Classification on Manifolds

by
Suman K. Sen

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
2008

Approved by

Advisor: Dr. James S. Marron

Reader: Dr. Douglas G. Kelly

Reader: Dr. Mark Foskey

Reader: Dr. Martin A. Styner

Reader: Dr. Yufeng Liu

© 2008
Suman Kumar Sen
ALL RIGHTS RESERVED

ABSTRACT

SUMAN KUMAR SEN: Classification on Manifolds
(Under the direction of Dr. James S. Marron)

This dissertation studies classification on smooth manifolds and the behavior of High Dimensional Low Sample Size (HDLSS) data as the dimension increases. In modern image analysis, statistical shape analysis plays an important role in understanding several diseases. One of the ways to represent three dimensional shapes is the *medial representation*, the parameters of which lie on a smooth manifold, and not in the usual d-dimensional Euclidean space. Existing classification methods like Support Vector Machine (SVM) and Distance Weighted Discrimination (DWD) do not naturally handle data lying on manifolds. We present a general framework of classification for data lying on manifolds and then extend SVM and DWD as special cases. The approach adopted here is to find *control points* on the manifold which represent the different classes of data and then define the classifier as a function of the distances (geodesic distances on the manifold) of individual points from the control points. Next, using a deterministic behavior of Euclidean HDLSS data, we show that the generalized version of SVM behaves asymptotically like the Mean Difference method as the dimension increases. Lastly, we consider the manifold $(S^2)^d$, and show that under some conditions, data lying on such a manifold has a deterministic geometric structure similar to Euclidean HDLSS data, as the dimension (number of components d in $(S^2)^d$) increases. Then we show that the generalized version of SVM behaves like the Geodesic Mean Difference (extension of the Mean Difference method to manifold data) under the deterministic geometric structure.

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my advisor, Professor J. S. Marron for his guidance, motivation and support. It is indeed a privilege to be his student.

I would also like to thank my committee members: Dr. Douglas G. Kelly, Dr. Mark Foskey, Dr. Martin A. Styner and Dr. Yufeng Liu for their suggestion and comments. Special thanks to Dr. Mark Foskey for making so much effort to answer my questions and providing valuable insights. He was always there to help. I thank Dr. Martin A. Styner for providing me with the data and his help to interpret the results.

I thank the MIDAG community for providing an enriching research environment, and Dr. Stephen Pizer for his extraordinary leadership. Special thanks to Dr. S. Joshi for helping me to get started with this research project. Without the gracious help of Ja-Yeon, Josh, Rohit, Ipek and Qiong, this document could not have been completed. I thank the professors and my colleagues at the Department of Statistics and Operations Research: they made my experience at UNC fruitful and enjoyable.

I express my deepest gratitude to my parents R. K. Sen and Anuradha Sen for their unconditional love, support and belief in me. I must mention my younger brother, Sumit: he is an epitome of discipline and hard work. I have learnt a few things from him.

No words are enough to thank my wife Paramita for being such a wonderful companion over the period of last nine years. Thank you for your unrelenting support and boundless love, not to mention the constructive inputs as an expert statistician. You are my strength.

CONTENTS

1 Introduction **1**

1.1 Motivation 1

1.2 Overview of Chapters 3

1.3 The Medial Locus and M-reps 5

1.3.1 Riemannian metric, Geodesic curve, Exponential and Log maps 7

1.3.1.1 Exponential and Log maps for S^2 7

1.3.1.2 Exponential and Log maps for M-reps 8

1.4 Statistical Methods on M-reps and on General Manifolds 9

2 Overview of Classification **12**

2.1 The Problem of Classification 12

2.2 Popular Methods of Classification 12

2.3 Value Added by Working on Manifolds 14

2.3.1 Importance of Geodesic Distance on Manifolds 14

2.3.2 Choice of base point for Euclidean Classification on the Tangent Plane 17

2.3.3 Validity and Interpretability of Projections 18

3 Classification on Manifolds **20**

3.1 Control Points and the General Classification Rule 20

3.1.1 The Implied Separating Surface and Direction of Separation . 21

3.1.2 Choice of Control Points 24
3.2 The Geodesic Mean Difference (GMD) Method 24
3.3 Support Vector Machine on Manifolds 25
3.3.1 A Brief Overview of Support Vector Machine (SVM) 25
3.3.2 Iterative Tangent Plane SVM (ITanSVM) 26
3.3.3 The Manifold SVM (MSVM) method 28
3.3.3.1 A Gradient Descent Approach to the MSVM Objective Function 31
3.3.4 Results 33
3.3.4.1 Application To Hippocampi Data 34
3.3.4.2 Application to Generated Ellipsoid Data 38
3.4 DWD on Manifolds 40
3.4.1 A Brief Overview of Distance Weighted Discrimination (DWD) 41
3.4.2 Iterative Tangent Plane DWD (ITanDWD) 42
3.4.3 The Manifold DWD (MDWD) method 42
3.4.4 Results 46
3.4.4.1 Application To Hippocampi Data 46
3.4.4.2 Application To Generated Ellipsoid Data 47
3.4.4.3 Discussion 48
3.5 MSVM with Enhanced Robustness 49
3.5.1 Shrinking the Control Points towards the Means 50
3.5.2 Results 51
3.5.2.1 Training and Cross Validation Errors 51
3.5.2.2 Projections on Direction of Separation 52
3.5.2.3 Sampling Variation 53
3.6 Summary 55

4 Asymptotics of HDLSS Manifold Data **57**

4.1 Geometric Representation of Euclidean HDLSS Data 57

4.1.1 Behavior of MSVM under Euclidean HDLSS Geometric Structure 60

4.1.2 Asymptotic Behavior of MSVM for Euclidean Data 68

4.2 Geometric Representation of Manifold HDLSS Data 73

4.2.1 Asymptotic Behavior of MSVM for Manifold Data 86

4.3 Summary 89

4.4 Technical Details 90

5 Discussion and Future Work **100**

5.1 Implementing MDWD 100

5.2 Role of the Parameter k in MSVM 100

5.3 Asymptotic Behavior of Manifold Data under Milder Conditions . . . 101

5.4 Application to DT-MRI 102

5.5 Generalizing Proposed Methods to Multiclass 102

Bibliography **103**

LIST OF FIGURES

1.1 A medial atom and boundary surface of hippocampus 6

1.2 The Riemannian exponential map 8

2.1 Toy data on a cylinder showing the importance of geodesics 16

2.2 Toy data on a sphere illustrating the effect of base point of tangency . 17

2.3 Invalid projection of m-rep illustrating the importance of working on the manifold 18

3.1 Control points on a sphere and their respective separating boundaries 22

3.2 Toy data showing the SVM hyperplane 26

3.3 The m-rep sheet of a hippocampus and the surface rendering 34

3.4 Performance of GMD, ITanSVM, TSVM and MSVM for Hippocampus data 35

3.5 Distance between control points against tuning parameter λ 36

3.6 Change captured by GMD, ITanSVM, TSVM, MSVM for Hippocampus data 37

3.7 The m-rep sheet and surface rendering of a deformed ellipsoid 38

3.8 Performance of GMD, ITanSVM, TSVM and MSVM for Ellipsoid data 39

3.9 Change captured by ITanSVM, TSVM, MSVM for Ellipsoid data . . 40

3.10 Toy data showing the DWD hyperplane 42

3.11 Figure showing the discontinuous nature of the MDWD objective function as a function of the step size along the negative gradient direction. 45

3.12 Comparison of TSVM, ITanSVM, TDWD, ITanDWD for Hippocampus data 46

3.13 Change captured by TSVM, ITanSVM, TDWD, ITanDWD for Hippocampus data 47
3.14 Comparison of TSVM, ITanSVM, TDWD, ITanDWD for Ellipsoid data 48
3.15 Change captured by TSVM, ITanSVM, TDWD, ITanDWD for Ellipsoid data 49
3.16 Performance of MSVM$_\nu$'s for Hippocampi data 52
3.17 Performance of MSVM$_\nu$'s for Ellipsoid data 52
3.18 Change captured by the MSVM$_\nu$ directions for Hippocampus data . . 53
3.19 Change captured by the MSVM$_\nu$ directions for Ellipsoid data 54
3.20 Comparison of the sampling variation of MSVM$_\nu$'s for Hippocampi data 54
3.21 Comparison of the sampling variation of MSVM$_\nu$'s for Ellipsoid data 55

4.1 Toy data in $\Re^2$ illustrating the deterministic structure in Euclidean HDLSS data 62
4.2 Toy data in $\Re^3$ illustrating the deterministic structure in Euclidean HDLSS data 65

CHAPTER 1

Introduction

1.1 Motivation

Statistical shape analysis is important in but not limited to understanding and diagnosing a number of challenging diseases. For example, brain disorders like autism and schizophrenia are often accompanied by structural changes. By detecting the shape changes, statistical shape analysis can help in diagnosing these diseases. One of the many ways to represent anatomical shape models is a medial representation or M-rep. The medial locus, which is a means of representing the "middle" of a geometric object, was first introduced by Blum (1967). Its treatment for 3D objects is given by Nackman and Pizer (1985). The medial atom represents a sampled place in the medial locus. Atoms form the building blocks for m-reps. In particular, the m-rep sheet can be thought of as representing a continuous branch of medial atoms. For details of discrete and continuous m-reps, see Pizer *et al.* (1999) and Yushkevich *et al.* (2003) respectively. The M-rep approach has proven to be useful in describing various aspects of shape, in capturing important summaries of the object's interior and boundary, and in providing relationships between neighboring objects. A major advantage of the m-reps approach to object representation over competitors is that it allows superior correspondence of features across a population of objects, which is critical to statistical analysis.

The elements of m-rep space are most naturally understood as lying in a curved

manifold, and not in the usual Euclidean space. We are interested in classification of data which lie on this curved m-rep space. A major contribution of this work is to use the geometric information inherent to the manifold. This enables the capture of a wide range of nonlinear shape variability including local thickness, twisting and widening of the objects. Principal geodesic analysis, the nonlinear analog of principal component analysis in this type of manifold setting, was developed using the geometry that can be derived from the Riemannian metric, including geodesic curves and distances (see Fletcher *et al.*, 2003, 2004) . Classification methods like Fisher Linear Discrimination (Fisher, 1936), Support Vector Machines (see Vapnik *et al.*, 1996; Burges, 1998), Distance Weighted Discrimination (Marron *et al.*, 2004) were designed for data which are vectors in Euclidean space and do not deal extensively with data that are parameterized by elements in curved manifolds. See Duda *et al.* (2001) and Hastie *et al.* (2001) for an overview of common existing classification methods. The challenge addressed here is to develop classification methods, or extend the existing methods so that they can handle data in curved manifolds. The notion of separating hyperplane, fundamental to many Euclidean classifiers, is challenging to even define in manifolds. The approach adopted here is to find *control points* on the manifold which represent the different classes of data and then define the classifier as a function of the distances (geodesic distances on the manifold) of individual points from the control points. We thus bypass the problem of explicitly finding separating boundaries on the manifold. The control points chosen will be those which optimize some objective function and the performance of several reasonable objective functions will be investigated and compared.

This approach will enable us not only to use the method on our motivating example of m-rep data (Sen *et al.*, 2008), but it is also applicable for Diffusion Tensor Magnetic Resonance Imaging (DT-MRI), and several other sciences like human movement, mechanical engineering, robotics, computer vision and molecular biology where non-

Euclidean data often appear.

Data sets with more variables (i.e., attributes or entries in the data vector) than observations are now important in many fields. This type of data is called High Dimension Low Sample Size (HDLSS) data. 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 typically less than several hundreds. Data from medical imaging, and from text recognition also often have a much larger dimension d than the sample size n. In our motivating example of m-reps, we have the HDLSS situation too, but the entries in each dimension are not Euclidean. As pointed out earlier, they lie on a smooth manifold. Due to the limitations of human visual perception beyond three dimensions, the behavior of HDLSS data is often counter-intuitive (Hall *et al.*, 2005; Donoho and Tanner, 2005), even in the Euclidean case. Ge and Simpson (1998) provided a framework for evaluating dimensional asymptotic properties of classification methods such as the Mean Difference. For the Euclidean case, Hall *et al.* (2005) have studied some deterministic behavior of the data and used the observations to analyze the asymptotic properties (as $d \to \infty$) of classification methods such as Mean Difference, SVM, DWD and One-Nearest-Neighbor. As a part of this dissertation, we have studied some geometric properties of HDLSS manifold data and used them to analyze the asymptotic behavior of one of our proposed methods, which is an extension of SVM for manifold data.

1.2 Overview of Chapters

This dissertation has been organized into chapters as follows:

Section 1.3 gives an overview of the medial representation. Section 1.4 gives an overview of different statistical methods for manifold data and m-reps.

Section 2.1 gives an overview of the problem of classification while Section 2.2

contains a literature review of popular classification methods like Mean Difference, Fisher Linear Discrimination, Support Vector Machines, Distance Weighted Discrimination. Section 2.3 illustrates why these methods are not always ideal for data on manifolds.

Chapter 3 presents the key ideas of our methods. Section 3.1 introduces the fundamental idea of control points, and based on it proposes the general classification rule. Section 3.1.2 illustrates the importance of a reasonable choice of control points. Section 3.3 gives a brief overview of SVM, and generalizes it to manifold data. It also provides a solving algorithm to the resulting optimization problem. Section 3.4 gives a brief overview of DWD, and develops an optimization problem which generalizes DWD to work with manifold data.

Chapter 4 studies some geometric properties of HDLSS data. Section 4.1 briefly discusses some deterministic properties of Euclidean HDLSS data and studies the asymptotic behavior (as dimension $d \to \infty$) of one of our developed methods (manifold SVM, also called MSVM) under such conditions. Section 4.2 studies conditions under which there is a deterministic structure in HDLSS manifold data. This deterministic structure is then used to study some properties of the MSVM method.

