 1000$ can be explained by the asymptotic geometric representation of HDLSS data, as in Section 2.5. In a binary classification setting, the training data vectors become two simplices, one for each class, which makes MDP the heuristically desired classifier. Also any reasonable classification method will find the same direction as the MDP eventually when d becomes very large.

The FLD and MDP have exactly the same error rates up to $d = 48 = n - 2$ in the right hand panel, as expected from Theorem 3.3.5. An interesting feature is that, however, when d becomes close to n, the error rates of both FLD and MDP get worse, with peaks at $d = n-2$ for FLD and $d = n - 1$ for MDP as shown in the right hand panel of Figure 3.6. After that, their error rates start decreasing, with MDP recovering faster than FLD. This phenomenon can be explained by the following. As d grows close to n, the estimation of the covariance structure becomes unreliable due to the lack of data points, which yields increasing error rates. The effect of this unreliable covariance estimation problem peaks at around $d = n$, (more precisely at the ranks of the estimated covariance matrices, $n - 2$ for FLD and $n - 1$ for MDP) and remains until d is somewhat higher than n. However, as d grows well past n, the HDLSS asymptotics begin to take effect as discussed earlier, with the increasing distance between classes for this particular setting, resulting in decreasing error rates.

CHAPTER 4

Bandwidth Selection for Kernel Based Classification

4.1 Introduction to the Kernel Based Classification

In discrimination (Section 3.1), sometimes a linear classifier between two classes is not flexible enough for the data with complicated structure. Aizerman *et al.* (1964) introduced the idea of data embedding, which maps the data in the original space into a higher dimensional space via an embedding function ϕ.

Figure 4.1 illustrates data embedding with a toy example. The original data, shown in the left, are not linearly separable in the original space $\mathcal{R}^1$. However, after mapping by the function $\phi : x \rightarrow (x, x^2)^\mathrm{T}$, the embedded data are linearly separable in $\mathcal{R}^2$.

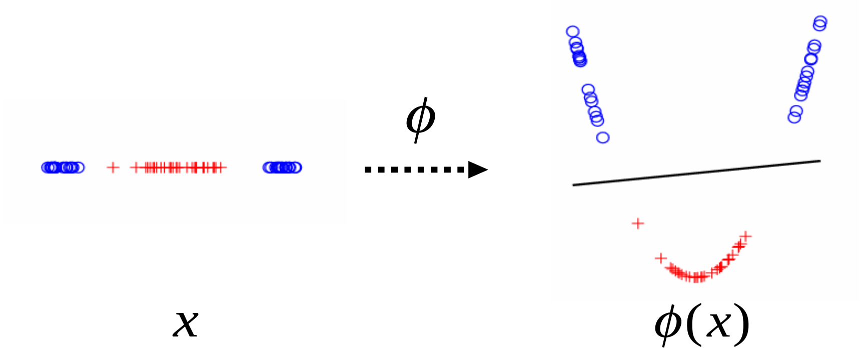

Figure 4.1: *An illustration of data embedding. The data in $\mathcal{R}^1$ (left) becomes linearly separable in the two dimensional space (right). The DWD (Section 3.1.4) separating hyperplane is also shown in the right figure.*

The higher dimensional embedded space is usually called the feature space, denoted by $\mathcal{F}$. We already introduced the term in Section 3.1 as the space where the data points are. We can avoid possible confusion by using the term "feature space" for only the space where the actual classification occurs.

If we map the data in $\mathcal{R}^d$ into a feature space $\mathcal{F}$ using a nonlinear function ϕ, a linear classifier in the feature space $\mathcal{F}$ has the form

$$f(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}) + b,$$

where $\mathbf{w}$ is the normal vector of the separating hyperplane in the feature space and b is the threshold. Note that this classifier creates a nonlinear classification boundary in the original space $\mathcal{R}^d$.

If ϕ has an explicit form, as it does in the toy example above, the data embedding is called *explicit*. For example, the polynomial embedding is one of the simplest examples of explicit embedding. Let the original data lie in $\mathcal{R}^d$, i.e., $\mathbf{x} = (x_1, x_2, \cdots, x_d)^{\mathrm{T}} \in \mathcal{R}^d$. If we define ϕ as

$$\phi(\mathbf{x}) = (x_1, \cdots, x_d, x_1^2, \cdots, x_d^2, x_1x_2, \cdots, x_{d-1}x_d)^{\mathrm{T}},$$

then a linear classifier in the feature space of dimension $d(d+3)/2$, is a quadratic function of $\mathbf{x}$, which yields a nonlinear classifying boundary in the original data space.

Algorithms of some linear classification methods only depend on inner products between the data vectors, i.e., we can obtain a classifier without knowing all the coordinates of the data vectors as long as we have their pairwise inner products. For these classification methods, replacing the inner product by some nonlinear function yields more general types of classifiers. In other words, we can have a nonlinear classifier by switching $\langle \mathbf{x}_i, \mathbf{x}_j \rangle$ to a kernel function $K(\mathbf{x}_i, \mathbf{x}_j)$, $i, j = 1, \cdots, n$. This kernel function implicitly defines the feature space which we do not need to construct explicitly. Examples of these types of methods include the mean difference (MD) method (Section 3.1.1) and the support vector machine (SVM) (Section 3.1.3), whose nonlinear versions will be discussed in Section 4.1.2.

The idea of data embedding is not restricted only to discrimination. It can also be applied to other problems such as regression and clustering, etc. The extension of any linear method to a nonlinear one can be easily done by mapping the data via the embedding function ϕ (the explicit embedding). The implicit embedding can also be applied to any linear method, if that method only needs pairwise inner products of data. The explicit/implicit data embedding method is called the *kernel method*. Note that a nonlinear data analysis procedure using the kernel method is a linear task in a kernel embedded feature space. Sometimes the term "kernel trick" is reserved only for the implicit embedding method due to the fact that it only needs the kernel evaluations of data, not the whole coordinate information.

Nonlinear classification by data embedding is called kernel based classification, which we will focus on in this chapter. In the following subsections, some properties of a kernel function will be discussed briefly (Section 4.1.1) and some examples of kernel based classification will be introduced (Section 4.1.2). In Section 4.1.3, the Gaussian kernel and its feature space are discussed and the rest of this chapter is outlined.

4.1.1 Kernel Function

This section briefly discusses some theoretical properties of kernel methods that are well developed over the years. Cristianini and Shawe-Taylor (2000), Schölkopf and Smola (2001), Shawe-Taylor and Cristianini (2004) and references therewithin are good sources for the deeper understanding of the subject. Suppose $\mathcal{X}$ is the space of input variables, and note that it may not be a Euclidean space. The kernel function K, defined on the Cartesian product space of $\mathcal{X}$

$$K : \mathcal{X} \times \mathcal{X} \longmapsto \mathcal{R},$$

is considered as a similarity measure between inputs. Thus the choice of a kernel function must take the structure of the data and our knowledge of the particular application into account. Also the kernel function K is considered as the inner product of the embedding function ϕ, i.e.,

$$K(\mathbf{x}, \mathbf{y}) = \langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle,$$

in other words,

$$\phi(\mathbf{x}) = K(\mathbf{x}, \cdot).$$

Here $\cdot$ indicates the position of the argument of the function.

The *kernel matrix* $\mathbf{K}$ is defined as the $n \times n$ matrix whose entries are

$$\mathbf{K}_{ij} = K(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle.$$

In particular, the kernel matrix for the linear kernel, which is just the usual inner product $\langle \cdot, \cdot \rangle$, is called the *Gram matrix.* Note that all the necessary information from the input data is in $\mathbf{K}$. Also note that a kernel matrix is positive semi-definite, since for any vector $\mathbf{v} = (v_1, \cdots, v_n)^{\mathrm{T}}$,

$$\begin{aligned} \mathbf{v}^{\mathrm{T}}\mathbf{K}\mathbf{v} &= \sum_{i,j=1}^{n} v_i v_j \mathbf{K}_{ij} = \sum_{i,j=1}^{n} v_i v_j \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle \\ &= \left\langle \sum_{i=1}^{n} v_i \phi(\mathbf{x}_i), \sum_{i=1}^{n} v_i \phi(\mathbf{x}_i) \right\rangle \\ &= \left\| \sum_{i=1}^{n} v_i \phi(\mathbf{x}_i) \right\|^2 \geqslant 0. \end{aligned}$$

For a symmetric function to be a kernel that constructs a feature space, it needs to satisfy Mercer's condition (Mercer, 1909): Suppose

$$\int_{\mathcal{X}\times\mathcal{X}} K(\mathbf{x}, \mathbf{y}) f(\mathbf{x}) f(\mathbf{y}) d\mathbf{x} d\mathbf{y} \geqslant 0,$$

for all $f \in L_2(\mathcal{X})$, where

$$L_2(\mathcal{X}) = \left\{ f : \int_{\mathcal{X}} f(\mathbf{x})^2 d\mathbf{x} < \infty \right\}.$$

Then $K(\mathbf{x}, \mathbf{y})$ can be expanded in a uniformly convergent series in terms of basis functions ϕ_j

$$K(\mathbf{x}, \mathbf{y}) = \sum_{j=1}^{\infty} \phi_j(\mathbf{x}) \phi_j(\mathbf{y}). \tag{4.1}$$

Furthermore, the series $\sum_{i=1}^{\infty} \|\phi_i\|^2_{L_2(\mathcal{X})}$ is convergent. Kernel embedded feature space $\mathcal{F}$ is a Hilbert space, which means a separable, complete inner product space. For this reason, sometimes the feature space associated with the kernel K, is denoted by $\mathcal{H}_K$.