Chapter 5 discusses some avenues for future research, involving unresolved questions and possible applications of the developed methods in new areas.

The next section gives an overview of the medial locus and some of its mathematical properties. It describes m-reps and the deformable models approach based on them. The deformable m-reps approach to image segmentation is described by Pizer *et al.* (2003). A fine overview of medial techniques that goes beyond the material covered in this section can be found in the Ph.D. dissertation of Yushkevich (2003) and the book by Siddiqi and Pizer (2007).

1.3 The Medial Locus and M-reps

The medial locus is a means of representing the "middle" or "skeleton" of a geometric object. Such representations have found wide use in computer vision, image analysis, graphics, and computer aided design (Bloomenthal and Shoemake (1991); Storti *et al.* (1997)). Psychophysical and neurophysiological studies have shown evidence that medial relationships play an important role in the human visual system (Leyton (1992); Lee *et al.* (1995)). The medial locus was first proposed by Blum (1967), and its properties were later studied in 2D by Blum and Nagel (1978) and in 3D by Nackman and Pizer (1985). The definition of the medial locus of a set $A \in \Re^n$ is based on the concept of a maximal inscribed ball.

The medial representation is based on the medial axis of Blum (1967). In this framework, a geometric object is represented as a set of connected continuous medial manifolds. For three-dimensional (3-D) objects, these medial manifolds are formed by the centers of all spheres that are interior to the object and tangent to the object's boundary at two or more points. The medial description is defined by the centers of the inscribed spheres and by the associated vectors, called *spokes*, from the sphere centers to the two respective tangent points on the object boundary. Each continuous segment of the medial manifold represents a *medial figure*.

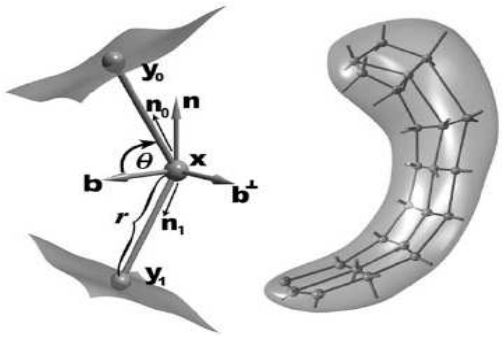


Figure 1.1: *Medial atom with a cross section of the boundary surface it implies (left). An m-rep model of a hippocampus and its boundary surface (right).*

The medial manifold is sampled over an approximately regular lattice and the elements of this lattice are called *medial atoms*. A medial atom (Fig. 1.1) is defined as a 4-tuple $m = \{x, r, n_0, n_1\}$, consisting of: $x \in \Re^3$, the center of the inscribed sphere; $r \in \Re^+$, the local width defined as the common spoke length; $n_o, n_1 \in S^2$, the two unit spoke directions (represented as points on S^2, the unit sphere in $\Re^3$). The medial atom implies two opposing boundary points, y_o, y_1, called *implied boundary* points, which are given by

$$y_0 = x + rn_0 \text{ and } y_1 = x + rn_1. \tag{1.1}$$

The surface normals at the implied boundary points y_0, y_1 are given by n_0, n_1, respectively.

A medial atom, as defined above, is represented as a point on the manifold $\mathcal{M}(1) = \Re^3 \times \Re^+ \times S^2 \times S^2$. Moreover, an m-rep model consisting of n medial atoms may be considered as a point on the manifold cartesian product $\mathcal{M}(n) = \prod_{i=1}^{n} \mathcal{M}(1)$. This space is a particular type of manifold known as a Riemannian symmetric space, which simplifies certain geometric computations, such as computing geodesic distances. We briefly review some of the concepts now. See Boothby (1986); Helgason (1978); Fletcher (2004), Fletcher *et al.* (2003, 2004) for more details. Pizer *et al.*

(1999) describes discrete m-reps. For details on continuous m-reps, see Yushkevich *et al.* (2003); Terriberry and Gerig (2006).

1.3.1 Riemannian metric, Geodesic curve, Exponential and Log maps

A *Riemannian metric* on a manifold M is a smoothly varying inner product $< \cdot, \cdot >$ on the tangent plane T_pM at each point $p \in M$. The norm of a vector $X \in T_pM$ is given by $||X|| =< X, X >^{(1/2)}$. The Riemannian distance between two points $x, y \in M$, denoted by $d(x, y)$, is defined as the minimum length over all possible smooth curves between x and y. A *geodesic curve* is a curve that locally minimizes the distance between points.

Given a tangent vector $X \in T_pM$, there exists an unique geodesic, $\gamma_X(t)$, with X as its initial velocity. The Riemannian *exponential map*, denoted by Exp_p, maps X to the point at time one along the geodesic γ_X. The exponential map preserves distances from the initial point, i.e., $d(p, \text{Exp}_p(X)) = ||X||$. In the neighborhood of zero, its inverse is defined and is called the Riemannian *log map*, denoted by Log_p. Thus, for a point y in the domain of Log_p, the geodesic distance between p and y is given by

$$d(p, y) = ||\text{Log}_p(y)|| \tag{1.2}$$

1.3.1.1 Exponential and Log maps for S^2

On the sphere S^2, the geodesics at the base point $p = (0, 0, 1)$ are great circles through p. If we consider the tangent vector $v = (v_1, v_1, 0) \in T_pS^2$ in the $x - y$ plane, the exponential map at p is given by

$$\text{Exp}_p(v) = (v_1.\frac{\sin||v||}{||v||}, v_2.\frac{\sin||v||}{||v||}, \cos||v||) \tag{1.3}$$

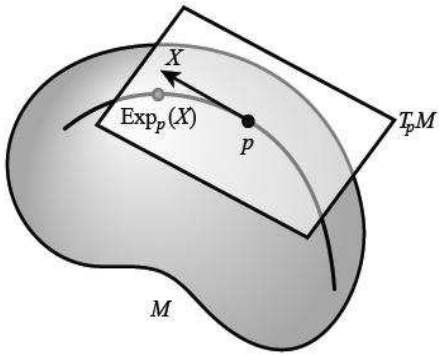


Figure 1.2: *The Riemannian exponential map at $p \in M$. $X \in T_pM$*

where $||v|| = \sqrt{v_1^2 + v_2^2}$.

The corresponding log map for a point $x = (x_1, x_2, x_3) \in S^2$ is given by

$$\mathrm{Log}_p(x) = (x_1.\frac{\theta}{\sin\theta}, x_2.\frac{\theta}{\sin\theta}) \tag{1.4}$$

where $\theta = \arccos(x_3)$ is the spherical distance from the base point p to the point x. Note that the antipodal point $-p$ is not in the domain of the log map.

1.3.1.2 Exponential and Log maps for M-reps

Recall, from first part of Section 1.3 that the medial atom $m = \{x, r, n_0, n_1\} \in \mathcal{M}(1) = \Re^3 \times \Re^+ \times S^2 \times S^2$ and a m-rep model consisting of n medial atoms may be considered as a point on the manifold $\mathcal{M}(n) = \prod_{i=1}^{n} \mathcal{M}(1)$. Let $p = (0, 1, p_0, p_1) \in \mathcal{M}(1)$ be the base medial atom, where $p_0 = p_1 = (0, 0, 1)$ are base points for the spherical components. Let us write a tangent vector $u \in T_p\mathcal{M}(1)$ as $u = (x, \rho, v_0, v_1)$, where $x \in \Re^3$ is the positional tangent component, $\rho \in \Re$ is the radius tangent component, and $v_0, v_1 \in \Re^2$ are the spherical tangent component. Then, for $\mathcal{M}(1)$ we have

$$\mathrm{Exp}_p(u) = (x, e^{\rho}, \mathrm{Exp}_{p_0}(v_0), \mathrm{Exp}_{p_1}(v_1)) \tag{1.5}$$

where the Exp maps on the right-hand side are the spherical exponential maps given by equation (1.3). Likewise, the log map of $m = \{x, r, n_0, n_1\}$ is

$$\mathrm{Log}_p(m) = (x, \log r, \mathrm{Log}_{p_0}(n_0), \mathrm{Log}_{p_1}(n_1)) \tag{1.6}$$

where the Log maps on the right-hand side are the spherical log maps given by (1.4).

Finally, the exponential and log maps for the m-rep model space $\mathcal{M}(n)$ is the cartesian product of corresponding maps in $\mathcal{M}(1)$.

The norm for vector $u \in T_p\mathcal{M}(1)$ is

$$||u|| = (||x||^2 + \bar{r}^2(\rho^2 + ||v_1||^2 + ||v_2||^2))^{\frac{1}{2}} \tag{1.7}$$

and the geodesic distance between two atoms $m_1, m_2 \in \mathcal{M}(1)$ is given by

$$d(m_1, m_2) = ||\mathrm{Log}_{m_1}(m_2)|| = ||\mathrm{Log}_{m_2}(m_1)|| \tag{1.8}$$

1.4 Statistical Methods on M-reps and on General Manifolds

The study of anatomical shape and its relation to biological growth and function dates back to the landmark work of Thompson (1942). While most work on the statistical analysis of shape has focused on linear methods, there has been some work on statistical methods for nonlinear geometric data. Hunt (1956) describes probability measures on Lie groups that satisfy the semigroup property under convolution. This leads to a natural definition of a Gaussian distribution on a Lie group as a fundamental solution to the heat equation. Wehn (1959, 1962) shows that such distributions satisfy a law of large numbers as in the Euclidean Gaussian case. Grenander (1963)'s book on probabilities on algebraic structures includes a review of these works on Gaussian dis-

tributions on Lie groups. Pennec (1999) defines Gaussian distributions on a manifold as probability densities that minimize information. Bhattacharya and Patrangenaru (2002) develop nonparametric statistics of the mean and dispersion values for data on a manifold. Mardia (1999) describes several methods for the statistical analysis of directional data, i.e., data on spheres and projective spaces. Kendall (1984) and also Mardia and Dryden (1989) have studied the probability distributions induced on shape space by independent identically distributed Gaussian distributions on the landmarks. Similar ideas in the theory of shape were independently developed by Bookstein (1978, 1986). Ruymgaart (1989) studied convergence of density estimators on spheres. Ruymgaart *et al.* (1992) gave a Rao-Cramer type inequality on Euclidean manifolds. Olsen (2003) and Swann and Olsen (2003) describe Lie group actions on shape space that result in nonlinear variations of shape. Klassen *et al.* (2004) develop an infinite-dimensional shape space representing smooth curves in the plane. Chikuse (2003) concentrates on the statistical analysis of two special manifolds, the Stiefel manifold and the Grassmann manifold, treated as statistical sample spaces consisting of matrices.

A standard technique for describing the variability of linear shape data is principal component analysis (PCA), a method whose origins go back to Pearson (1901) and Hotelling (1933). One of the earliest applications of PCA in functional data analysis was given by Rao (1958). Its use in shape analysis and deformable models was introduced by Cootes *et al.* (1993). Principal geodesic analysis, the nonlinear analog of principal component analysis in this type of manifold setting was developed using the geometry that can be derived from the Riemannian metric, including geodesic curves and distances (see Fletcher *et al.* (2003, 2004)). A popular approach to handling data in manifolds is "Kernel Embedding" (see Schölkopf and Smola (2002)) where the data are mapped to a higher dimensional feature space.

The next chapter provides a brief overview of the problem of classification. It also

discusses the challenges faced by the classical methods when applied to data lying on a manifold.

CHAPTER 2

Overview of Classification

The first section gives the general setup followed by a description of popular methods in Section 2.2. The last section explains why these methods fail in the context of data naturally understood to be lying on curved manifolds. See Duda *et al.* (2001) and Hastie *et al.* (2001) for an overview of common existing classification methods. Note that, in the literature, *classification* and *discrimination* are interchangeable terms. In our discussion we mostly refer to it as classification.

2.1 The Problem of Classification

Let X_i denote the attributes that describe the i^{th} individual and Y_i denote its group label, $i = 1, 2, \ldots, n$. X_i can be vector valued (in most cases it is a high dimensional vector) while Y_i is a scalar taking values in the set $\{1, 2, \ldots, K\}$ if there are K classes. In our discussion, we will always have only two classes. For some mathematical convenience, let the corresponding labels $Y \in \{-1, 1\}$.

Given a set of individuals (and their group labels), the goal of classification methods is to find a rule $f(x)$ that assigns a new individual to a group on the basis of its attributes X.

2.2 Popular Methods of Classification

There are quite a few popular methods of classification. Mean Difference is the simplest of them. It assigns a new observation to that class whose mean is closest

to it. It is the optimal classification rule when the data comes from distributions which only differ by their means and have common covariance matrix, which is the identity. A classical approach to improve the Mean Difference method was proposed by Fisher (1936), now called Fisher Linear Discrimination (FLD). It is the optimum rule when the two classes is assumed to have the same covariance matrix (but not limited to the identity). Since FLD approaches the problem by *sphering* the data it is frequently useless in many modern day applications, particularly High Dimension Low Sample Size (HDLSS) situations, where we cannot calculate the inverse of the covariance matrix.

The Support Vector Machine (SVM), proposed by Vapnik (1982, 1995) is a powerful classification method used in HDLSS situations. See Burges (1998) for a lucid overview. Marron *et al.* (2004) showed that SVM suffers from "data piling" at the "margin" and introduced a related method called Distance Weighted Discrimination (DWD). DWD avoids data piling and improves generalizability of the decision rule. Both SVM and DWD are "linear" classifiers in the sense that the rules are linear functions of the data vector. Geometrically, the separating surface they provide is a linear hyperplane and thus cannot separate data which need a nonlinear separating boundary. This problem is overcome by "kernel embedding". The data vectors are embedded in a higher dimensional space where linear methods, such as SVM can be much more effective. The book by Schölkopf and Smola (2002) gives a fine overview of the kernel methods. Readers can also visit *http://www.kernel-machines.org/publications.html* for a list of publications on kernel methods.

In this dissertation, we focus on the methods of SVM and DWD. Brief overviews of these methods are given in Sections 3.3.1 and 3.4.1 respectively.

2.3 Value Added by Working on Manifolds

Recall that our goal is to help image analysts and doctors understand how two groups of objects differ. For example, let us consider a study of shapes of hippocampi for groups of schizophrenic and normal individuals. There is a strong interest in knowing whether the occurrence of the disease is actually accompanied by a structural difference of the hippocampus. If there is an association, the next question is how these shapes change as we look at the direction of separation.

Some common approaches to handling data on manifolds are:

- Flatten, as for data on a cylinder (Section 2.3.1).
- Work in the tangent plane, with base as the overall geodesic mean of the data (Section 2.3.2).
- Treat data as points embedded in a higher dimensional Euclidean space (Section 2.3.3).

The drawbacks of these approaches are illustrated in the above mentioned subsections.

2.3.1 Importance of Geodesic Distance on Manifolds

Throughout our discussion, the geodesic distance will play a crucial role. Let us use an example to explain its importance.

Fig. 2.1(a) shows a cylinder with data points on its surface (different symbols denote different classes). Fig. 2.1(b) shows the same data set when the cylinder is flattened. The Mean Difference decision rule (given by the two shaded regions: blue and white) is calculated by using the usual Euclidean distance on the flattened two dimensional plane. This approach is not ideal. For example, it ignores the fact that

the point on the extreme left is actually close to the point to the extreme right when the geometry of the manifold (in this case, the cylinder) is considered. A better way to treat this data is to correctly account for the periodicity by repeating the flattened plane sideways (as shown in Fig. 2.1(c)) and consider the shortest possible distance between any pair of points while constructing a decision rule. This approach uses the Mean Difference rule based on the geodesic distance on the manifold and the corresponding decision rule is given by the shaded regions. The Mean Difference rule using Euclidean distance misclassifies a point (green star in the blue shaded region). The geodesic distance based Mean Difference classification rule (Fig. 2.1(c)) is able to find a separating surface (the boundary between the shaded regions) which properly separates the two classes. This illustrates the importance of using geodesic distances for classification on manifolds. The cylinder is a simple manifold ($\Re \times S^1$): even there geodesic distance makes a difference. This effect will usually be magnified for more complex manifolds. It is therefore recommended that geodesic distance be used when dealing with more complicated manifolds, such as $\mathcal{M}(n)$ (as defined in Section 1.3), where the M-reps live.

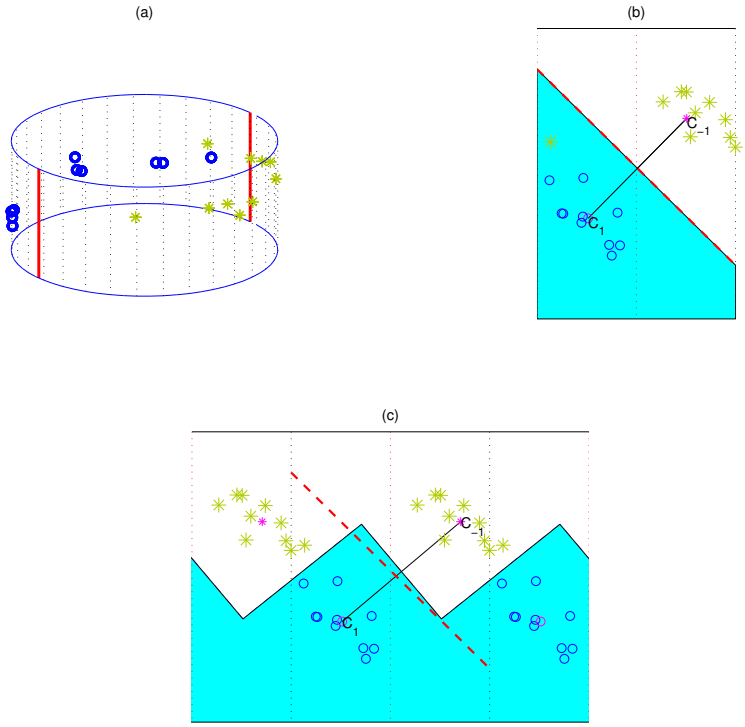


Figure 2.1: *(a): Toy data on the surface of a cylinder. Different colors (with symbols) denote different groups. (b): Data on the flattened cylinder. Shaded regions show the Mean Difference rule using Euclidean distance. A point (green star) is misclassified. (c): Tiled flattened planes capturing the structure (periodicity) of the manifold. Geodesic distance used to construct the Mean Difference rule which has no misclassified points. The red dotted line is the Mean Difference separating surface if Euclidean distance is used.* c_1 *and* c_{-1} *are means of the classes in each case.*

2.3.2 Choice of base point for Euclidean Classification on the Tangent Plane

As pointed out in the Sections 1.1 and 1.4, when data appear in a small neighborhood, some statistical analysis, such as finding means and Principal Component Analysis can be successfully implemented. This is done in the tangent plane by projecting the data from the manifold to the tangent plane with base point as the geodesic mean (see Fletcher *et al.*, 2003, 2004). This suggests implementing SVM and DWD in the tangent plane at the geodesic mean of the data. Fig 2.2 shows why this might not always be a good idea. In particular, this example shows that we can end up with a tangent plane where the data is not linearly separable while there is another tangent plane where separation is possible. Thus, the choice of the base point is a crucial issue.

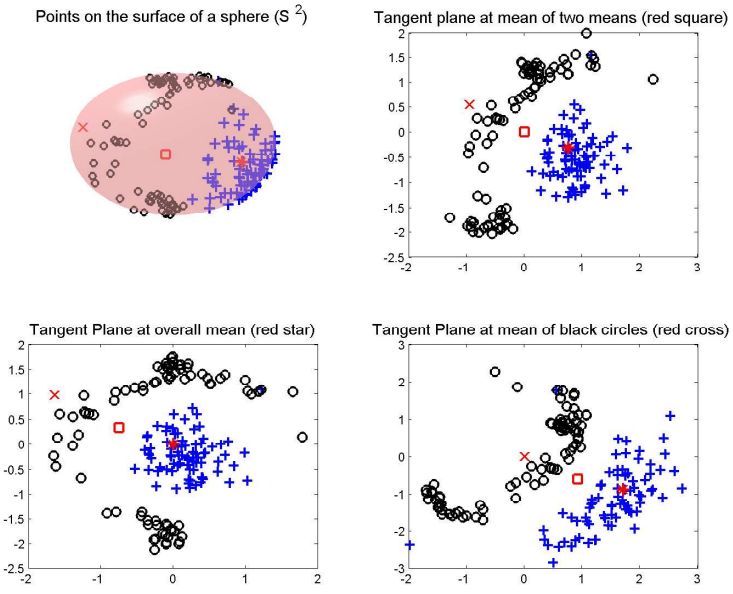


Figure 2.2: *Top left: Toy data on the surface of a sphere. Different colors (with symbols) denote different groups. Bottom Left: Tangent plane with base as overall geodesic mean of the entire data. The data are not linearly separable. Bottom right: Tangent plane at the geodesic mean of the black circles. The data can be separated linearly. Top Right: Tangent plane at the geodesic mean of the geodesic means of the two groups. The points are not as well-separable as in the bottom right panel.*

2.3.3 Validity and Interpretability of Projections

Another approach to handle manifold data is standard Euclidean SVM and DWD, treating the data as points embedded in the higher dimensional Euclidean space. For example, planar angles ($\theta \in (-\pi, \pi)$) can be considered as embedded in $\Re^2$ (as $(\sin(\theta), \cos(\theta))$), while naturally they are understood to be lying on a one-dimensional manifold (the unit circle in $\Re^2$). Similarly solid angles are embedded points in $\Re^3$ while naturally they are points restricted to the surface of the unit sphere (S^2) in $\Re^3$. A separating rule will be obtained by taking this approach, but geometrically the separating surface will not relate properly to the manifold. For example, in case of data on the surface of a sphere a separating plane cutting through the sphere will be obtained. This will probably not give a geodesic (in this case, a great circle), which is the analog of a separating hyperplane.

More importantly, when the original data is projected on to the separating direction, most of them will be somewhere inside the sphere and not the surface. These projections are not interpretable because they are not valid representations of shape objects. Recall, in this example, our data objects must lie on the surface of the sphere.

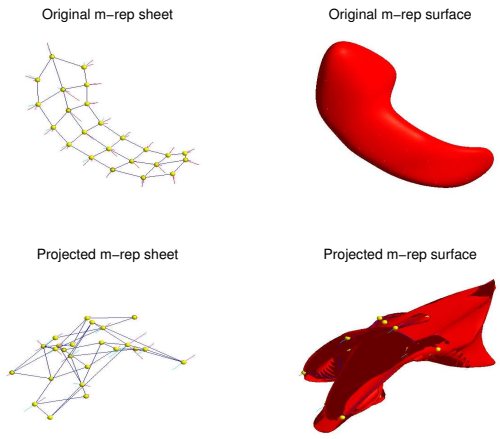


Figure 2.3: *Top left and right: Original m-rep sheet of a hippocampus and its surface rendering respectively. Bottom left: M-rep model projected on to the separating direction (obtained when data objects are considered as points in $\Re^{240}$). Since it is not in the manifold it is not a valid m-rep. Bottom right: Surface rendering of the m-rep sheet on the left. It is not an interpretable shape object.*

To emphasise this issue consider Figure 2.3. The upper figure is the medial representation of a human hippocampus. In our study we had two groups: 56 patients and 26 controls (see Styner *et al.* (2004)). Each of these models have 24 medial atoms, placed in a 8×3 lattice (see Fig. 2.3, top left). Therefore, each of these figures belong to $\mathcal{M}(24) = \{\Re^3 \times \Re^+ \times S^2 \times S^2\}^{24}$. But when standard SVM is implemented on the data we consider them as elements of $\{\Re^3 \times \Re \times \Re^3 \times \Re^3\}^{24} = \Re^{240}$. Naturally, when we project the data sets on to the separating directions, what we get back are also elements of $\Re^{240}$ and not in $\mathcal{M}(24)$. Therefore the projection is not a valid medial representation and thus not interpretable (Fig. 2.3, bottom panels). This motivates our approach to work on manifolds.

The next chapter provides a framework for classification on manifolds using geodesic distances. This framework is then used to extend methods like Mean Difference, SVM and DWD for manifold data.

CHAPTER 3

Classification on Manifolds

The main problem with classification on manifolds is that it is very difficult to derive analytical expressions for geodesics and separating surfaces. Recall, from Section 1.3, a geodesic is the local shortest path along the manifold and the distance between two points is obtained as the arc length of this geodesic. Our goal is to extend the idea of a separating hyperplane (which is the foundation of many Euclidean classification methods such as Mean Difference, FLD, SVM and DWD) to data lying on a manifold. A major challenge is to find an appropriate manifold analog of the separating hyperplane. Our solution is based on the idea of control points (and the geodesic distance of data from these control points), as described in the following section.

3.1 Control Points and the General Classification Rule

We think about *control points* as being representatives of the two classes. If we name the control points as c_1 and c_{-1}, then we propose the classification rule $f_{c_1,c_{-1}}(x)$ given by

$$f_{c_1,c_{-1}}(x) \;=\; d^2(c_{-1},x) - d^2(c_1,x), \tag{3.1}$$

where c_1, c_{-1}, and $x \in M$ and $d(\cdot,\cdot)$ is the geodesic distance metric defined on the manifold M. This rule assigns a new point x to class 1 if it is closer to c_1 than c_{-1}, and to class -1 otherwise. It is important to note here that the formulation also provides us with an implicitly defined separating surface and a direction of separation.

3.1.1 The Implied Separating Surface and Direction of Separation

The zero level set of $f_{c_1,c_2}(\cdot)$ is the analog of the separating hyperplane, while the geodesic joining c_1 and c_{-1} is the analog of the direction of separation. Thus, the separating surface is the set of points which is equidistant from c_1 and c_{-1}. If we denote it by $H(c_1, c_{-1})$, we can write,

$$\begin{aligned} H(c_1, c_{-1}) &= \{x \in M : f_{c_1,c_{-1}}(x) = 0\} \\ &= \{x \in M : d^2(c_1, x) = d^2(c_{-1}, x)\} \end{aligned} \tag{3.2}$$

In d-dimensional Euclidean space, $H(c_1, c_{-1})$ is a hyperplane of dimension $d-1$ that is the perpendicular bisector of the line segment joining c_1 and c_{-1} (see Lemma 3.1.1). Note that the Mean Difference method is a particular case of this rule, where the control points are the means of the respective classes. Similarly, the general control point classifier reduces to Fisher Linear Discrimination in Euclidean space by taking the control points as the means of the sphered data (using the pooled within class covariance estimate).

On the sphere (S^2), $H(c_1, c_{-1})$ is the great circle equidistant from c_1 and c_{-1} (see Fig. 3.1.) This shows that this approach provides us with an useful representation of a separating surface, that avoids the need to explicitly solve for it, which can be intractable as pointed out earlier.