It can be shown that the feature space is a set of functions of a form (Shawe-Taylor and

Cristianini, 2004, Chapter 2):

$$\mathcal{F} = \left\{ \sum_{i=1}^{\ell} \alpha_i K(\mathbf{x}_i, \cdot) \ : \ \ell \in \mathbb{N}, \mathbf{x}_i \in \mathcal{X}, \alpha_i \in \mathcal{R}, i = 1, \cdots, \ell \right\}.$$

Let $f, g \in \mathcal{F}$ be given by

$$f(\mathbf{x}) = \sum_{i=1}^{\ell} \alpha_i K(\mathbf{x}_i, \mathbf{x}) \quad \text{and} \quad g(\mathbf{x}) = \sum_{j=1}^{k} \beta_j K(\mathbf{y}_j, \mathbf{x}),$$

then the inner product between f and g is defined by

$$\langle f, g \rangle = \sum_{i=1}^{\ell} \sum_{j=1}^{k} \alpha_i \beta_j K(\mathbf{x}_i, \mathbf{y}_j) = \sum_{i=1}^{\ell} \alpha_i g(\mathbf{x}_i) = \sum_{j=1}^{k} \beta_j f(\mathbf{y}_j).$$

Interestingly, the inner product between f and $K(\mathbf{x}, \cdot)$ becomes f itself, since

$$\langle f, K(\mathbf{x}, \cdot) \rangle = \sum_{i=1}^{\ell} \alpha_i K(\mathbf{x}_i, \mathbf{x}) = f(\mathbf{x}).$$

Due to this "reproducing" property, $\mathcal{H}_K$ is called Reproducing Kernel Hilbert Space (RKHS).

4.1.2 Examples

In this section, some examples of kernel based classification are presented.

Kernel Mean Difference

Ignoring the norm constraint on $\mathbf{w}$ in (3.1), the linear Mean Difference (MD) classifier is

$$\begin{aligned} f(\mathbf{z}) &= \mathbf{w}^{\mathrm{T}} \mathbf{z} + b \\ &= (\bar{\mathbf{x}} - \bar{\mathbf{y}})^{\mathrm{T}} \mathbf{z} - (\bar{\mathbf{x}} - \bar{\mathbf{y}})^{\mathrm{T}} \frac{(\bar{\mathbf{x}} + \bar{\mathbf{y}})}{2}, \end{aligned} \tag{4.2}$$

where $\bar{\mathbf{x}}$ and $\bar{\mathbf{y}}$ are the sample means from Class $+1$ and -1, respectively. Note that (4.2) can be solely expressed by inner products, because

$$f(\mathbf{z}) = \frac{1}{n_1} \sum_{i=1}^{n_1} \langle \mathbf{x}_i, \mathbf{z} \rangle - \frac{1}{n_2} \sum_{j=1}^{n_2} \langle \mathbf{y}_j, \mathbf{z} \rangle + b, \tag{4.3}$$

where the threshold

$$b = -\frac{1}{2}\left(\frac{1}{n_1^2}\sum_{i,i'=1}^{n_1}\langle \mathbf{x}_i, \mathbf{x}_{i'}\rangle - \frac{1}{n_2^2}\sum_{j,j'=1}^{n_2}\langle \mathbf{y}_j, \mathbf{y}_{j'}\rangle\right),$$

and n_1 and n_2 are the numbers of the samples from each class.

Replacing $\langle\cdot,\cdot\rangle$ in (4.3) by a kernel function $K(\cdot,\cdot)$, the classifier of kernel (nonlinear) MD becomes

$$f(\mathbf{x}) = \frac{1}{n_1}\sum_{i=1}^{n_1} K(\mathbf{x}_i, \mathbf{z}) - \frac{1}{n_2}\sum_{j=1}^{n_2} K(\mathbf{y}_j, \mathbf{z}) + b, \tag{4.4}$$

where the threshold becomes

$$b = -\frac{1}{2}\left(\frac{1}{n_1^2}\sum_{i,i'=1}^{n_1} K(\mathbf{x}_i, \mathbf{x}_{i'}) - \frac{1}{n_2^2}\sum_{j,j'=1}^{n_2} K(\mathbf{y}_j, \mathbf{y}_{j'})\right).$$

Note that in the case $b = 0$, this kernel MD method is equivalent to the classical nonparametric discrimination method using the differences in density estimates (Hall and Wand, 1988).

Nonlinear Support Vector Machine

Recall the dual Lagrangian of SVM optimization in (3.12):

$$L_D = \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j t_i t_j \langle \mathbf{x}_i, \mathbf{x}_j\rangle.$$

Note that this optimization problem only depends on the labels and pairwise inner products between inputs. Thus the kernel version of SVM maximizes

$$L_D = \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j t_i t_j K(\mathbf{x}_i, \mathbf{x}_j),$$

and the resulting classifier is

$$f(\mathbf{z}) = \sum_{i=1}^{n}\hat{\alpha}_i K(\mathbf{z}, \mathbf{x}_i) + b. \tag{4.5}$$

4.1.3 Gaussian Kernel and the Infinite Dimensional Feature Space

One of the most widely used kernel functions is Gaussian Radial Basis Function (RBF) kernel, which is

$$K(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{y}\|^2}{2h^2}\right). \tag{4.6}$$

Here, h is the bandwidth, or the width parameter that needs to be chosen. As the bandwidth parameter in kernel density estimation controls how rough or smooth the estimated curve is, the h in (4.6) determines the smoothness of the classifying boundary, usually along with other tuning parameters, such as the penalty parameter C in SVM.

The Gaussian kernel function implicitly embeds data into an infinite dimensional feature space, since the kernel in (4.6) has an infinite number of basis functions, as in (4.1). Although it is not easy to understand this space in a Euclidean geometrical sense, we attempt to visualize the embedded data in the feature space in this section.

Let ϕ be the embedding function for the Gaussian kernel, i.e.,

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2h^2}\right) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle.$$

Note that ϕ has no closed form and each of the embedded data points $\phi(\mathbf{x}_i)$ lies within the infinite dimensional feature space $\mathcal{F}$. Nevertheless, the intrinsic dimensionality of $\mathcal{F}$ is known to be the sample size n (Shawe-Taylor and Cristianini, 2004, Chapter 9). The challenge is to understand this n-dimensional manifold.

Even though it is impossible to know the coordinates of embedded data points $\phi(\mathbf{x}_i)$, the pairwise distance between them can be easily calculated. The squared distance between the embedded data points, $\phi(\mathbf{x}_i)$ and $\phi(\mathbf{x}_j)$, is

$$\begin{aligned} \|\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)\|^2 &= \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_i) \rangle - 2\langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle + \langle \phi(\mathbf{x}_j), \phi(\mathbf{x}_j) \rangle \\ &= 2(1 - \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle) \\ &= 2\left[1 - \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2h^2}\right)\right]. \end{aligned} \tag{4.7}$$

The squared distance in (4.7) is an increasing function of Euclidean distance in the original space, and has the maximum value two. For a fixed h, if any two data vectors in the original space are close to each other, they are close in the feature space as well, having distance close to zero. If they have a large distance between each other originally, then the squared distance in the feature space is close to two. Note that Gaussian kernel carries only the proximity information from the original data: Any rotation around the origin or any parallel shift of the original data will not change the feature space. Also note that since the norm of the embedded data is always one (because $\langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_i) \rangle = 1$), all the embedded data vectors are in the unit sphere in the feature space.

If the bandwidth h is large, we can do Taylor series expansion as follows

$$\exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2h^2}\right) = 1 - \frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2h^2} + O(h^{-4}).$$

Thus the distance between $\phi(\mathbf{x}_i)$ and $\phi(\mathbf{x}_j)$ becomes

$$\|\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)\| = \frac{\|\mathbf{x}_i - \mathbf{x}_j\|}{h} + O(h^{-2}), \tag{4.8}$$

which means that the pairwise distances in the feature space are proportional to the ones in the original space. This implies that the embedding with a very large h keeps the geometry of the original data. Thus we can conjecture that kernel based classification with a very large h is equivalent to the linear classification in the original space.

On the other hand, if h is very small, the exponent in (4.7) becomes close to the negative infinity, which makes the distance between $\phi(\mathbf{x}_i)$ and $\phi(\mathbf{x}_j)$ close to $\sqrt{2}$, no matter what the original distance is. Noting that the feature space is very high dimensional, it can be seen that the embedded data are in the HDLSS geometric limit (Section 2.5), forming a regular n-simplex. Note that this argument holds regardless of the original dimensionality.

Some dimension reduction techniques such as the Multidimensional Scaling (MDS) generate low dimensional representations of any data, using only their pairwise proximity (distance) information. I.e., MDS projects the high dimensional data onto low dimensional space while trying to preserve the original pairwise proximity information as much as possible. See Cox and Cox (2000) for details about MDS. Since the pairwise distances in the feature space are readily given in (4.7), we attempt to visualize the infinite dimensional feature space using MDS as follows.

A target-shaped toy data example, shown in Figure 4.2, will be used for the illustration. This toy data set has 160 data points in $\mathcal{R}^2$, ranging about from -4 to 4 for each axis. The outermost and centermost points (80 points) are from Class $+1$, shown with the pluses $(+)$, and the middle ones (80 points) are from Class -1, shown with the circles (o). The Gaussian kernel function with $h = 10^{-2}, \cdots, 10^3$ is used. Figure 4.3 shows the result of two-dimensional MDS visualization. For $h = 10^{-2}$, the data points in feature space appear to be randomly spread over the surface of a sphere, which can be seen as a somewhat limited two dimensional visualization of the equidistance relationship, i.e., a simplex, as explained above. On the other hand, for a large h such as $h = 10^3$, it has the same target structure, as the original data, which can be easily explained from the above argument about the large h limit in (4.8).