A set of training points is to said to be *separable* by $H(c_1, c_{-1})$ if all the points

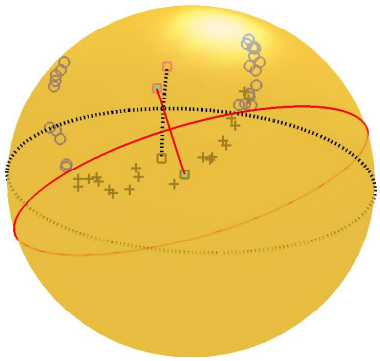


Figure 3.1: *Two pairs of control points showing their respective separating boundary and separating direction on the surface of the sphere. Different colors (with symbols) represent classes. The solid red surface (great circle) separates the data better than the dotted black surface.*

are classified correctly by $H(c_1, c_{-1})$. Mathematically, the training set T is said to be separable by $H(c_1, c_{-1})$, if, $\forall x_i \in T, i = 1, 2, \ldots, n$,

$$f_{c_1,c_{-1}}(x_i) \begin{cases} > 0 & \text{if } y_i = 1 \\ < 0 & \text{if } y_i = -1 \end{cases} \tag{3.3}$$

Lemma 3.1.1. *Consider the separating surface defined in equation (3.2). Let the data live in* $\Re^d$. *Then*

(1) $H(c_1, c_{-1})$ *is a* $d-1$ *dimensional hyperplane.*

(2) The level set of $f_{c_1,c_{-1}}(x) = k$ *is a hyperplane, parallel to* $H(c_1, c_{-1})$.

(3) The distance of any point x *from the separating surface* $H(c_1, c_{-1})$ *is given by*

$$d(x, H(c_1, c_{-1})) = \frac{|f_{c_1,c_{-1}}(x)|}{2d(c_1, c_{-1})} \tag{3.4}$$

Proof. Since the space in which the data live is Euclidean, the distance between any

two points $x, y \in \Re^d$ is given by

$$\begin{aligned} d(x,y) &= ||x-y|| \\ &= \sqrt{(x-y)^T(x-y)}. \end{aligned} \tag{3.5}$$

Therefore, we can write the following:

$$\begin{aligned} f_{c_1,c_{-1}}(x) &= d^2(c_{-1},x) - d^2(c_1,x) \\ &= (c_{-1}-x)^T(c_{-1}-x) - (c_1-x)^T(c_1-x) \\ &= w^Tx + b, \end{aligned} \tag{3.6}$$

where $w = 2(c_1 - c_{-1})$ and $b = (c_{-1}^T c_{-1} - c_1^T c_1)$. Thus, the equation of the level set of $f_{c_1,c_{-1}}(x) = k$, for any k can be written as

$$\begin{aligned} w^Tx + b &= k \\ \Rightarrow w^Tx + (b-k) &= 0 \end{aligned} \tag{3.7}$$

Note, that Equation (3.7) says that for all k, the level set is a $d-1$ dimensional hyperplane with common normal vector w and intercept $b-k$. This proves part (i), as we note that $H(c_1, c_{-1})$ is the level set for $k = 0$. Moreover, for any other $k \neq 0$, the normal to the resulting hyperplane is the same ($= w$). This proves part (ii).

For part (iii), we use the fact that the distance from any point z to a plane $w^Tx + b = 0$ is given by $\dfrac{|w^Tz + b|}{||w||}$. Therefore, using equation (3.6) we can write the distance of any point x from $H(c_1, c_{-1})$ as

$$d(x, H(c_1, c_{-1})) = \frac{|w^Tx + b|}{||w||}$$

$$
\begin{aligned}
&= \frac{|f_{c_1,c_{-1}}(x)|}{||2(c_1 - c_{-1})||}, \\
&= \left| \frac{f_{c_1,c_{-1}}(x)}{2d(c_1, c_{-1})} \right| \qquad (3.8)
\end{aligned}
$$

□

3.1.2 Choice of Control Points

Having set the framework for the general decision rule for manifolds the critical issue now is the choice of control points. For example, Fig. 3.1 shows that for the given set of data, the control points corresponding to the red solid separating boundary do a better job of classification than the pair corresponding to the black dotted boundary. So, the key to the construction of a good classification rule is to find the right pair of control points.

The rest of the chapter develops new methods for finding control points. The first approach, motivated by Mean Difference, chooses the control points as the geodesic means and is called the Geodesic Mean Difference (GMD) Method. Then we propose methods to extend SVM and DWD for manifold data.

3.2 The Geodesic Mean Difference (GMD) Method

This is motivated by the Mean Difference method in the Euclidean case. Here we replace the Euclidean means of the two classes by their geodesic means. This is a special case of the general classification rule (3.1) when c_1 and c_{-1} are the geodesic means of the two groups of data. The geodesic mean m of a set of observations $x_1, x_2, \ldots, x_n \in M$ is defined as

$$
m = \underset{p \in M}{\operatorname{argmin}} \sum_{i=1}^{n} d^2(p, x_i). \qquad (3.9)
$$

See Fletcher *et al.* (2003, 2004) and Pennec (1999) for more details. In Fig. 3.1, the black dotted great circle shows the GMD separating surface. The two square points (joined by dotted black curve) are the geodesic means of the two classes. Some points have been misclassified by this rule.

3.3 Support Vector Machine on Manifolds

This section generalizes SVM to the manifold case. Two new approaches have been developed here. In the following subsection, we review the standard SVM method.

3.3.1 A Brief Overview of Support Vector Machine (SVM)

SVM is a recently developed method which has been successfully implemented in a wide variety of applications involving classification. Here, we review this method in a simple set up. Let us first assume that the training data set is linearly separable, i.e., there is a linear classifier that can have zero training error. Consider a linear classifier $f(x) = w^T x + b$. The SVM first finds two hyperplane margins (over w and b) which are defined by $f(x) = \pm 1$, such that there are some observations on the margins and there are no observations between these two margins. The points on the margin are called "support vectors". The SVM finds w and b such that the distance between the margins (which is equal to $\frac{2}{||w||}$) is maximized. The hyperplane between the two margins: $f(x) = 0$ is the SVM discrimination hyperplane. Given w and b, the class label $+1$ is given to a new sample x_i, if $f(x_i) > 0$ and the class label -1 is given if $f(x_i) < 0$. The SVM optimization problem over w and b is given by:

$$\begin{aligned} &\text{minimize}_{w,b} \frac{||w||^2}{2} \\ &\text{subject to } y_i f(x_i) \geq 1; \quad i = 1, \ldots, n, \end{aligned} \tag{3.10}$$

where y_i represents the class membership of the i^{th} sample x_i in the training data set. The normalized direction vector of w represents the SVM direction. The constraints

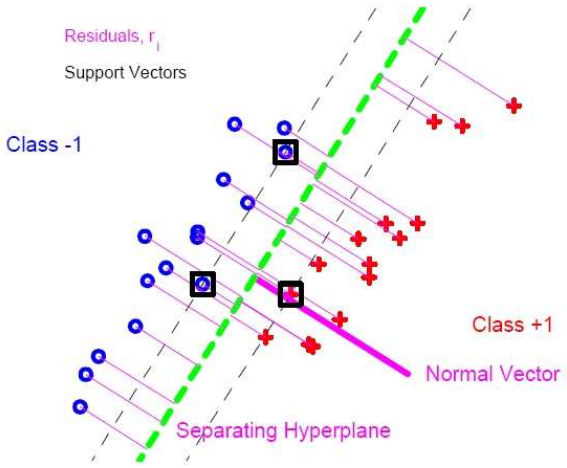


Figure 3.2: *SVM hyperplane (broken green line) to separate two classes, represented by crosses and pluses. The purple vector is the SVM direction of separation.*

$y_i f(x_i) \geq 1, \quad i = 1, \ldots, n$ indicate that the $f(x)$ must classify all the samples in the training data set correctly. The distance of a point x_i from the separating hyperplane is denoted as r_i.

Figure 3.2 shows the SVM separating hyperplane (green dashed line) for classifying a toy data set, with the two classes represented by blue circles and red pluses respectively. The support vectors are shown in black boxes. The distance r_i's of only these support vectors play a role in determining the separating hyperplane.

The next subsection extends SVM to manifold data by iteratively constructing tangent planes on the manifold and implementing Euclidean SVM on the tangent planes. We call this method Iterative Tangent Plane SVM (ITanSVM).

3.3.2 Iterative Tangent Plane SVM (ITanSVM)

In this section, we propose an extension of SVM in Euclidean space to manifold data. A common approach is to implement SVM in the tangent plane at the overall mean of the data. In Section 2.3.2, we have discussed possible drawbacks of this approach which arises out of taking the point of tangency at a predetermined point (the geodesic mean). Thus, when classification is done on the tangent plane, the choice of the base point is a crucial issue. This motivates the need to find the base

point that is determined by the separability of the data in the tangent plane. In particular, we develop an iterative approach with changing point of tangency.

We start with the overall geodesic mean as the initial base point and implement Euclidean SVM in that tangent plane. Given the SVM separating hyperplane, we find out the pair of points (on the tangent plane) which determine that hyperplane and is closest to the present pair of control points (the geodesic means of the two classes, mapped to the tangent plane). The next point of tangency is taken to be the geodesic mean of the new pair of control points (after being mapped back to the manifold). We repeat these steps until convergence. The detailed algorithm is given below:

1. Let c_1^0 = mean of the data in class 1 and c_{-1}^0=mean of the data in class 2. Let $b_0 = \text{mean}(c_1^0, c_{-1}^0)$. The superscript of zero means it is the present solution.

2. Compute the tangent plane $T_{b_0}M$ at b_0 and find the separating hyperplane $w'x + b = 0$ by doing linear SVM on $T_{b_0}M$.

3. Given (w, b), find $Lc_1^1, Lc_{-1}^1 \in T_{b_0}M$ that minimize the sum of the squares of their respective distances to $\text{Log}_{b_0}(c_1^0)$ and $\text{Log}_{b_0}(c_{-1}^0)$, subject to the constraint that $w'x + b$ is the perpendicular bisector of the the line segment joining Lc_1^1 and Lc_{-1}^1.

4. If $\{d^2(\text{Log}_{b_0}(c_1^0), Lc_1) + d^2(\text{Log}_{b_0}(c_{-1}^0), Lc_{-1})\}$ is very small then stop. Otherwise go to the next step.

5. Set $c_1^0 = \text{Exp}_{b_0}(Lc_1^1)$ and $c_{-1}^0 = \text{Exp}_{b_0}(Lc_{-1}^1)$ and then compute $b_0 = \text{mean}(c_1^0, c_{-1}^0)$. Go to step 2.

We call this method the Iterative Tangent Plane SVM or ITanSVM. In Fig. 3.1 the solid red great circle shows the ITanSVM separating surface obtained after one iteration. The square points (joined by red solid curve) are the ITanSVM control

points after the first iteration. In this example, ITanSVM solution does a better job of classification than GMD (given by the dotted black great circle).

3.3.3 The Manifold SVM (MSVM) method

An unappealing feature of ITanSVM is the continual approximation of the data by projections on to the tangent plane. MSVM appears to be the first approach where all calculations are done on the manifold. MSVM determines a pair of control points that maximizes the minimum distance to the separating boundary. While the SVM criterion has many interpretations, it is the maximum margin idea that generalizes most naturally to manifolds where some Euclidean notions such as distance, are much more readily available than other (e.g., inner product). The mathematical formulation of the method and an algorithm for solving the resulting optimization problem are given below.

As given in equation (3.1), the decision function $f_{c_1,c_{-1}}(x)$ is

$$f_{c_1,c_{-1}}(x) = d^2(c_{-1}, x) - d^2(c_1, x)$$

The zero level set of $f_{c_1,c_{-1}}(\cdot)$ defines the separating boundary $H(c_1, c_{-1})$ for a given pair (c_1, c_{-1}). Also, let $\widehat{X}_{(c_1,c_{-1})}$ denote the set (to handle possible ties) of training points which are nearest to $H(c_1, c_{-1})$. We would like to solve for some $\tilde{c_1}$ and $\tilde{c}_{-1}$ such that

$$(\tilde{c_1}, \tilde{c}_{-1}) = \underset{c_1,c_{-1}\in M}{\operatorname{argmax}} \min_{i=1\ldots n} d(x_i, H(c_1, c_{-1})) \tag{3.11}$$

In other words, we want to maximize the minimum distance of the training points from the separating boundary. Recall, from Chapter 2, that as in the classical SVM literature, we denote y_i to be the class label taking values -1 and 1.

By Lemma 3.1.1, in Euclidean space, note that the distance of any point x from

the separating boundary $H(c_1, c_{-1})$ is

$$d(x, H(c_1, c_{-1})) \;=\; \left| \frac{f_{c_1,c_{-1}}(x)}{2d(c_1, c_{-1})} \right| \tag{3.12}$$

Therefore, for a separable training set (see Section 3.1), using (3.3) and (3.12), we can write the distance from the training points to the separating surface as

$$d(x, H(c_1, c_{-1})) \;=\; \frac{y f_{c_1,c_{-1}}(x)}{2d(c_1, c_{-1})}, \tag{3.13}$$

where $y = \pm 1$ is the class label for x. Relation (3.13) will be used as an approximation that is reasonable for data on manifolds lying in a small (convex) neighborhood as this is directly computable for manifold data. Then, using (3.13) in (3.11) we would like to solve for some $\tilde{c_1}$ and $\tilde{c}_{-1}$ such that

$$(\tilde{c_1}, \tilde{c}_{-1}) = \underset{c_1,c_{-1} \in M}{\operatorname{argmax}} \min_{i=1\ldots n} \left\{ \frac{y_i f_{c_1,c_{-1}}(x_i)}{2d(c_1, c_{-1})} \right\} \tag{3.14}$$

It is important to note that the solution of $(\tilde{c_1}, \tilde{c}_{-1})$ in (3.14) is not unique. In fact, in the d-dimensional Euclidean case there is a $(d-1)$-dimensional space of solutions. Therefore, in order to make the search space for $(\tilde{c_1}, \tilde{c}_{-1})$ smaller we propose to find $(\tilde{c_1}, \tilde{c}_{-1})$ as follows:

$$(\tilde{c_1}, \tilde{c}_{-1}) = \underset{(c_1,c_{-1}) \in C_k}{\operatorname{argmax}} \min_{i=1\ldots n} \left\{ \frac{y_i f_{c_1,c_{-1}}(x_i)}{2d(c_1, c_{-1})} \right\} \tag{3.15}$$

where, for a given $k > 0$,

$$C_k = \{(c_1, c_{-1}) : \hat{y}_{(c_1,c_{-1})} f_{c_1,c_{-1}}(\hat{x}_{(c_1,c_{-1})}) = k\} \tag{3.16}$$

and,

$$\hat{x}_{(c_1,c_{-1})} = \underset{x \in \widehat{X}_{(c_1,c_{-1})}}{\operatorname{argmin}} \; f_{c_1,c_{-1}}(x) \tag{3.17}$$

and $\hat{y}_{(c_1,c_{-1})}$ is the class label of $\hat{x}_{(c_1,c_{-1})}$.