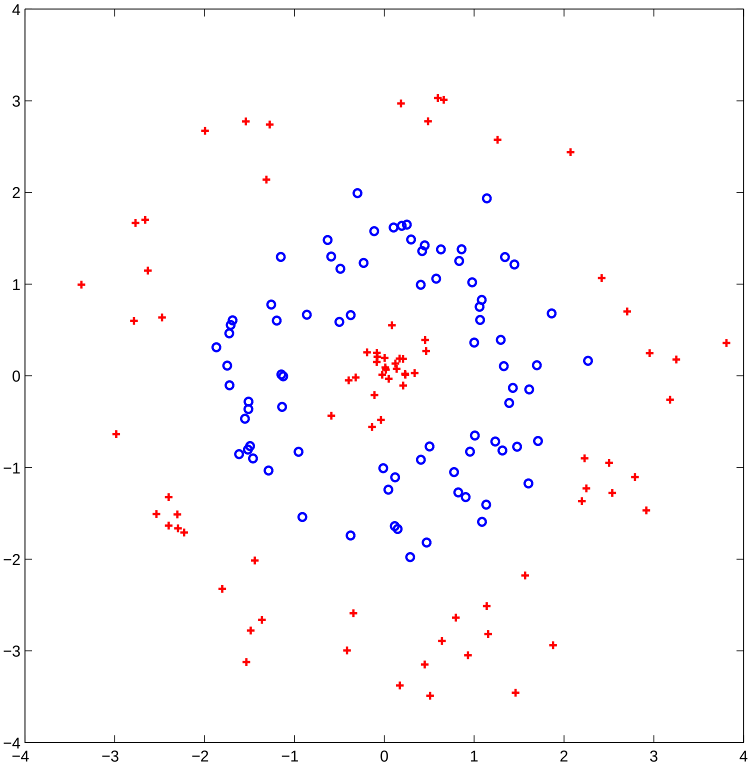

Figure 4.2: *Target-shaped toy data set*

Just as the bandwidth selection problem is crucial to kernel density estimation, it is also an

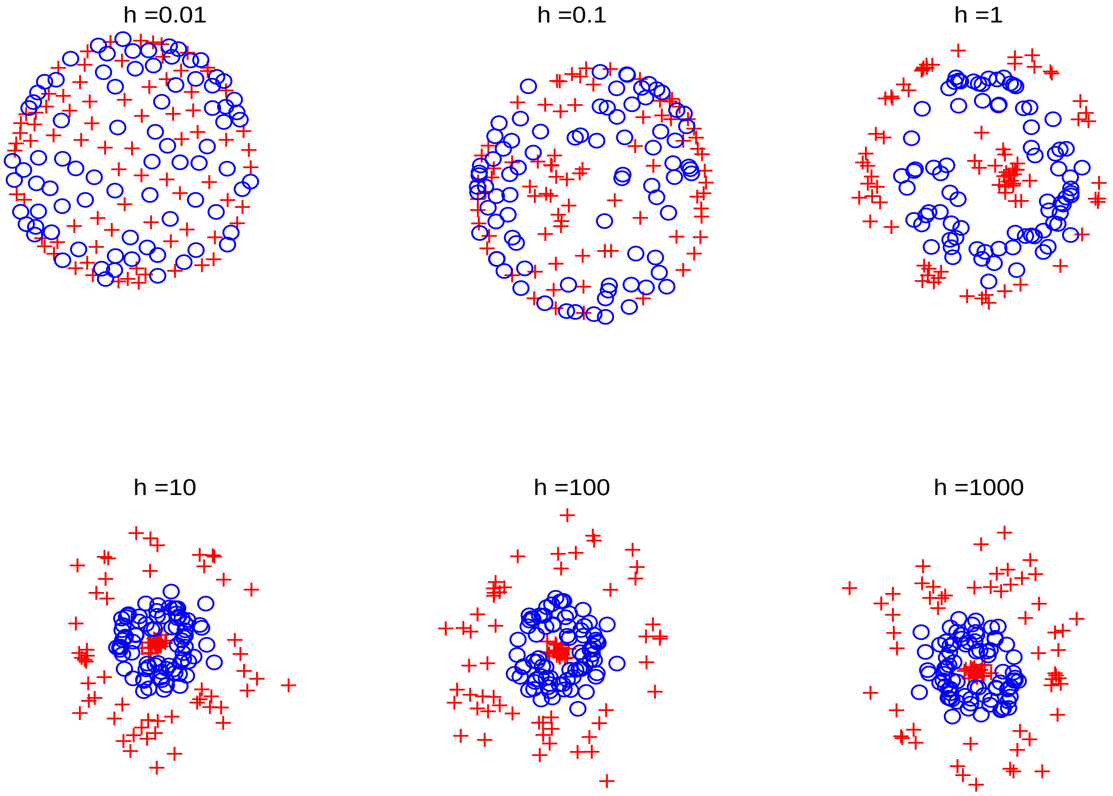

Figure 4.3: *Two dimensional MDS representations of the target toy data in the feature space generated by the Gaussian kernels with different choices of the bandwidths.*

important decision in kernel based classification (see for example Ben-Hur *et al.* (2001)) hence it is the main topic of this chapter. Some existing studies of the bandwidth selection problem, such as asymptotics and some current tuning methods, will be reviewed in Section 4.2.

In one dimensional kernel density estimation, there has been extensive work on bandwidth selection. (See the survey paper by Jones *et al.* (1996) for a good review.) We try some of those approaches in Section 4.3 to see if they can be useful in the classification setting as well. A novel bandwidth selection approach is introduced in Section 4.4. The proposed criterion is motivated by the following two facts: (1) different values of h construct different kernel embedded feature spaces, in particular, different geometry of the embedded data. (2) the kernel based classification is a linear classification in the embedded feature space. In Section 4.4.2, three real/simulated data examples are used to compare the proposed criterion and some existing methods.

4.2 Current Approaches to Bandwidth Selection

In this section, some existing studies on bandwidth selection are reviewed. Most of the studies are regarding the two tuning parameters of SVM, which is the penalty parameter C and the bandwidth parameter h. Also they treat these two parameters in the same way, in the sense that they are tuned by same criteria.

In Section 4.2.1, asymptotics of large and small tuning parameters are reviewed. Some existing tuning methods will be discussed in Section 4.2.2.

4.2.1 Asymptotics of the Bandwidth Parameter

The nonlinear SVM with Gaussian kernel involves careful tuning of two parameters, the penalty parameter C in (3.7) and the bandwidth h inside the Gaussian kernel function. Keerthi and Lin (2003) and Shawe-Taylor and Cristianini (2004, Chapter 9) studied asymptotic behaviors of the SVM classifier when these parameters take very large or very small values. Table 4.1 shows a quick summary of their conclusions.

	small h	large h
small C	underfitting	underfitting
large C	overfitting	underfitting

Table 4.1: *Asymptotic SVM classification performances with small/large tuning parameters*

Table 4.1 indicates that a very small h, when C is fixed to a sufficiently large value, yields severe "overfitting" which means that small regions around the training data points of the minority class are assigned to that class, while the rest of the data space is assigned to the majority class. Also a very small h, when C is fixed to a sufficiently small value, yields "underfitting" which means that the entire data space is assigned to the majority class. A very large h yields severe underfitting when C is fixed (no matter how large or small).

The results of Keerthi and Lin (2003) are illustrated using the target toy example in Figure 4.4. Nine different versions of the nonlinear SVM are considered over combinations of different values of h and C. The bright grey area is assigned to the Class $+1$ and the dark grey area is assigned to the Class -1 and the white area is the undecided area where the value of the classifier is zero.

As expected from the conclusion summarized in Table 4.1, Figure 4.4 shows severe under-

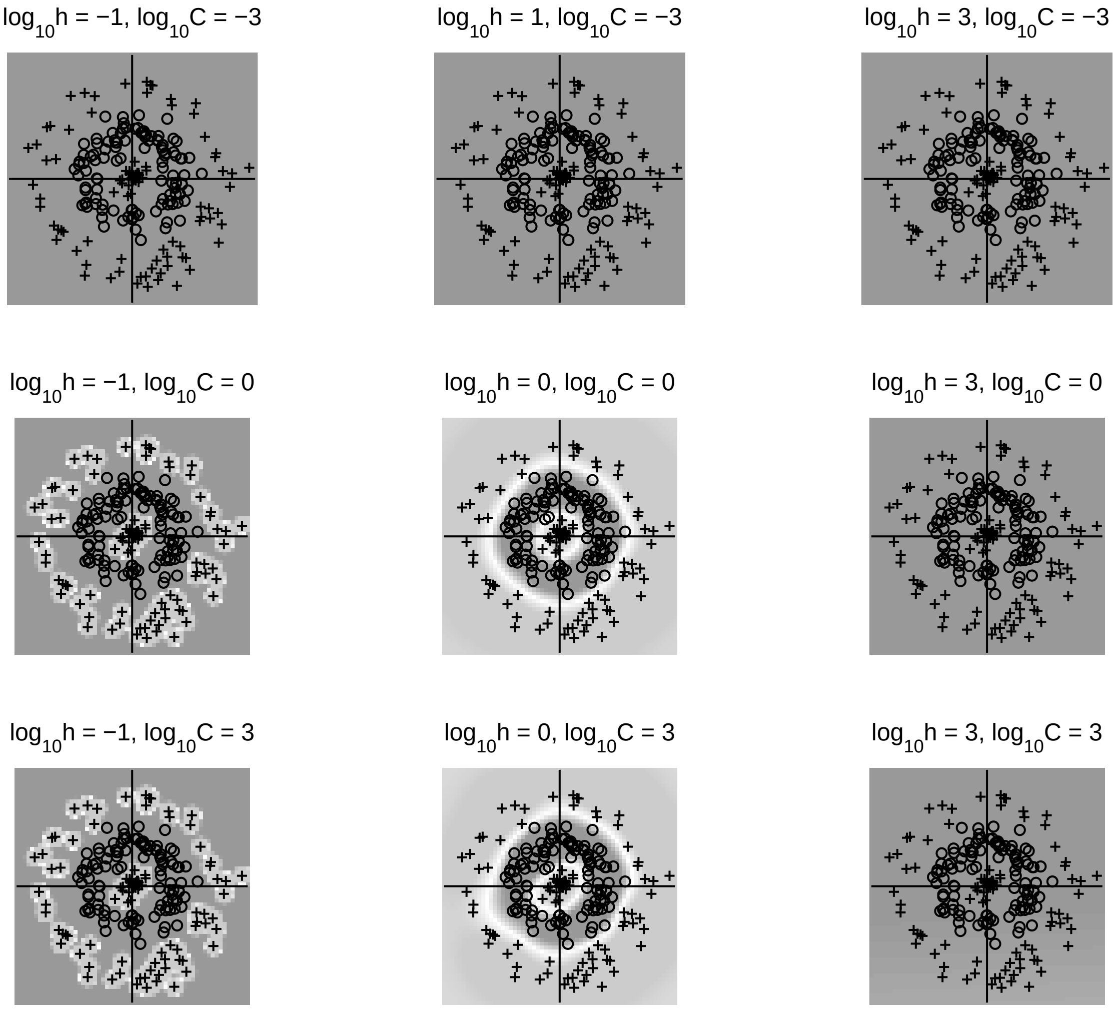

Figure 4.4: *SVM classification result on target-shaped toy example with different combinations of h and C*

fitting with a very small h (10^{-1}) and C (10^{-3}), as shown in the upper left plot. Also we have underfitting with a very large h (10^3) and any values of C, as shown in the plots in the right column. On the other hand, overfitting occurs with a very small h (10^{-1}) and large C ($10^0, 10^3$), as seen in the bottom two plots on the left column. The two bottom plots in the center column show that we have good classification performances with a reasonable choice of h (10^0), when C is reasonably large ($10^0, 10^3$).

4.2.2 Existing Tuning Methods

Many approaches to the bandwidth selection problem in kernel based classification have focused on minimizing the estimated misclassification rate. Cross-validation (CV) is the most widely used method for the error estimation.

Suppose an separate tuning data set is available, which is generated from the same underlying probability model as the training and testing set. Cross-validation applies a wide range of tuning parameters to the training data to obtain a classifier, and then selects the values of parameters that yield the smallest error in the tuning set, i.e., the smallest cross validation error. The finally fitted model (classifier) is based on the training data with the selected tuning parameters, and the reported misclassification rate estimate is evaluated from the testing data. In this way, model fitting and model evaluation precesses are separate so that we do not create a classifier biased to the testing data.

If a separate tuning data set is not available, as for most of real-world situations, the V-fold cross-validation and leave-one-out cross-validation are popular choices to estimate the cross-validation error. The V-fold cross-validation divides the training data set into V roughly equal-sized subsets. For the vth subset, a classification method is applied to the other $V - 1$ subsets and the misclassification rate is calculated from the vth subset that has been left out. We repeat this for $v = 1, \cdots, V$ and estimate the cross-validation error by averaging the V error estimates. The tuning parameters that minimize this cross-validation error is chosen. 5 or 10 is the most popular choice for V, and the case when $V =$ the sample size, is known as the leave-one-out cross-validation. The fitted model is calculated using all the training data with the selected tuning parameters and its reporting error is evaluated with the testing data set if it is available.

Sometimes we only have a training data set, without the luxury of having separate data sets for tuning and testing the classifier. In that case, the double-layered V-fold cross-validation is recommended. I.e., we divide the data set to V subsets, and then we treat each $V - 1$ subset as the training set and we do the tuning as explained in the previous paragraph, by splitting this subset again into V' subsets. The left-out subset is the testing set to evaluate the fitted model. In this way, we end up with V fitted models and V misclassification rate estimates, whose average is reported.

Cross-validation has been used for bandwidth selection in kernel density estimation for a long time. However, it also has been realized that cross-validation is of very limited practical value due to the large sampling variability. It is quite likely that the practice of cross-validation for the classification task also would have a similar disadvantage. We will empirically show its large sampling variability in classification in Section 4.4.2.

Also the computational complexity of cross-validation is one of the reasons that many researchers have been trying to find a faster way of tuning parameters. For example, Wahba *et al.* (2000) proposed a tuning criterion called the Generalized Approximate Cross Validation (GACV), which is basically an upper bound for the expected misclassification rate of a future observation. The Xi-alpha method by Joachims (2000) uses a criterion which is an upper bound of the leave-one-out cross validation error.

Duan *et al.* (2003) compared various tuning methods for SVM with some benchmark data sets (Rätsch, 1999). They compared 5-fold cross-validation, GACV, X-alpha, Approximate span bound by Vapnik and Chapelle (2000), VC bound by Burges (1998), Radius-margin bound by Vapnik and Chapelle (2000), and Modified radius-margin bound by Chapelle *et al.* (2002). In each training, they fixed the penalty parameter C at some value and varied the bandwidth h, and then fixed h and vary the value of C. They concluded 5-fold cross-validation gives overall best performances over various data sets. GACV and Xi-alpha show good performances for some data sets, especially GACV has smoother variation with respect to tuning parameters. Other bounds could not give a useful result, probably because these approximate bounds are too loose.

4.3 Some Ideas from Kernel Density Estimation

Here we employ some bandwidth selection ideas from kernel density estimation, where this problem has been extensively studied. Among them, we adopt the over-smoothing idea (Section 4.3.1) and the scale-space approach (Section 4.3.2).

4.3.1 Maximal Smoothing Bandwidth Idea from Kernel Density Estimation

Terrell (1990) proposed the maximal smoothing principle which is originated from the over-smoothing idea by Terrell and Scott (1985). The maximal smoothing principle in kernel density estimation allows the most smoothing that is consistent with the estimated scale of the data. We will call the bandwidth h which is chosen by this principle the maximal smoothing bandwidth, denoted by h_{MS}. The maximal smoothing bandwidth is supposed to be the most conservative choice, in the sense that it should be used as a possible upper bound for the bandwidth selection procedure.

In this section we apply the maximal smoothing bandwidth to kernel based classification. The same toy example in the previous section is again used here. The formula for the maximal smoothing principle for multivariate density estimation is given in Terrell (1990, p.474). The $d \times d$ matrix bandwidth $\mathbf{A}$ is any solution of

$$\mathbf{A}\mathbf{A}^{\mathrm{T}} = \left[\frac{(d+8)^{(d+6)/2}\pi^{d/2} \int K^2}{16n\Gamma[(d+8)/2](d+2)} \right]^{2/(d+4)} \hat{\mathbf{\Sigma}}, \tag{4.9}$$

where $\hat{\mathbf{\Sigma}}$ is the sample covariance matrix. Note that here K is the d-dimensional Gaussian density with identity covariance matrix and mean zero. Adjusting (4.9) to the single bandwidth case, we have

$$h_{\mathrm{MS}} = \left[\frac{(d+8)^{(d+6)/2}\pi^{d/2} \int K^2}{16n\Gamma[(d+8)/2](d+2)} \right]^{1/(d+4)} \hat{\sigma}, \tag{4.10}$$

where $\hat{\sigma}$ is the pooled estimate of the standard deviation.

For the target toy data, we have $d = 2$ and $n = n_1 + n_2 = 80 + 80 = 160$. The maximal

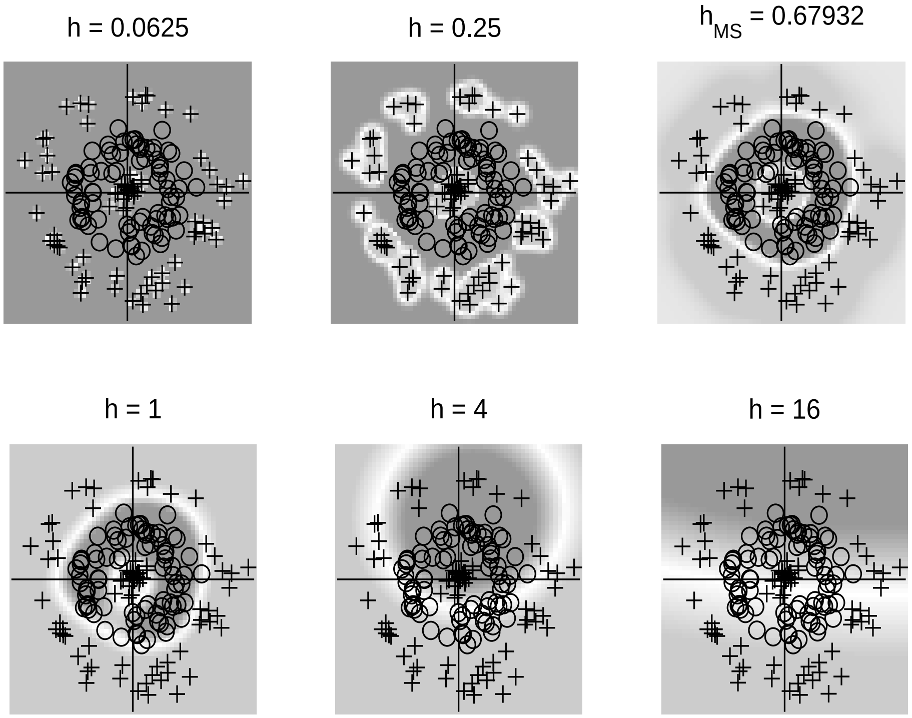

Figure 4.5: *Kernel MD Classification result of the maximal smoothing bandwidth h_{MS} for target-shaped toy example, shown with a range of h*