Therefore, using (3.15) - (3.17), we have

$$(\tilde{c_1}, \tilde{c}_{-1}) = \underset{(c_1,c_{-1})}{\operatorname{argmax}} \left\{ \frac{k}{2d(c_1, c_{-1})} \right\} \tag{3.18}$$

Now, recall that $\hat{x}_{(c_1,c_{-1})}$ is one of the training points closest to $H(c_1, c_{-1})$. This means, no other training point should be closer to $H(c_1, c_{-1})$ than $\hat{x}_{(c_1,c_{-1})}$. This presents us with a set of constraints which should be considered while solving for $(\tilde{c_1}, \tilde{c}_{-1})$ in (3.18). The constraints are given as follows:
∀i=1,2,. . . , n,

$$\begin{aligned} d(x_i, H(c_1, c_{-1})) &\geq d(\hat{x}_{(c_1,c_{-1})}, H(c_1, c_{-1})) \\ \Rightarrow \frac{y_i f(x_i)}{2d(c_1, c_{-1})} &\geq \frac{k}{2d(c_1, c_{-1})} \\ \Rightarrow y_i f(x_i) &\geq k \\ \Rightarrow k - y_i\{d^2(x_i, c_{-1}) - d^2(x_i, c_1)\} &\leq 0 \qquad (3.19) \\ \Rightarrow h_i &\leq 0 \qquad (3.20) \end{aligned}$$

where $h_i = k - y_i\{d^2(x_i, c_{-1}) - d^2(x_i, c_1)\}$.

Combining the constraints ($h_i < 0, \forall\ i = 1, 2, \ldots, n$) with (3.18), the optimization problem becomes

$$\max_{c_1,c_{-1}} \frac{1}{d(c_1, c_{-1})} \text{ s.t. } h_i \leq 0$$

or,

$$\min_{c_1,c_{-1}} d(c_1, c_{-1}) \text{ s.t. } h_i \leq 0 \tag{3.21}$$

Rather than solving the constrained optimization problem in (3.21), we consider the penalized minimization problem as defined below:

Minimize

$$g_\lambda(c_1, c_{-1}) = d^2(c_1, c_{-1}) + \frac{\lambda}{n} \sum_{i=1}^{n} (h_i)_+$$

or,

$$g_\lambda(c_1, c_{-1}) = d^2(c_1, c_{-1}) + \frac{\lambda}{n}\sum_{i=1}^{n}\Big[k - y_i\{d^2(x_i, c_{-1}) - d^2(x_i, c_1)\}\Big]_+ \qquad (3.22)$$

where λ is the penalty for violating the constraints given by (3.20).

Equation (3.22) gives the function to be minimized in the separable case. In the non-separable case, the form of the function $g_\lambda(c_1, c_{-1})$ to be minimized remains the same. What changes is the definition of $\widehat{X}_{(c_1,c_{-1})}$, and hence, the definition of $\hat{x}_{(c_1,c_{-1})}$, $\hat{y}_{(c_1,c_{-1})}$ and k. $\widehat{X}_{(c_1,c_{-1})}$ is now defined as the set of *correctly classified* training points which are closest to $H(c_1, c_{-1})$ (in the separable case, the term *correctly classified* was not necessary since all the training points are correctly classified by $H(c_1, c_{-1})$. In the non-separable case there can be a misclassified point which is closest to $H(c_1, c_{-1})$ among all training points, but it is not an element of $\widehat{X}_{(c_1,c_{-1})}$). Note that the second term in (3.22) not only penalizes misclassification, but also penalizes cases where training points come too close to the separating boundary.

The exact solution which minimizes the objective function in Eq. (3.22) will be referred to as the *"idealized"* MSVM solution. The following subsection assumes that the data lies in a small convex neighborhood and then proposes a gradient descent approach to minimize the objective function (Eq. 3.22). The resulting solution will be referred to as the MSVM solution.

3.3.3.1 A Gradient Descent Approach to the MSVM Objective Function

Here we propose and develop an algorithm to minimize the objective function given by (3.22).

Given $c_1 = m_1$ and $c_{-1} = m_2$, let us denote by $\Delta_1(m_1, m_2)$ and $\Delta_2(m_1, m_2)$, the gradient of $g_\lambda(c_1, c_{-1})$ with respect to c_1 and c_{-1} respectively. It is assumed that the data lie in a small convex neighborhood. Therefore, a negative gradient approach is followed, which has been successfully used in finding geodesic means on manifolds (see Fletcher *et al.*, 2003, 2004), using arguments in Karcher (1977) and Pennec (1999).

The gradients of the objective function are given by

$$\begin{aligned}\Delta_1(m_1, m_2) &= \frac{\partial g_\lambda}{\partial m_1} \\ &= -2\text{Log}_{m_1}(m_2) - 2\frac{\lambda}{n}\sum_{(i:h_i\geq 0)}(y_i\text{Log}_{m_1}(x_i))\end{aligned} \tag{3.23}$$

and,

$$\begin{aligned}\Delta_2(m_1, m_2) &= \frac{\partial g_\lambda}{\partial m_2} \\ &= -2\text{Log}_{m_2}(m_1) + 2\frac{\lambda}{n}\sum_{(i:h_i\geq 0)}(y_i\text{Log}_{m_2}(x_i))\end{aligned}$$

Algorithm:

Start with initial value of $c_1 = c_1^0$ and $c_{-1} = c_{-1}^0$

Set $\Delta = 1, i = 0$

While $\Delta > \varepsilon(small)$

{

$i = i + 1$

Calculate $\Delta_1(c_1^{i-1}, c_{-1}^{i-1})$

UPDATE: $c_1^i = \text{Exp}_{c_1^{i-1}}(-t\Delta_1(c_1^{i-1}, c_{-1}^{i-1})), \qquad t = \text{step size} \in (0, 1)$

Calculate $\Delta_2(c_1^i, c_{-1}^{i-1})$

UPDATE: $c_{-1}^i = \text{Exp}_{c_{-1}^i}(-t\Delta_2(c_1^i, c_{-1}^{i-1})), \qquad t = \text{step size} \in (0, 1)$

$\Delta = ||\Delta_1(c_1^i, c_{-1}^i)|| + ||\Delta_2(c_1^i, c_{-1}^i)||$

}

In the next section we will compare the results of MSVM and ITanSVM along with the Geodesic Mean Difference (GMD) method.

3.3.4 Results

In the previous sections, three classification methods, GMD (Section 3.2), ITanSVM (Section 3.3.2) and MSVM (Section 3.3.3) were proposed. In this section, we compare the performance of these methods along with the method of Euclidean SVM on a single tangent plane (with the overall geodesic mean as base point). We will call this method *TSVM*.

Associated with every classification rule are two types of errors, the training error and the cross-validation error (also called test error). Training error is the proportion of the training data (data used to find the rule) that is misclassified by the rule. The cross-validation error is the proportion of the test data (data believed to be behaving like the training data but not used to find the rule) that is misclassified. For example, let us suppose we have a set of 50 data points. We randomly choose 40 of them and use those to train a classification rule. If, out of this training set of 40 data points, 4 are misclassified by the rule, the training error for this particular rule is $\frac{4}{40} = 0.1$. On the other hand, the remaining 10 data points form our test data. If four of them are misclassified by the rule then the cross-validation error is $\frac{3}{10} = 0.3$.

Recall, from Section 3.3.3, that the classification rule MSVM depends on the choice of a tuning parameter λ. In particular, refer to (3.22) to see how the objective function (which is minimized to find the classification rule) depends on λ. For small λ, the training error tends to decrease. But increasing λ indiscriminately tends to result in overfitting. This tradeoff is reflected by the cross-validation error, which initially decreases, but increases when λ becomes large enough that the error is driven by overfitting. A sensible choice of λ is one which has low value of the cross-validation error. ITanSVM and TSVM are also dependent on λ in a similar way while GMD

does not depend on λ. In our experiments, we will consider several values of λ ($\lambda = 15^k, k = 0, 1, \ldots, 7$) for each of MSVM, ITanSVM and TSVM. The choice of the base 15 for λ is not set in stone: we chose it as a reasonable compromise between coverage of a large range of values and computational cost. Note that the lowest values for the errors (training and cross-validation) can be attained at different λ values for MSVM, ITanSVM and TSVM. The value of the parameter k (see Eq. 3.22) was set equal to 0.01.

3.3.4.1 Application To Hippocampi Data

This data consists of 82 m-rep models (of Hippocampi), 56 of which are from schizophrenic individuals and the remaining 26 are from healthy control individuals (see Styner *et al.* (2004)). Each of these models have 24 medial atoms, placed in a 8×3 lattice (see Fig. 3.3).

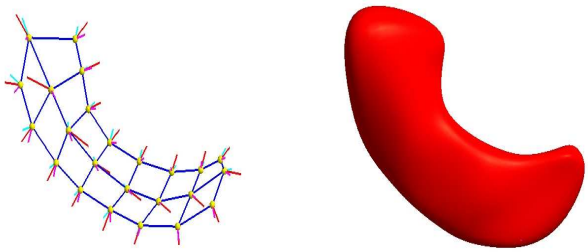


Figure 3.3: *Left: M-rep sheet of a hippocampus with the 24 medial atoms. Right: Surface rendering of the m-rep model.*

We conduct the simulation study in the following way. For each run we randomly remove five data points from the population of 82 and train our classifiers on the remaining 77 data points. For each such population we consider several values of the cost parameter λ ($\lambda = 15^k, k = 0, 1, \ldots, 7$) for ITanSVM, MSVM and TSVM. GMD does not depend on λ. After training we test the classifiers by classifying the five test data points. Aggregating over several simulated replications, the training error and the cross-validation error are calculated.

Fig. 3.4 shows the performance (training error (left panel) and cross-validation error(right panel)) of the different methods.

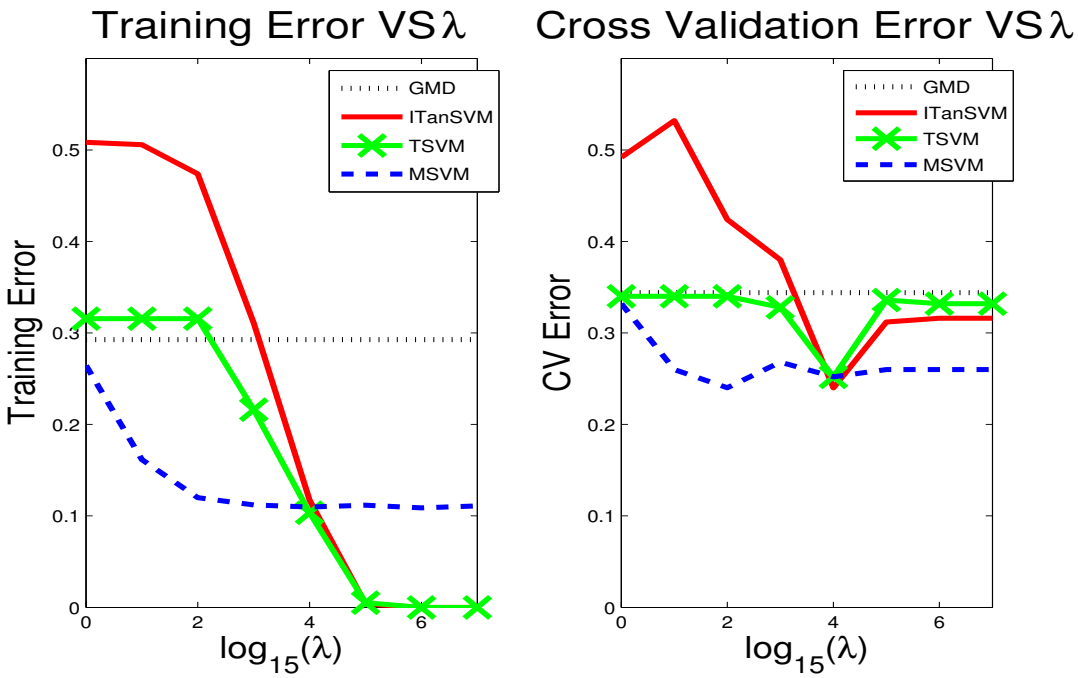


Figure 3.4: *Left: Training errors against cost* $\log_{15}\lambda$. *Right: Cross-validation errors against cost* $\log_{15}\lambda$. *Cross-validation error for MSVM is robust to the choice of* λ.

From Fig. 3.4, we see that MSVM has training error either very close to GMD ($\lambda = 1$) or substantially smaller. On the other hand, for small values of λ the training error of ITanSVM is much higher. MSVM fails to attain a training error of zero while both TSVM and ITanSVM achieve zero training error (at $\lambda = 15^5$ or higher). But this could be due to overfitting by ITanSVM and TSVM, and this idea is validated by their increased cross-validation error for high λ values. We note that the cross-validation errors of ITanSVM and TSVM is very sensitive to the choice of λ, i.e., a good choice of λ appears to be critical for these two methods. In contrast MSVM is much more robust against the choice of λ. In particular, the fact that the cross-validation of MSVM is much more stable for high values of λ is promising. We also note that the cross-validation error of MSVM (at $\lambda = 15^2$) is the least among all methods.

The MSVM algorithm suggests that with increasing λ the distance between c_1 and c_2 should increase (see equation 3.22). We monitor this tendency by plotting $d(c_1, c_2)$ against λ for MSVM, and for ITanSVM and GMD in order to compare the behavior of the solutions. Note that TSVM cannot be compared here since there are no control points involved. Fig. 3.5 plots $\log_{10}(d(c_1, c_2))$ against $\log_{15} \lambda$.

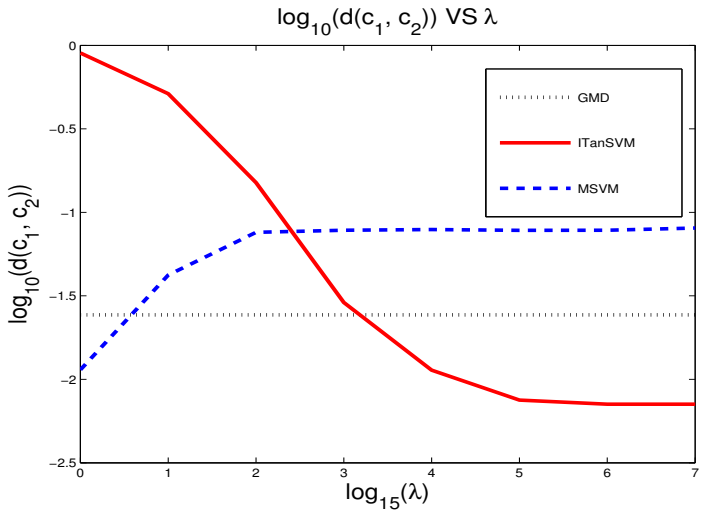


Figure 3.5: $\log_{10}(d(c_1, c_2))$ *against cost* $\log_{15} \lambda$. *The distance between the MSVM control points increases with increasing* λ*, as suggested by the problem formulation (equation 3.22).*

Fig. 3.5 verifies that the solution of the MSVM algorithm behaves as expected: the distance between c_1 and c_2 increases with increasing λ. The behavior of the ITanSVM solution is just the opposite. This behavior may be consistent with overfitting.

An important part of this study is to find out whether the classification rules under consideration give meaningful directions of difference between the classes. In the context of the given problem, the rule that best shows the structural change in the hippocampus is the most valuable. The structural change captured by each method is shown in Figure 3.6. For each classification rule (at the λ which has the least cross-validation error), we project the data points on to direction of separation. The mean of the projected data is calculated. The projected data points with the lowest and highest projection scores give the extent of structural change captured by

the separating direction. The objects in the left are the projected shapes with lowest score, and on the right, with the highest score. The color map shows the surface distance maps of the mean (of projected data points) and projected shapes. Red, green, and blue are inward distance, zero distance, and outward distance respectively.

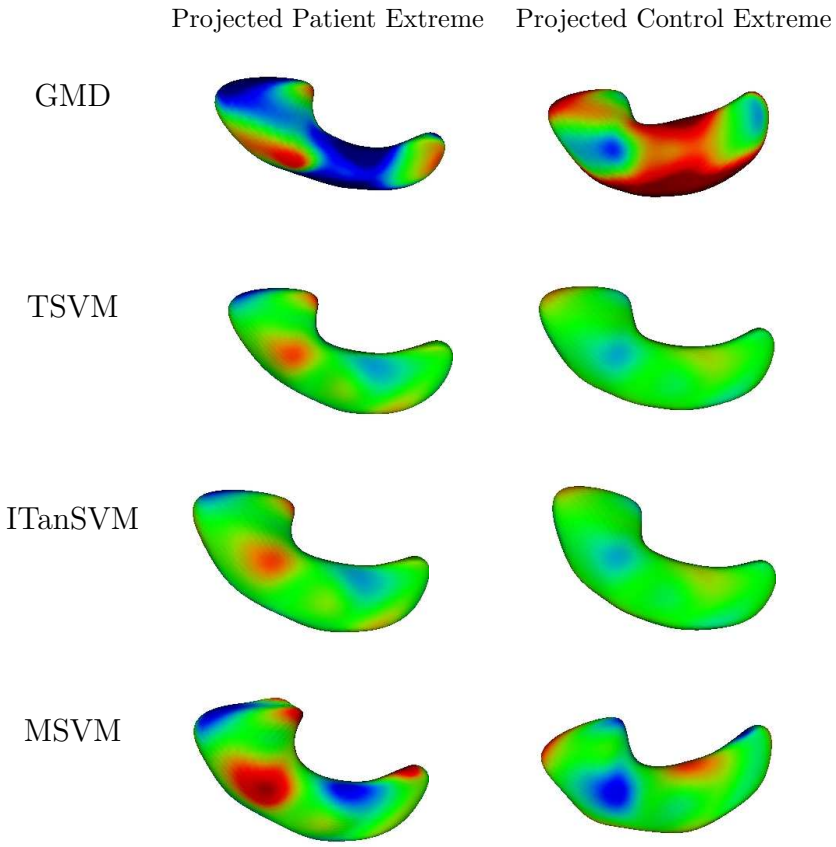


Figure 3.6: *Diagram showing the structural change captured by the different methods. Red, green, and blue are inward distance, zero distance, and outward distance respectively.*

Fig. 3.6 shows that GMD represents a large structural change. But its relevance is questionable because of its poor discriminating performance (Fig. 3.4). GMD shows a lot of structural change, but it fails to isolate the important features which actually separate the two groups. Among the other three methods, MSVM captures the change most strongly. ITanSVM hardly captures the change. This could be related to overfitting where the separating direction feels the microscopic noisy features of the training data and thus fails to capture the relevant structural change. We can

conclude that MSVM provides the best balance between classifying performance and capturing changes in the shapes. Recall, that MSVM is the only method discussed here which works intrinsically on the manifold (and not on a tangent plane, like TSVM and ITanSVM) and this can be attributed to its desirable properties of good classification and informative separating direction.

3.3.4.2 Application to Generated Ellipsoid Data

This data consists of 25 m-rep models of generated distorted ellipsoids. They are simulated by randomly introducing a bending, twisting and tapering of an ellipsoid. We divide them into two groups, a group of 11 with negative twisting parameter and another group of 14 with positive twisting parameter. For our reference we call them the *control* group and the *patient* group respectively. Each of these models have 21 medial atoms, placed in a 7×3 lattice (see Fig. 3.7).

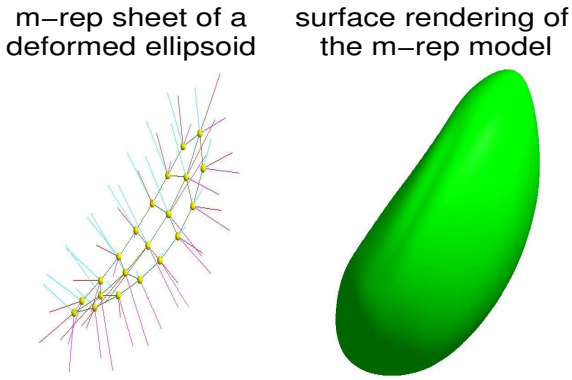


Figure 3.7: *Left: M-rep sheet of one of the simulated distorted ellipsoids used in our study. It has 21 medial atoms. Right: Surface rendering of the m-rep model.*

As in the Hippocampi data set, we compare the performance of the different methods. The values of λ considered are the same as before. Instead of leaving five data points for each simulation, we leave out 3 data points in this case.

Fig. 3.8 shows the performance (training error (left panel) and cross-validation error(right panel)) of the different methods.

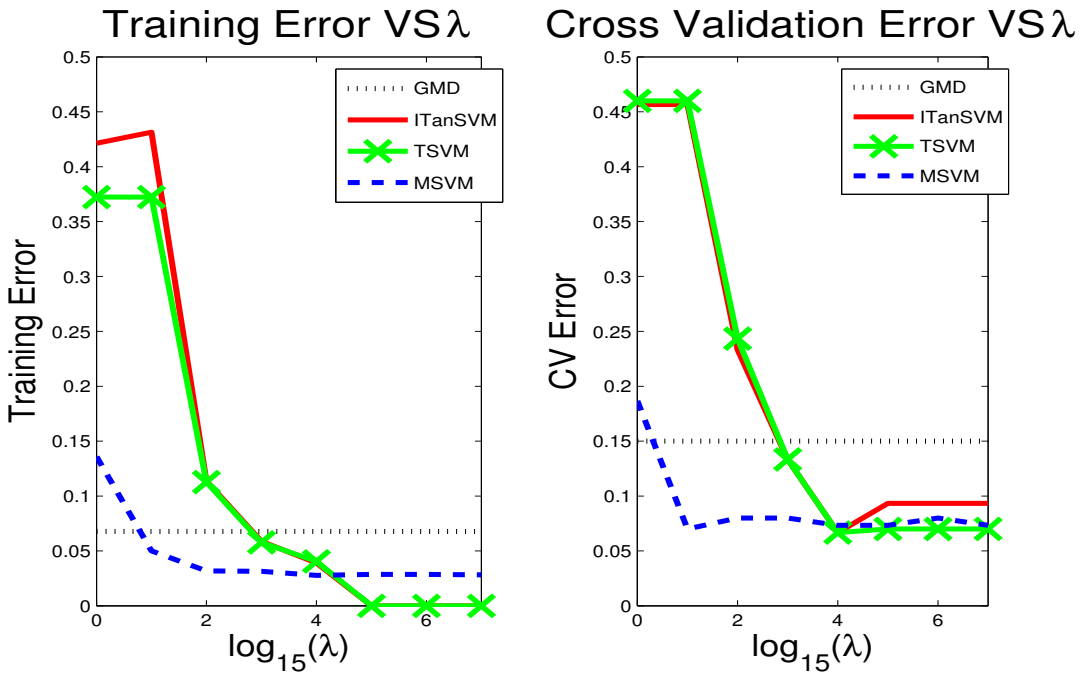


Figure 3.8: *Left: Training Errors against* $\log_{15}\lambda$. *Right: cross validation Errors against cost* $\log_{15}\lambda$. *From the cross validation errors, none of the methods seem to overfit. MSVM is least sensitive to choice of* λ.

MSVM (at $\lambda = 15, 15^4$) has cross validation error very close to the lowest among all the methods. Just as in the Hippocampus data set, the MSVM seems to be the least sensitive to high values of λ. But unlike the hippocampus data set, the cross validation errors of ITanSVM and TSVM do not increase with large λ. It seems for this dataset, these two methods are not overfitting. This can be attributed to the fact that the modes of noise component in this data set are far less than what we have in the real data set of hippocampi.

Fig. 3.9 shows the structural change shown by TSVM, ITanSVM and MSVM. Here, surface distance maps (like Fig. 3.6) are not shown, since it will not be useful to show twisting of the object surface. Instead, wire mesh rendering of the deformed ellipsoid surfaces are shown here. From Fig. 3.9, it can be noted that the structural change shown by ITanSVM and TSVM is a tapering of the ends, while the true mode of difference (twisting) is effectively captured by MSVM.

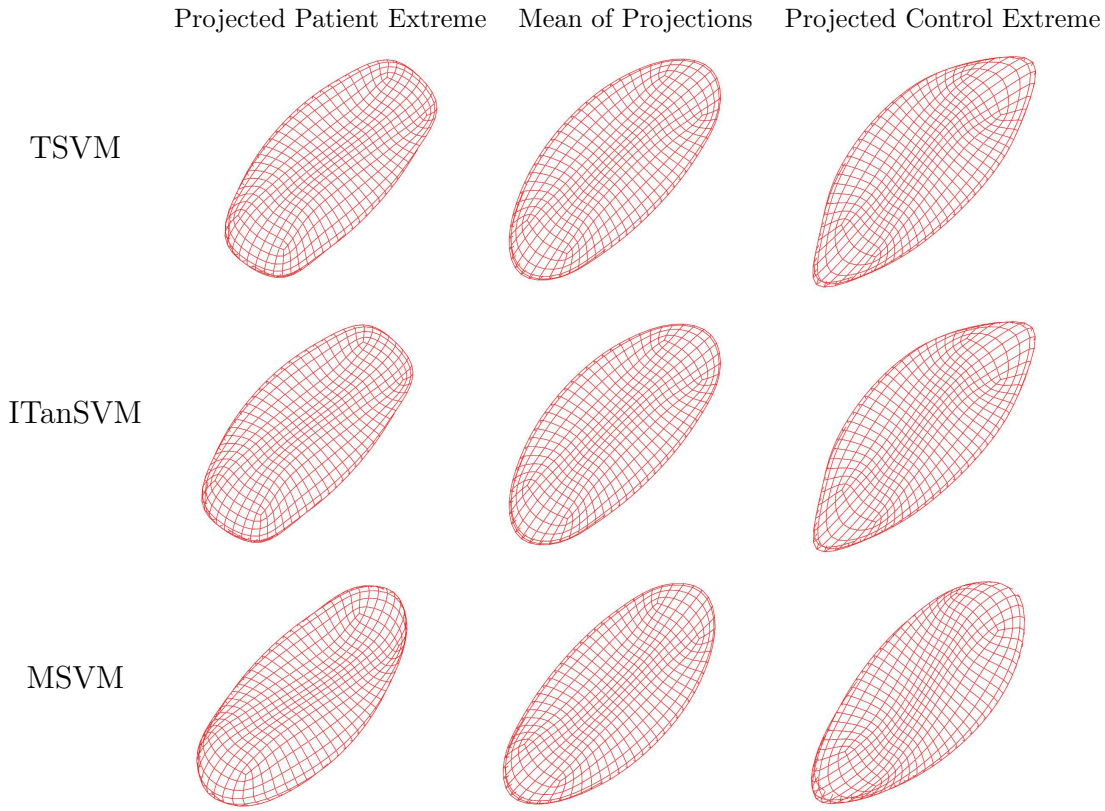


Figure 3.9: *Structural change shown by TSVM, ITanSVM and MSVM. MSVM captures the true mode of difference (twisting), while TSVM and ITanSVM shows tapering/shrinking effect at the ends.*

Again, MSVM seems to be bringing out the best balance of classifying power and capturing of separating features. Though, in this case, ITanSVM and TSVM do not seem to be overfitting, they fail to capture the true mode of change. This example again shows that MSVM, by virtue of its formulation (working on the manifold), captures the nonlinear modes of variation better than methods like ITanSVM and TSVM.