smoothing bandwidth (4.10) for this data set is

$$h_{\mathrm{MS}} = \left[\frac{10^4\pi(4\pi)^{-1}}{16 \times 160\Gamma(5)4}\right]^{1/6} \hat{\sigma} \tag{4.11}$$

$$= 0.67932. \tag{4.12}$$

Kernel MD is used as a classification rule because it is easy to implement and there is no other tuning parameter. Figure 4.5 shows the kernel MD classification results over a range of h. We can see a reasonably smooth separating boundary when $h = h_{\mathrm{MS}}$, while a smaller h yields overfitting and a larger h yields underfitting. Hall and Kang (2005) argued that the optimal bandwidth for classification is the same as that which would be used when we were constructing pointwise density estimators in low dimensional, high sample size settings. This argument may explain the reasonable classification performance of the maximal smoothing bandwidth in this example.

Figure 4.6 shows the between class pairwise distances and within class pairwise distances in

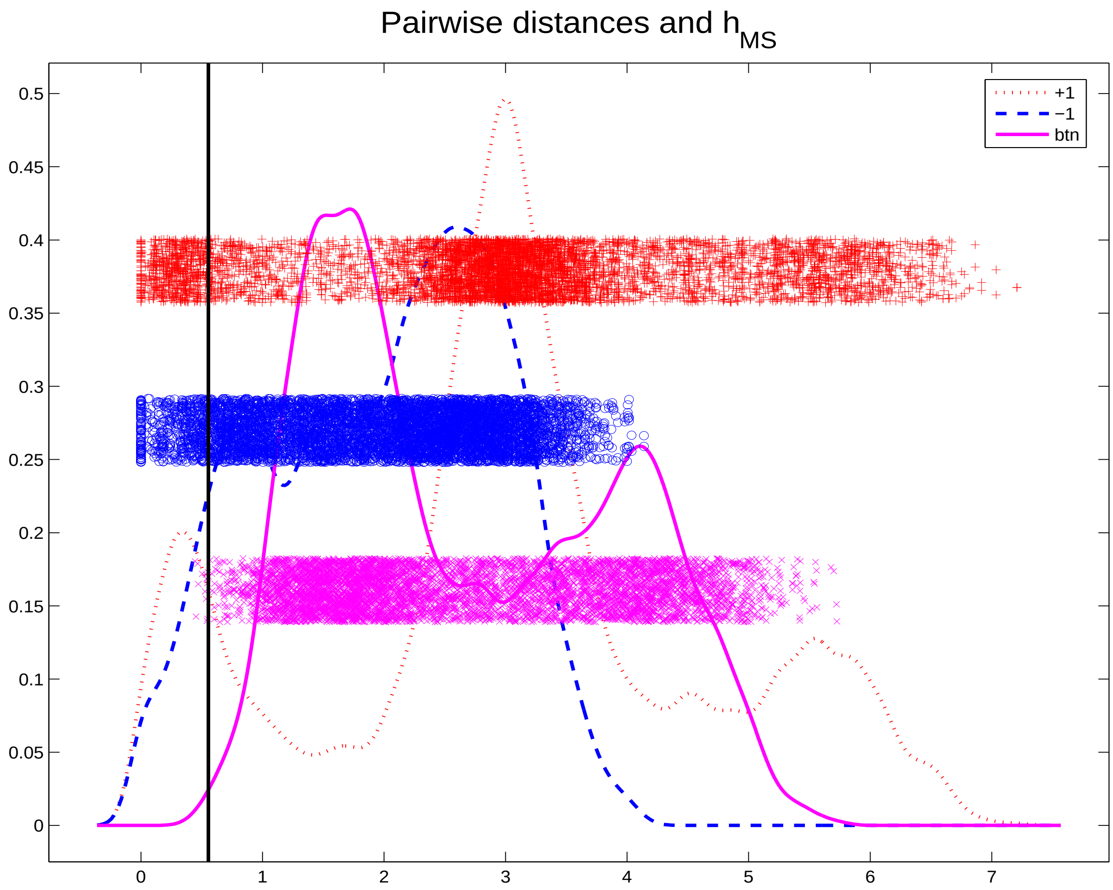

Figure 4.6: *Pairwise distances in the feature space and the maximal smoothing bandwidth h_{MS}. The '+' symbol and 'o' are used for Class +1 and Class −1, respectively. The within pairwise distances of Class +1 (−1) are shown in the top (middle) jittered layer and their kernel density estimates are shown with the dotted (dashed) curve. The between class pairwise distances are shown as the x's in the bottom jittered layer and the kernel density estimate is shown with the solid curve. The maximal smoothing bandwidth $h_{\mathrm{MS}} = 0.68$ is shown with the vertical line.*

the feature space in a jitter plot. (See Section 3.2 for detailed explanation about this type of visualization and the caption of the figure for a detailed description.) The first small peak of the dotted curve around 0.5 is for the distances within the points around the origin, the second large peak around 3 is for the distances between the points in the center and the ones in the outer ring, and the third peak around 5.4 is for the distances between the points lying on the opposite sides of the outer ring. Also the largest peak of the dashed curve around 2.7 is for the distances between the points lying on the opposite sides in the inner ring. The first big peak of the solid curve around 1.6 is for the distances between the points from different classes but with a similar polar angle and the second somewhat small peak around 4.1 is for the ones in the opposite directions.

Also, the maximal smoothing bandwidth h_{MS} is also shown as the vertical line in Figure 4.6. Note that h_{MS} is very close to the minimum between-class pairwise distance. This fact may account for the good performance of the maximal smoothing bandwidth, implying that the good bandwidths must be chosen to be smaller than at least the majority of the between class pairwise distances.

4.3.2 Scale-space Approach

Chaudhuri and Marron (1999) adopted the scale-space approach for bandwidth selection for kernel density estimation. In the scale-space approach, we use a range of bandwidths instead of a single chosen bandwidth. Figure 4.5 can be viewed as a scale-space approach to the bandwidth selection problem. A movie version of the figure, with a finer range of h, can be found at:

http://www.unc.edu/~jyahn/target_change_h.avi.

In kernel density estimation, the scale-space approach can be useful for an overview of the data at a range of scales. Likewise, this approach can be used to get some ideas on the data structure. In Figure 4.7, two toy examples in $\mathcal{R}^2$ are shown with kernel MD training errors over a range of bandwidths. The toy data set shown in the top left panel is the same target data set used in the previous examples, and the second data set shown in the top right panel is a linearly separable point cloud toy data set. The bottom left panel shows the training error rates for the target data and the bottom right panel shows for the point cloud data.

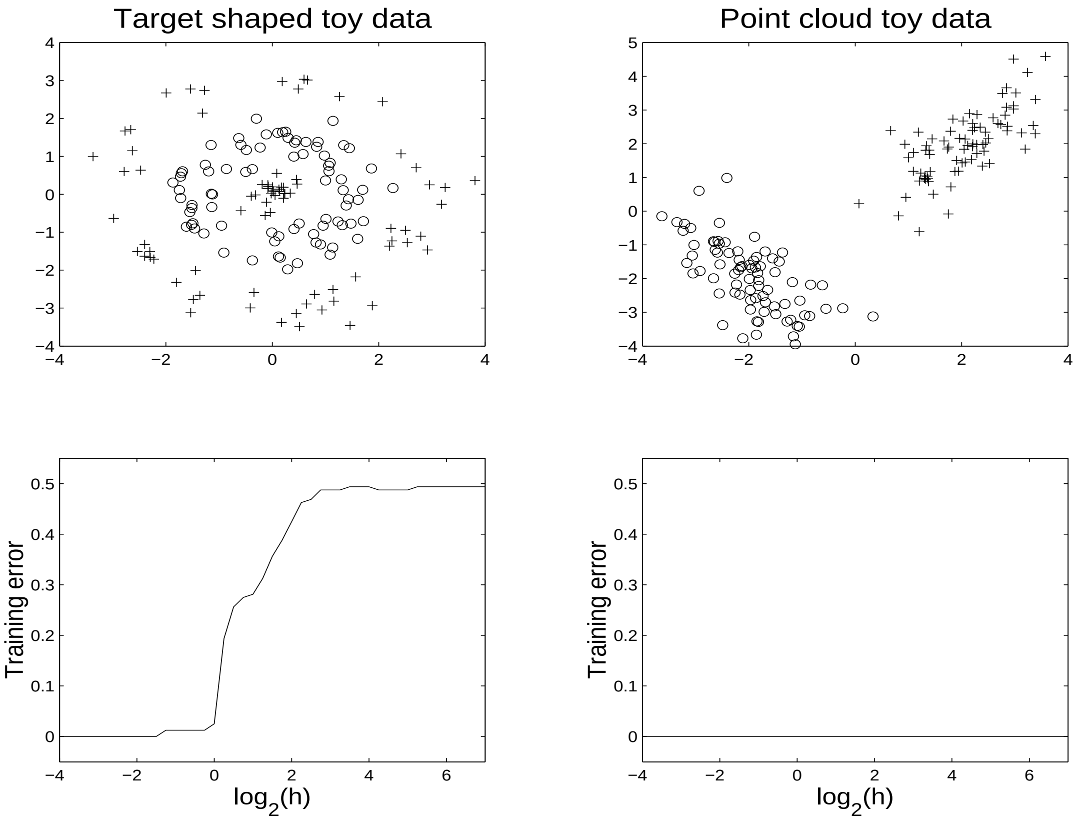

Figure 4.7: *Training error of kernel MD over a range of h, for target-shaped data and point cloud data. Data are shown in the top panels and the training errors are shown in the bottom panels.*

Note that the training error curve on the left exhibits an S-shape, with the error being close to zero until $\log_2(h)$ is about 0. The error then increases rapidly until $\log_2(h)$ reaches around 2.8, and very close to a half for bigger bandwidths, where the kernel classification is essentially linear, in the spirit of the discussion in Section 4.1.3. In this sense, a desirable choice of h is under 2^0, because within the range $0 < \log_2(h) < 2.8$, a slight change in h yields a big difference in the training error, and the error becomes too big after that.

The bottom right panel in Figure 4.7 shows that for the point cloud data the training error is always zero no matter what value of h is selected. Here the large h also has zero training error, because the geometry of the data in the feature space is actually very similar to the original space, thus the embedded data points are also linearly separable.

When selecting the bandwidth parameter, hence eventually the classifier, the training error itself is not at all enough. One must consider other criteria such as the test error to ensure the good generalizability of the classifier. However, as we observed in Figure 4.7, training error over a range of bandwidth can give an insight of the data structure, especially when the data are multi-dimensional.

4.4 New Approach

In this section we propose a new idea for selecting the bandwidth parameter in kernel based classification. Even though only Gaussian kernel is used to illustrate the method here, the idea can be applied to any other kernel function.

4.4.1 Motivation

The idea of data embedding is to obtain a complicated (nonlinear) decision boundary by implementing a simple (linear) task in the high dimensional embedded feature space. For example, kernel SVM does linear SVM in the feature space, searching for the separating hyperplane that maximizes the margin between the classes, i.e., the distance between the convex hulls of each class. As shown in Figure 4.3, different bandwidths yield different geometry in the feature space. The bandwidth parameter that yields a good nonlinear classifier also makes a good linear classifier with the embedded data in the feature space. In this spirit, we want the desirable bandwidth to make this linear task "easier".

Since the feature space generated by the Gaussian kernel only keeps the information about the pairwise distances between the data points, it makes sense to take those distances into account. For the classification purpose, it would be easier to separate the two classes when the within class pairwise distances are relatively smaller than the between pairwise distances.

Suppose $\mathbf{x}_1, \cdots, \mathbf{x}_{n_1}$ are data vectors from Class $+1$ and $\mathbf{y}_1, \cdots, \mathbf{y}_{n_2}$ are from Class -1. Let us denote the difference between the average of between squared pairwise distances and the average of within squared pairwise distances by D.

$$\begin{aligned} D &\equiv \frac{1}{n_1 n_2} \sum_i^{n_1} \sum_j^{n_2} \|\phi(\mathbf{x}_i) - \phi(\mathbf{y}_j)\|^2 \\ &\quad -\frac{1}{2} \left(\frac{1}{n_1(n_1-1)} \sum_{i \neq j} \|\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)\|^2 + \frac{1}{n_2(n_2-1)} \sum_{i \neq j} \|\phi(\mathbf{y}_i) - \phi(\mathbf{y}_j)\|^2 \right) \\ &= \frac{1}{n_1(n_1-1)} \sum_{i \neq j} K(\mathbf{x}_i, \mathbf{x}_j) + \frac{1}{n_2(n_2-1)} \sum_{i \neq j} K(\mathbf{y}_i, \mathbf{y}_j) - \frac{2}{n_1 n_2} \sum_i^{n_1} \sum_j^{n_2} K(\mathbf{x}_i, \mathbf{y}_j) \end{aligned}$$

$$
\begin{aligned}
= \quad & \frac{1}{n_1(n_1-1)}\sum_{i\neq j}\exp\left(-\frac{\|\mathbf{x}_i-\mathbf{x}_j\|^2}{2h^2}\right)+\frac{1}{n_2(n_2-1)}\sum_{i\neq j}\exp\left(-\frac{\|\mathbf{y}_i-\mathbf{y}_j\|^2}{2h^2}\right)\\
& -\frac{2}{n_1n_2}\sum_{i}^{n_1}\sum_{j}^{n_2}\exp\left(-\frac{\|\mathbf{x}_i-\mathbf{y}_j\|^2}{2h^2}\right).
\end{aligned}
$$

We suggest to choose h that maximizes D or to use the range of the bandwidths with $D > 0$. When $D < 0$ for all h, it is still desirable to use h which is the maximizer of D.

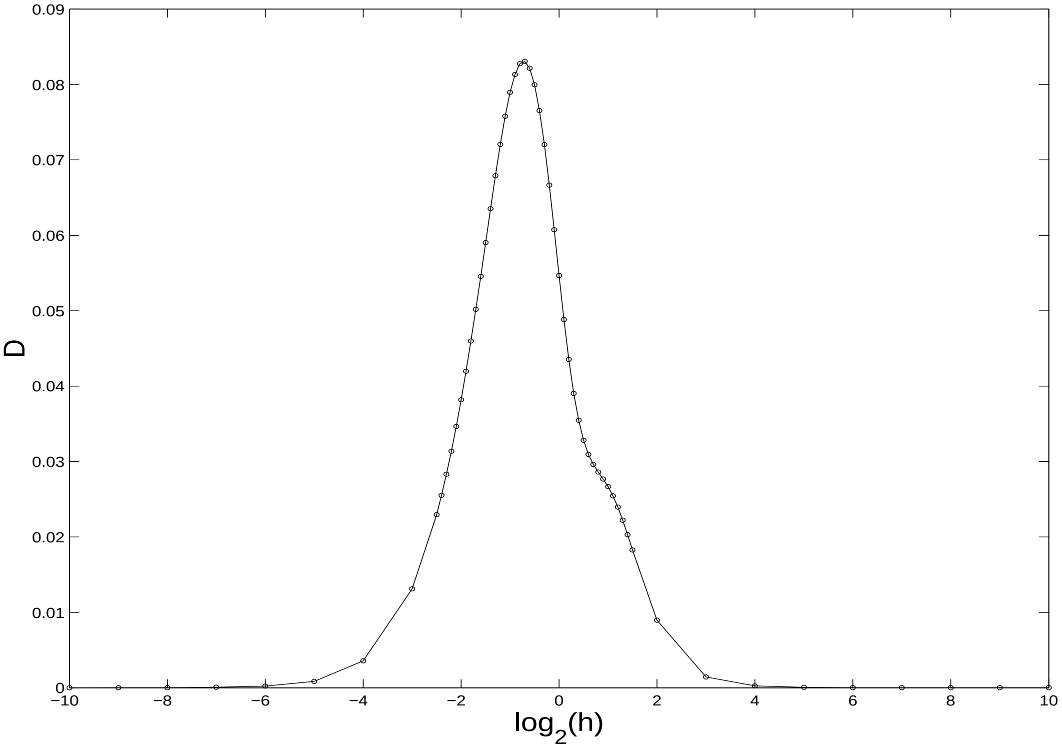

Figure 4.8: *Values of D for target toy example over a range of bandwidth h*

Now we apply this approach to the target toy data example from the previous section. A wide range of $h = 2^{-10}, \cdots, 2^{10}$ are considered and the corresponding D is calculated (Figure 4.8). Note that D is always positive in this range of h and has the highest value at $h = .6156$. Hence $h = .6156$ is our best choice of bandwidth according to this criterion. It is not particularly surprising regarding the result in Section 4.3.1, where the maximal smoothing bandwidth h_{MS}, purportedly an upper bound of a reasonable bandwidth, was 0.6793, which was slightly higher than the chosen bandwidth here.

Note that this method cannot be applied to tune the other tuning parameters outside the kernel function such as the penalty parameter C in SVM, because those parameters are independent of the feature space. When there are other parameters to tune, not inside the kernel function, we suggest to use the V-fold cross-validation. Also note that unlike other methods such as GACV and Xi-alpha, this criterion does not depend on particular discrimination methods. Thus given a data set, the selected bandwidth is the same for any discrimination method, since only the kernel function, not the specific classification rule, determines the feature space, and the bandwidth parameter is only related to that kernel function.

4.4.2 Data Examples

In this section we demonstrate the proposed criterion D in the previous section using three real/simulated data examples. The first data set is "banana" toy data set from Rätsch *et al.* (2001), which has two input variables, 400 training samples, 4900 test samples, and 100 realizations. The second data set is "diabetes" data set from 1994 AAAI Spring Symposium on Artificial Intelligence in Medicine, available at "http://www.ics.uci.edu/ mlearn/MLSummary.html". It has 8 input variables, 468 training samples, 300 test samples, and 100 realizations. The third data set is the "target" toy data set used in the previous sections. We generate 100 training samples and 1000 test samples, and repeat 100 times. The kernel MD and SVM with Gaussian kernel is used, with tuning methods 5-fold cross validation (CV) and the D criterion for MD and additionally GACV and Xi-alpha for SVM.

Figures 4.9 and 4.10 show boxplots of test errors, computing time, and selected bandwidths from 100 realizations of the data for MD and SVM, respectively.