3.4 DWD on Manifolds

In this section, we introduce two approaches to extend DWD to manifold data. The following subsection reviews the standard DWD method.

3.4.1 A Brief Overview of Distance Weighted Discrimination (DWD)

The DWD method, developed by Marron *et al.* (2004) is an improvement upon the Support Vector Machine in HDLSS contexts. For a recent application of DWD to microarray gene expression analysis, see Benito *et al.* (2004). Suppose two classes are separable, which is very likely for HDLSS data. Again, suppose the separating hyperplane is $f(x) = w^T x + b$. DWD finds the hyperplane that minimizes the sum of the inverse distances. This gives larger influence to those points which are close to the hyperplane relative to the points that are farther away from the hyperplane (unlike SVM, where only the points closest to the separating hyperplane have important influence). For separable classes, the DWD method is the solution of the following optimization problem,

$$\begin{aligned} &\text{minimize}_{w,b} \sum_{i=1}^{n} \frac{1}{r_i} \\ &\text{subject to } y_i f(x_i) \geq 0; \quad i = 1, \ldots, n, \end{aligned} \tag{3.24}$$

where r_i is the distance of x_i from the separating hyperplane. As shown in Figure 3.10, DWD finds a hyperplane (green) to separate the two classes (blue circles and red pluses) as well as possible, in the sense of minimizing the sum of the inverse distances from the samples to the hyperplane. The normal to the hyperplane is called the DWD separating direction. The computation of this hyperplane can be formulated as a Second-Order Cone Programming (SOCP) problem and is solved using the software package SDPT3 (for Matlab), which is web-available at Toh *et al.* (2006).

In the following section, an extension of DWD is proposed.

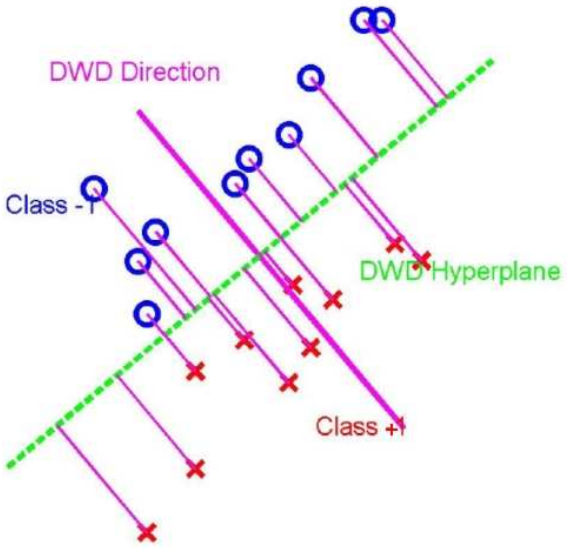


Figure 3.10: *DWD hyperplane (broken green line) to separate two classes, represented by crosses and pluses. The purple vector is the DWD direction of separation.*

3.4.2 Iterative Tangent Plane DWD (ITanDWD)

In this subsection, we generalize DWD to manifold data by implementing standard DWD on multiple tangent planes, which are carefully chosen by an iterative approach. The algorithm is the same as proposed for ITanSVM in Section 3.3.2, except for the fact that in step 2, instead of SVM, the standard DWD method (described in 3.4.1) is implemented. The resulting method is called Iterative Tangent Plane DWD or ITanDWD.

3.4.3 The Manifold DWD (MDWD) method

Analogous to the ITanSVM method, ITanDWD also works by continual approximation of the data by projections on to the tangent plane. This makes ITanDWD unappealing. MDWD aims to be the first approach where all calculations are done on the manifold. Following the framework of Section 3.4.1, MDWD determines a pair of control points that minimizes the sum of the inverse distances from the training data to the separating boundary. The mathematical formulation of the method and the resulting optimization problem are given below.

As given in equation (3.1), the decision function $f_{c_1,c_{-1}}(x)$ is

$$f_{c_1,c_{-1}}(x) = d^2(c_{-1}, x) - d^2(c_1, x)$$

The zero level set of $f_{c_1,c_{-1}}(\cdot)$ defines the separating boundary $H(c_1, c_{-1})$ for a given pair (c_1, c_{-1}). We would like to solve for some $\tilde{c_1}$ and $\tilde{c}_{-1}$ such that

$$(\tilde{c_1}, \tilde{c}_{-1}) = \underset{c_1,c_{-1}}{\operatorname{argmin}} \sum_{i=1}^{n} \frac{1}{d(x_i, H(c_1, c_{-1}))} \tag{3.25}$$

In other words, we want to minimize the sum of the inverse distances from the training data to the separating boundary. Recall, from Chapter 2, that as in the classical SVM literature, we denote y_i to be the class label taking values -1 and 1.

By Lemma 3.1.1, in Euclidean space, note that the distance of any point x from the separating boundary $H(c_1, c_{-1})$ is

$$d(x, H(c_1, c_{-1})) \;=\; \left| \frac{f_{c_1,c_{-1}}(x)}{2d(c_1, c_{-1})} \right| \tag{3.26}$$

Therefore, for a separable training set (see Section 3.1), using (3.3) and (3.12), we can write the distance from the training points to the separating surface as

$$d(x, H(c_1, c_{-1})) \;=\; \frac{y f_{c_1,c_{-1}}(x)}{2d(c_1, c_{-1})}, \tag{3.27}$$

where $y = \pm 1$ is the class label for x. Relation (3.27) will be used as an approximation that is reasonable for data on manifolds lying in a small (convex) neighborhood as this is directly computable for manifold data. Then, using (3.27) in (3.25) we would like to solve for some $\tilde{c_1}$ and $\tilde{c}_{-1}$ such that

$$(\tilde{c_1}, \tilde{c}_{-1}) = \underset{c_1,c_{-1}}{\operatorname{argmin}} \sum_{i=1}^{n} \frac{2d(c_1, c_{-1})}{y_i f_{c_1,c_{-1}}(x_i)} \tag{3.28}$$

Now, in order that the training points are correctly classified by $H(c_1, c_{-1})$, the following constraints should be satisfied:

$\forall i = 1 \ldots n$,

$$\begin{aligned} y_i f(x_i) &\geq 0 \\ \Rightarrow -y_i\{d^2(x_i, c_{-1}) - d^2(x_i, c_1)\} &\leq 0 \\ \Rightarrow h_i &\leq 0 \end{aligned} \tag{3.29}$$

where $h_i = -y_i\{d^2(x_i, c_{-1}) - d^2(x_i, c_1)\}$.

Combining the constraints ($h_i < 0 \ \forall \ i = 1 \ldots n$) with (3.28), the optimization problem becomes

$$\min_{c_1, c_{-1}} \sum_{i=1}^{n} \frac{2d(c_1, c_{-1})}{y_i f_{c_1, c_{-1}}(x_i)} \text{ s.t. } h_i \leq 0 \tag{3.30}$$

Rather than solving the constrained optimization problem in (3.30), we consider the penalized minimization problem as defined below:

Minimize
$$g_\lambda(c_1, c_{-1}) = \sum_{i=1}^{n} \frac{d(c_1, c_{-1})}{y_i f_{c_1, c_{-1}}(x_i)} + \frac{\lambda}{n} \sum_{i=1}^{n} (h_i)_+$$

or,
$$g_\lambda(c_1, c_{-1}) = \sum_{i=1}^{n} \frac{d(c_1, c_{-1})}{y_i\{d^2(x_i, c_{-1}) - d^2(x_i, c_1)\}} + \frac{\lambda}{n} \sum_{i=1}^{n} \Big[-y_i\{d^2(x_i, c_{-1}) - d^2(x_i, c_1)\} \Big]_+ \tag{3.31}$$

where λ is the penalty for violating the constraints given by (3.29).

Equation (3.31) gives the function to be minimized in the separable case. Note that the second term in (3.31) penalizes misclassification. In the non-separable case, the form of the function $g_\lambda(c_1, c_{-1})$ to be minimized is given by:

$$\begin{aligned} g_\lambda(c_1, c_{-1}) = & \sum_{i:\text{ correctly classified}}^{n} \frac{d(c_1, c_{-1})}{y_i\{d^2(x_i, c_{-1}) - d^2(x_i, c_1)\}} + \\ & + \frac{\lambda}{n} \sum_{i=1}^{n} \Big[-y_i\{d^2(x_i, c_{-1}) - d^2(x_i, c_1)\} \Big]_+ \end{aligned} \tag{3.32}$$

or,

$$g_\lambda(c_1, c_{-1}) = \sum_{i=1}^{n} \left[\frac{d(c_1, c_{-1})}{y_i\{d^2(x_i, c_{-1}) - d^2(x_i, c_1)\}} \right]_+ + \frac{\lambda}{n} \sum_{i=1}^{n} \left[-y_i\{d^2(x_i, c_{-1}) - d^2(x_i, c_1)\} \right]_+ \quad (3.33)$$

A gradient descent approach was attempted to solve the above optimization problem, but serious difficulties were encountered. Fig. 3.11 shows the MDWD objective

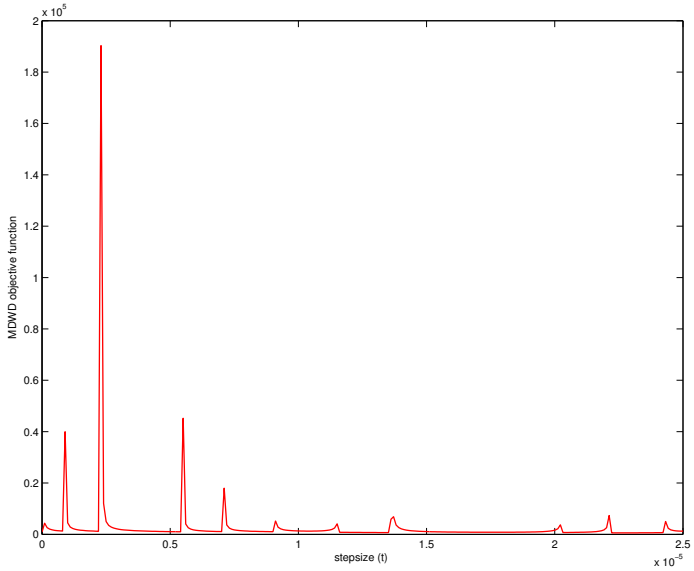


Figure 3.11: *Figure showing the discontinuous nature of the MDWD objective function as a function of the step size along the negative gradient direction.*

function as a function of the step size along the negative gradient direction (with respect to c_{-1}). Note that the curve has several 'spikes', which indicates that the objective function is discontinuous at several points. This phenomenon prevents the negative gradient descent approach from working properly. The discontinuities are due to the denominator in the first term of the objective function given in Eq. (3.33). As soon as one of the misclassified points becomes properly classified, the denominator assumes a very small value and thus the objective function explodes.

Hence, the MDWD method was not implemented. Therefore, in the next section, we only discuss the performance of TDWD (standard Euclidean DWD implemented on the data projected to the tangent plane at the overall geodesic mean) and

ITanDWD and compare them with their SVM counterparts.

3.4.4 Results

In this section, we compare the performance of ITanDWD, TDWD, ITanSVM and TSVM. First, the real data set of hippocampi is revisited. The training errors and cross validation errors are calculated the same way as in Section 3.3.4.

3.4.4.1 Application To Hippocampi Data

Fig. 3.12 shows the performance (training error (left panel) and cross-validation error(right panel)) of the different methods. We note that the training errors of all

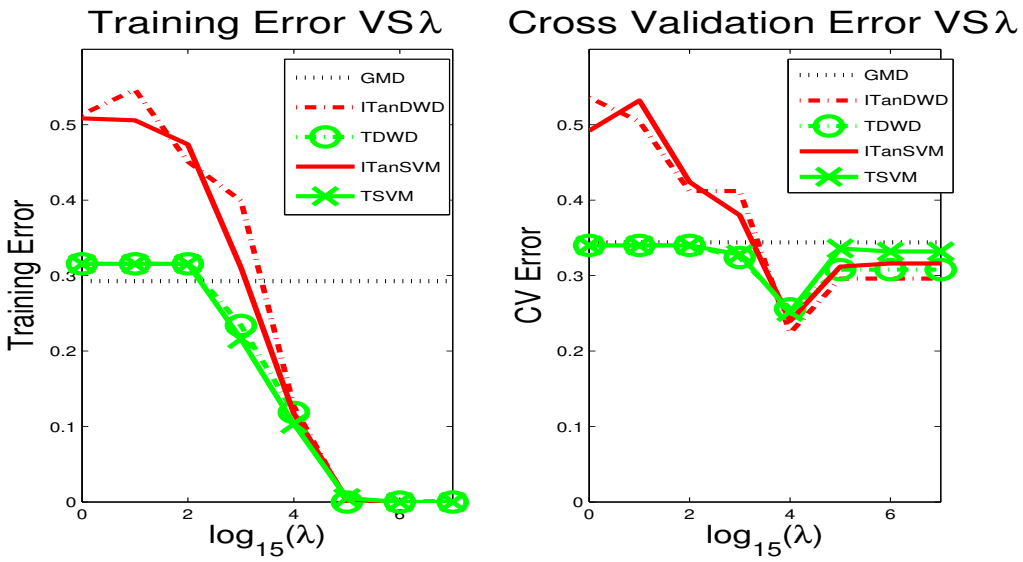


Figure 3.12: *Comparison of methods for Hippocampus data. Left: Training errors against cost* $\log_{15} \lambda$. *Right: Cross-validation errors against cost* $\log_{15} \lambda$. *TDWD and ITanDWD have lower cross validation errors than their SVM counterparts.*

the methods go to zero for higher values of λ. The cross validation errors of both TDWD and ITanDWD are less than their SVM counterparts (for higher values of $\lambda \geq 15^4$). This can be attributed to the fact that DWD is known to be more robust to noise, especially in HDLSS situations. In fact, Figure 3.13 suggests that structural change shown by TDWD and ITanDWD are stronger than their SVM counterparts.