The leftmost column of Figure 4.9 shows that for MD, 5-fold CV does slightly better than D for all three data sets. However, the center column shows a huge difference in computing times, favorable to our method. The selected bandwidths for 5-fold CV in the right column of the figure vary significantly , while D only picks one or two distinct values of h. Note that since the 100 realizations of the data are from the same underlying structure, it makes better sense if we choose the same bandwidth for every realizations.

Figure 4.10 has the results for SVM. For all three data sets, D shows much stronger performances than GACV and Xi-alpha, having almost the same performance as 5-fold CV.

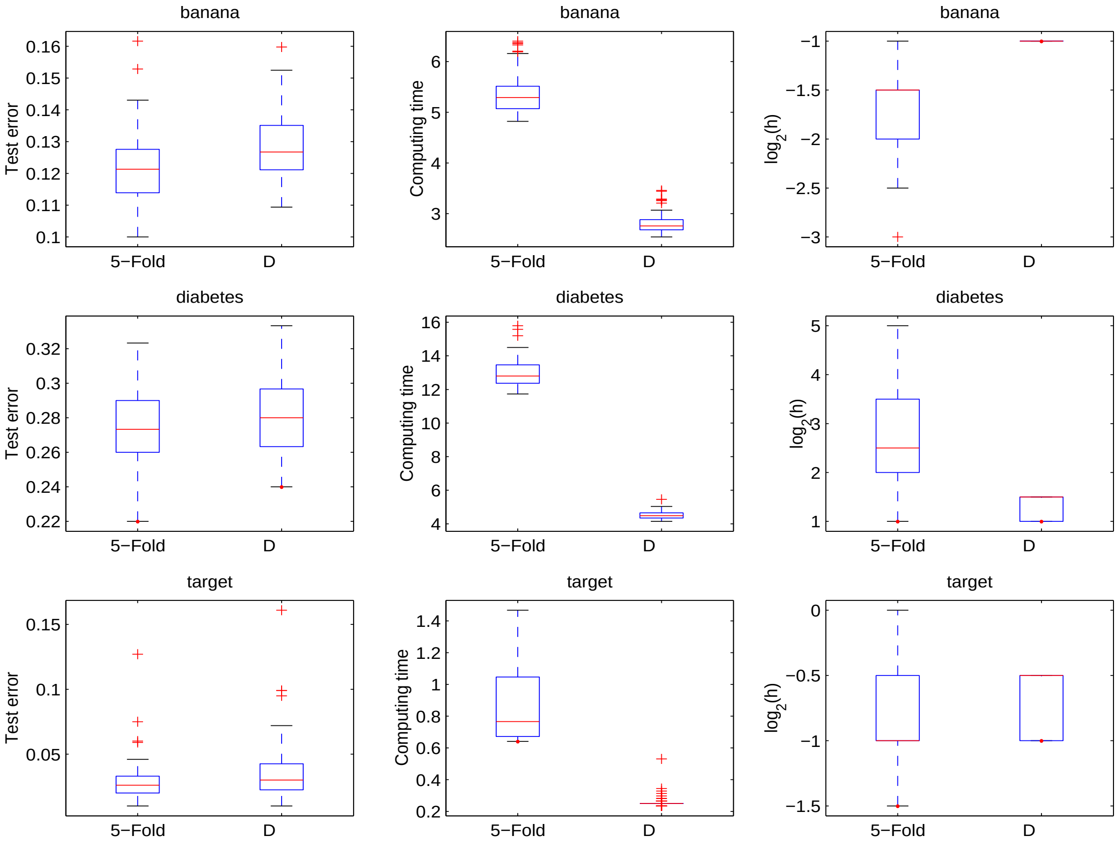

Figure 4.9: *Boxplots of test errors, computing time, and selected bandwidth of kernel MD for three data sets.*

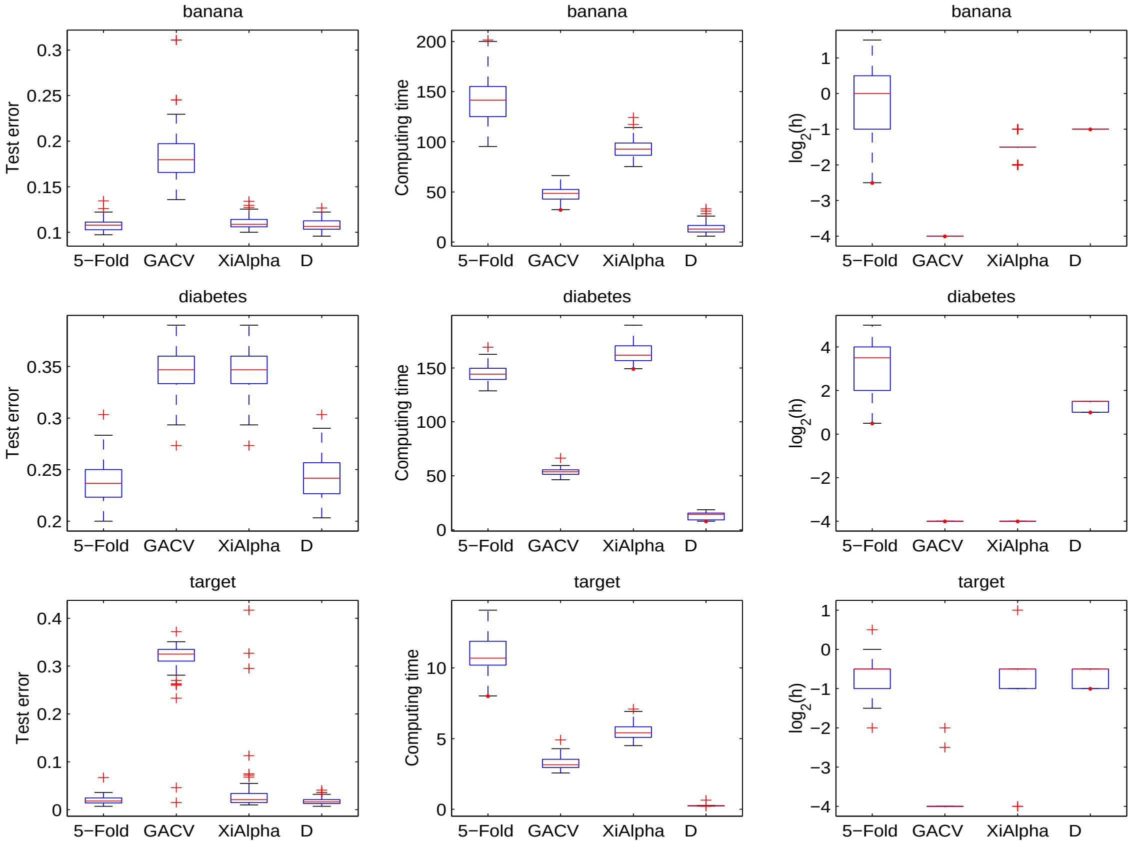

Figure 4.10: *Boxplots of test errors, computing time, and selected bandwidth of kernel SVM for three data sets.*

GACV shows poor performances, but it takes less time to compute than Xi-alpha. The performance of Xi-alpha varies from data to data and also it has huge outliers in the boxplot of test errors for the "target" data. The rightmost panels of selected bandwidths again shows much larger sampling variability of 5-fold CV than the other methods. GACV and Xi-alpha have some stability in choosing bandwidth in the first two examples, however, in the "target" data, they both have outliers, which suggests a possible instability.

BIBLIOGRAPHY

Aizerman M., Braverman E. and Rozonoer L. (1964). Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control* pp. 821–837.

Anderson T.W. (2003). *An Introduction to Multivariate Statistical Analysis*. Wiley-Interscience, 3rd edition.

Bai Z.D. (1999). Methodologies in spectral analysis of large dimensional random matrices, a review. *Statistica Sinica* **9**, 611–677.

Bai Z.D., Miao B. and Yao J.F. (2003). Convergence rates of spectral distributions of large sample covariance matrices. *SIAM Journal on Matrix Analysis and Applications* **25**, 105–127.

Bai Z.D. and Yin Y.Q. (1993). Limit of the smallest eigenvalue of a large dimensional sample covariance matrix. *The Annals of Probability* **21**, 1275–1294.

Baik J. and Silverstein J.W. (2004). Eigenvalues of large sample covariance matrices of spiked population models. ArXive:math.ST/048165 v1.

Ben-Hur A., Horn D., Siegelmann H.T. and Vapnik V. (2001). Support vector clustering. *Journal of Machine Learning Research* pp. 125–137.

Benito M., Parker J., Du Q., Wu J., Xiang D., Perou C.M. and Marron J.S. (2004). Adjustment of systematic microarray data biases. *Bioinformatics* pp. 105–144.

Bickel P. and Levina E. (2004). Some theory for Fisher's linear discriminant function, "naive bayes", and some alternatives when there are many more variables than observations. *Bernoulli* **10**, 989–1010.

Birke M. and Dette H. (2003). A note on testing the covariance matrix for large dimension. In manuscript, found at "http://www.sfb475.uni-dortmund.de/berichte/tr02-04.ps".

Bradley P. and Mangasarian O. (1954). Feature selection via concave minimization and support vector machines. In: *ICML'98*, edited by J. Shavlik. Morgan Kaufmann.

Buja A., Cook D. and Swayne D.F. (1996). Interactive high-dimensional data visualization. *Journal of Computational and Graphical Statistics* **5**, 78–99.

Buja A., Hastie T. and Tibshirani R. (1995). Penalized discriminant analysis. *The Annals of Statistics* **23**, 73–102.

Burges C.J.C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery* **2**, 121–167.

Burman P. (1989). A comparative study of ordinary cross-validation, v-fold cross-validation and the repeated learning-testing methods. *Biometrika* **76**, 503–514.

Chapelle V., Vapnik V., Bousquet O. and Mukherjee S. (2002). Choosing kernel parameters for support vector machines. *Machine Learning* **46**, 131–160.

Chaudhuri P. and Marron J.S. (1999). SiZer for exploration of structure in curves. *Journal of the American Statistical Association* **94**, 807–823.

Cox T.F. and Cox M.A.A. (2000). *Multidimensional Scaling*. CRC Press, 2 edition.

Cristianini N. and Shawe-Taylor J. (2000). *An introduction to Support Vector Machines and other kernel-based learning methods*. Cambridge University Press.

Deift P. (1999). Integrable systems and combinatorial theory. *Notices of the American Mathematical Society* **47**, 631–640.

Donoho D.L. and Tanner J. (2005). Neighborliness of randomly-projected simplices in high dimensions. Technical report, Stanford University.

Duan K., Keerthi S.S. and Poo A.N. (2003). Evaluation of simple performance measures for tuning svm parameters. *Neurocomputing* **51**, 41–59.

Duda R.D., Hart P.E. and Stork D.G. (2000). *Pattern Classification*. Wiley-Interscience.

Fujikoshi Y. (2004). Multivariate analysis for the case when the dimension is large compared to the sample size. *Journal of the Korean Statistical Society* **33**, 1–24.

Furey T.S., Christianini N., Duffy N., Bednarski D.W., Schummer M. and Hauessler D. (2000). Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics* **16**, 906–914.

Geman S. (1980). A limit theorem for the norm of random matrices. *The Annals of Probability* **8**, 252–261.

Golub T.R., Slonim D.K., Tamayo P., Huard C., Gaasenbeek M., Mesirov J.P., Coller H., Loh M., Downing J.R., Caligiuri M.A., Bloomfield C.D. and Lander E.S. (1999). Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science* **286**, 531–537.

Hall P. and Kang K.H. (2005). Bandwidth choice for nonparametric classification. *The Annals of Statistics* **33**, 284–306.

Hall P., Marron J.S. and Neeman A. (2005). Geometric representation of high dimension, low sample size data. *Journal of the Royal Statistical Society, Series B* **67**, 427–444.

Hall P. and Wand M.P. (1988). On nonparametric discrimination using density differences. *Biometrika* **75**, 541–547.

Hastie T. and Tibshirani R. (2004). Efficient quadratic regularization for expression arrays. *Biostatistics* **5**, 329–340.

Hastie T., Tibshirani R. and Friedman J. (2001). *The Elements of Statistical Learning: data mining, inference, and prediction*. Springer.

Hoyle D. and Rattray M. (2004). Principal component analysis eigenvalue spectra from data with symmetry breaking structure. *Physical Review E* **69**, 026124.

Inselberg A. (1985). The plane with parallel coordinates. *The Visual Computer* **1**, 69–91.

Joachims T. (2000). Estimating the generalization performance of a SVM efficiently. In: *Proceedings of the International Conference on Machine Learning*. Morgan Kaufman.

John S. (1971). Some optimal multivariate tests. *Biometrika* **58**, 123–127.

John S. (1972). The distribution of a statistic used for testing sphericity of normal distributions. *Biometrika* **59**, 169–173.

Johnson B.A., Lin D., Marron J.S., Ahn J., Parker J. and Perou C.M. (2005). Distance weighted discrimination with censored outcomes. In manuscript.

Johnstone I.M. (2001). On the distribution of the largest eigenvalue in principal components analysis. *The Annals of Statistics* **29**, 295–327.

Johnstone I.M. and Lu A.Y. (2004). Sparse principal component analysis. To appear in *Journal of the American Statistical Association.*

Jones M.C., Marron J.S. and Sheather S.J. (1996). A brief survey of bandwidth selection of density estimation. *Journal of the American Statistical Association* **91**, 401–407.

Kachigan S.K. (1991). *Multivariate Statistical Analysis: A Conceptual Introduction*. Radius Press.

Keerthi S.S. and Lin C.J. (2003). Asymptotic behaviours of support vector machines with gaussian kernel. *Neural Computation* **15**, 1667–1689.

Ledoit O. and Wolf M. (2002). Some hypothesis tests for the covariance matrix when the dimension is large compared to the sample size. *The Annals of Statistics* **30**, 1081–1102.

Mardia K.V. (1980). *Multivariate Analysis (Probability and Mathematical Statistics)*. Academic Press.

Marron J.S., Todd M. and Ahn J. (2005). Distance weighted discrimination. To appear in *Journal of the American Statistical Association.*

Marčenko V.A. and Pastur L.A. (1967). Distribution of eigenvalues of some sets of random matrices. *Mathematics of the USSR-Sbornik* **1**, 457–483.

Mauchly J.W. (1940). Significance test for sphericity of a normal n-variate distribution. *Annals of Mathemetical Statistics* **11**, 204–209.

Mercer J. (1909). Functions of positive and negative type and their connection with the theory of integral equations. *Philosophical Thansactions of the Royal Society of London, Series A* **209**, 415–446.

Muirhead R.J. (1982). *Aspects of Multivariate Statistical Theory*. Wiley-Interscience.

Muller K.E., Chi Y., Marron J.S. and Ahn J. (2005). High dimension, low sample size principal components: estimating eigenvalues of a singular Wishart. In manuscript.

Nagao H. (1973). On some test criteria for covariance matrix. *The Annals of Statistics* **1**, 700 – 709.

Paul D. (2005). Asymptotics of the leading sample eigenvalues for a spiked covariance model. Technical report, Stanford University.

Perou C.M., Sorlie T., Eisen M.B., van de Rijn M., Jeffrey S.S., Rees C.A., Pollack J.R., Ross D.T., Johnsen H., Akslen L.A., Fluge O., Pergamenschikov A., Williams C., Zhu S.X., Lonning P.E., Borresen-Dale A.L., Brown P.O. and Botstein D. (2000). Molecular portraits of human breast tumours. *Nature* **406**, 747–752.

Prähofer M. and Spohn H. (2003). http://www-m5.ma.tum.de/kpz/f1.040.200.0016.txt. Website.

Rätsch G. (1999). http://ida.first.fraunhofer.de/projects/bench/benchmarks.htm. Website.

Rätsch G., Onoba T. and Mueller K.R. (2001). Soft margins for AdaBoost. *Machine Learning* **42**, 287–320.

Schölkopf B. and Smola A.J. (2001). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond (Adaptive Computation and Machine Learning)*. The MIT Press.

Searle S.R. (1982). *Matrix Algebra Useful for Statistics*. John Wiley-Sons Inc.

Shawe-Taylor J. and Cristianini N. (2004). *Kernel Methods for Pattern Analysis*. Cambridge University Press.

Shen X., Tseng G.C., Zhang X. and Wong W.H. (2003). On ψ-learning. *Journal of the American Statistical Association* **98**, 724–734.

Silverstein J.W. (1985). The smallest eigenvalues of a large dimensional Wishart matrix. *Annals of Probability* **13**, 1364–1368.

Silverstein J.W. and Bai Z.D. (1995). On the empirical distribution of eigenvalues of a class of large dimensional random matrices. *Journal of Multivariate Analysis* **54**, 175–192.

Telatar E. (1999). Capacity of multi-antenna Gaussian channels. *European transactions on telecommunications* **10**, 585–595.

Terrell G.R. (1990). The maximal smoothing principle in density estimation. *Journal of the American Statistical Association* **85**, 470–477.

Terrell G.R. and Scott D.W. (1985). Oversmoothed nonparametric density estimates. *Journal of the American Statistical Association* **80**, 209–214.

Tibshirani R., Hastie T., Narasimhan B. and Chu G. (2003). Class prediction by nearest shrunken centroids, with applications to dna microarrays. *Statistical Science* **18**, 104–117.

Tracy C.A. and Widom H. (1996). On orthogonal and symplectic matrix ensembles. *Communications in Mathematical Physics* **177**, 727–754.

Tukey J.W. and Tukey P.A. (1990). *Strips Displaying Empirical Distributions: I. Textured Dot Strips*. Bellcore: Technical Memorandum.

Tukey P.A. and Tukey J.W. (1981). Graphical display of data sets in 3 or more dimensions. In: *Interpreting Multivariate Data*, edited by V. Barnett, pp. 189–257. John Wiley and Sons Ltd.

Vapnik V. and Chapelle O. (2000). Bounds on error expectation for support vector machine. *Neural Computation* **12**, 2013–2036.

Vapnik V.N. (1998). *Statistical Learning Theory*. Wiley-Interscience.

Wahba G., Lin Y. and Zhang H. (2000). Generalized approximate cross validation for support vector machines, or, another way to look at margin-like quantities. In: *Advances in Large Margin Classifiers*, edited by A. Smola, P. Bartlett, B. Schölkopf and D. Schhrmans, pp. 297–309. MIT Press.

Yin Y.Q., Bai Z.D. and Krishnaiah P.R. (1988). On the limit of the largest eigenvalue of the large dimensional sample covariance matrix. *Probability Theory and Related Fields* **78**, 509–521.

Zhang H.H., Ahn J., Lin X. and Park C. (2005a). Gene selection using support vector machines with nonconvex penalty. *Bioinformatics* **22**.

Zhang H.H., Marron J.S. and Todd M.J. (2005b). Weighted distance weighted discrimination. In manuscript.

Zhu J., Hastie T., Rosset S. and Tibshirani R. (2003). 1-norm support vector machines. In: *Neural Information Processing Systems*, volume 16.