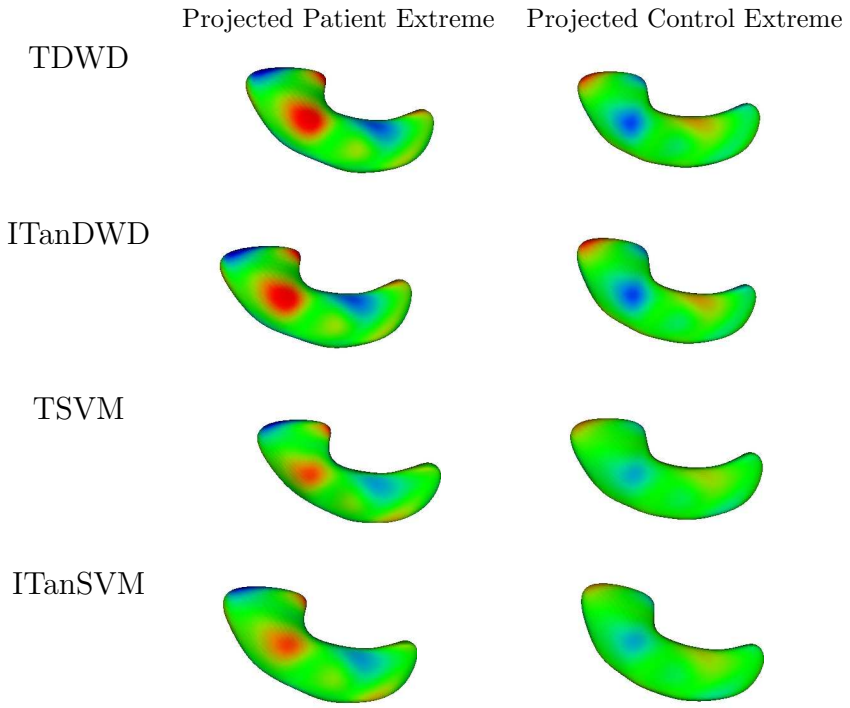


Figure 3.13: *Diagram showing the structural change captured by the different methods. Red, green, and blue are inward distance, zero distance, and outward distance respectively. TDWD and ITanDWD capture stronger changes than their SVM counterparts.*

3.4.4.2 Application To Generated Ellipsoid Data

Fig. 3.14 shows the performance (training error (left panel) and cross-validation error(right panel)) of the different methods. We note that the performances of TDWD and TSVM are very similar, while ITanDWD and ITanSVM are similar. It appears that relative to ITanSVM, ITanDWD is more robust to the choice of higher values of λ (the cross validation error for ITanDWD remains low for $\lambda \geq 15^4$ while that of ITanSVM increases).

Figure 3.15 shows the shape change shown by each of the methods. It is noted that both TDWD and ITanDWD show tapering/shrinking effect at the ends, just like TSVM and ITanSVM. It seems that these two methods also fail to capture the true mode of separation (twisting) owing to the fact that they work on tangent planes.

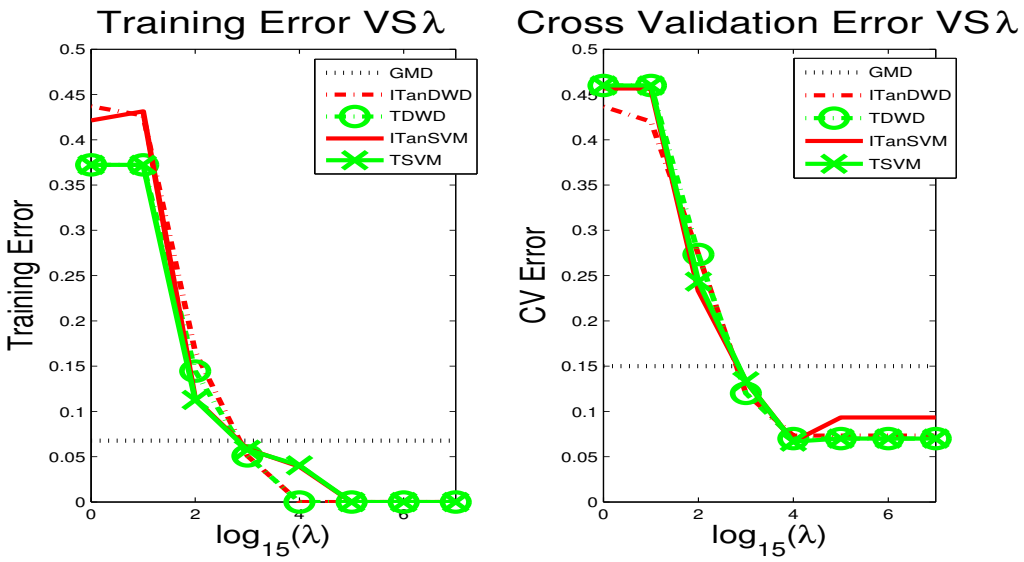


Figure 3.14: *Comparison of methods for Ellipsoid data. Left: Training errors against cost* $\log_{15} \lambda$. *Right: Cross-validation errors against cost* $\log_{15} \lambda$. *ITanDWD has lower cross validation error than ITanSVM for higher values of* λ.

3.4.4.3 Discussion

In both the examples considered here, it seems that the DWD based methods tends to give lower cross validation errors, especially for higher values of the tuning parameter λ. While in the real data set of hippocampi, there was a stronger structural difference captured by TDWD and ITanDWD, there was no such improved performance in the generated ellipsoids case. However, as pointed out earlier, the noise level in the hippocampi data set is much more than in the simulated example of deformed ellipsoids. These preliminary results suggests that it will be interesting to study the performance of the MDWD method.

In the next section we present a modified version of MSVM, which aims at making the method more robust.

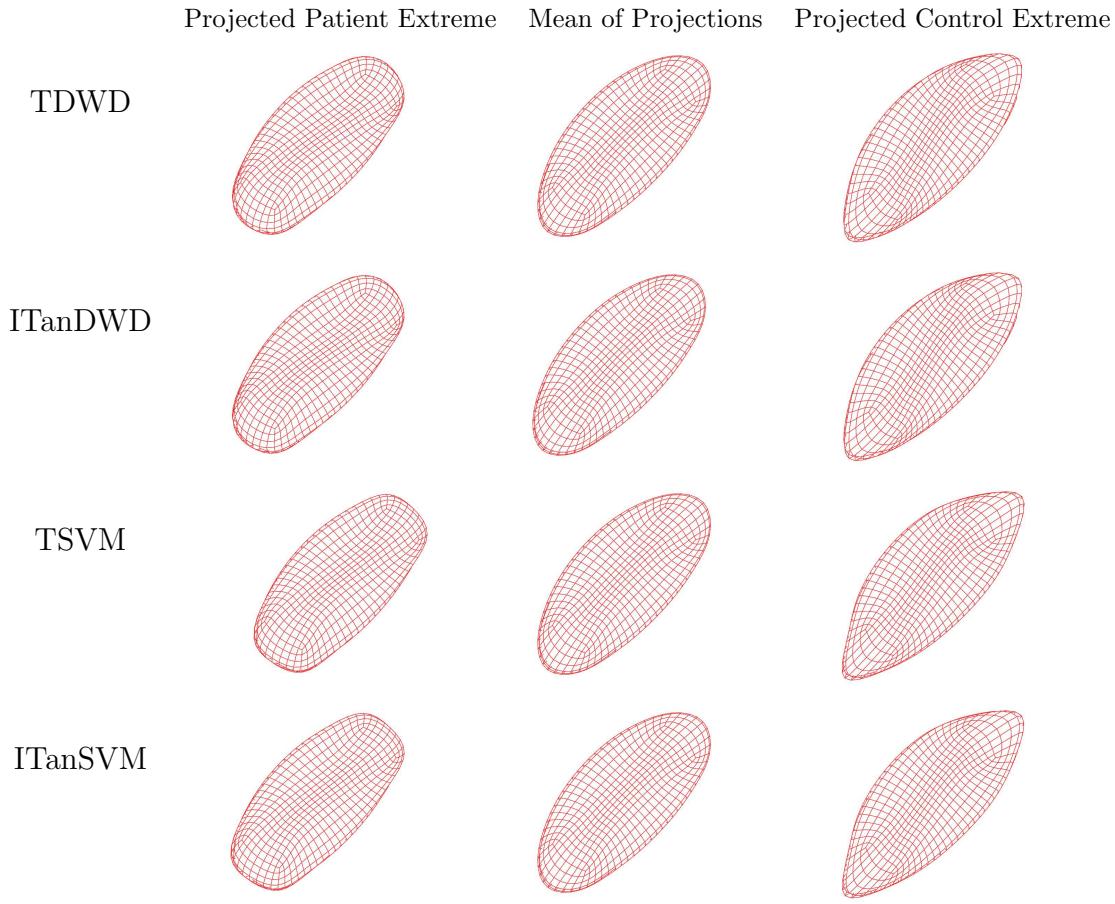


Figure 3.15: *Structural change shown by TDWD, ITanDWD, TSVM and ITanSVM. They show tapering/shrinking effect at the ends and is unable to capture the true mode of difference (twisting), which is captured by the MSVM direction (see Fig. 3.9).*

3.5 MSVM with Enhanced Robustness

The MSVM method was suggested in Section 3.3.3. From the results obtained in Section 3.3.4, we have seen that GMD tends to show maximum structural change while its discriminating power is not very good. MSVM was found to have a nice balance between capturing the shape changes and classifying the data properly. We introduce constraints in the algorithm which will restrict the solution of MSVM to be close to the geodesic means of corresponding classes. The method is described in the next section and results are compared in Section 3.5.2.

3.5.1 Shrinking the Control Points towards the Means

Equation (3.22) gives us the objective function $g_\lambda(c_1, c_{-1})$ to be minimized to construct the MSVM classification rule:

$$g_\lambda(c_1, c_{-1}) = d^2(c_1, c_{-1}) + \frac{\lambda}{n}\sum_{i=1}^{n}\Big[k - y_i\{d^2(x_i, c_{-1}) - d^2(x_i, c_1)\}\Big]_+$$

The first term in the equation maximizes the margin while the second term penalizes misclassified training data and data which are too close to the separating boundary $H(c_1, c_{-1})$. Let M_1 and M_{-1} be the geodesic means. In order to shrink the control points c_1, c_{-1} towards the respective class geodesic means, we introduce a term which attempts to minimize $d^2(c_1, M_1) + d^2(c_{-1}, M_{-1})$. The introduction of this criterion will prevent the solved control points from being too close to each other (provided the geodesic means are not too close to each other). Thus, this approach is expected to prevent overfitting and the potential finding of spurious directions of separation. This change is also expected to improve the identifiability of the solved control points c_1 and c_{-1}. The objective function to be minimized is:

$$\begin{aligned} g_{\lambda,\nu}(c_1, c_{-1}) &= d^2(c_1, c_{-1}) + \frac{\lambda}{n}\sum_{i=1}^{n}\Big[k - y_i\{d^2(x_i, c_{-1}) - d^2(x_i, c_1)\}\Big]_+ + \\ &\quad + \nu[d^2(c_1, M_1) + d^2(c_{-1}, M_{-1})] \end{aligned} \tag{3.34}$$

where ν is a tuning parameter in addition to λ.

Note that as $\nu \to \infty$, the solution to (3.34) will converge to the GMD solution. When $\nu = 0$, it will reduce to the MSVM method suggested in Section (3.3.3). We intend to find the right trade-off between ν and λ and examine whether it improves generalizability.

The resulting objective function in equation (3.34) is solved by the negative gradient descent approach, assuming that the data lie in a small neighborhood. The same

approach was used to solve for the MSVM method, and the algorithm was described in detail in Section 3.3.3.1. The gradients of the function $g_{\lambda,\nu}$ have been provided below.

Given $c_1 = m_1$ and $c_{-1} = m_2$, let us denote by $\Delta_1(m_1, m_2)$ and $\Delta_2(m_1, m_2)$, the gradient of $g_{\lambda,\nu}(c_1, c_{-1})$ w.r.t. c_1 and c_{-1} respectively. The gradients are given by:

$$\Delta_1(m_1, m_2) = \frac{\partial g_{\lambda,\nu}}{\partial m_1} = -2\text{Log}_{m_1}(m_2) - 2\frac{\lambda}{n}\sum_{(i:h_i\geq 0)}(y_i\text{Log}_{m_1}(x_i)) - 2\nu\text{Log}_{m_1}(M_1)$$

$$\Delta_2(m_1, m_2) = \frac{\partial g_{\lambda,\nu}}{\partial m_2} = -2\text{Log}_{m_2}(m_1) + 2\frac{\lambda}{n}\sum_{(i:h_i\geq 0)}(y_i\text{Log}_{m_2}(x_i)) - 2\nu\text{Log}_{m_2}(M_{-1})$$

In the next subsection, we report the behavior of this method as ν changes and compare it to MSVM.

3.5.2 Results

We will refer to the MSVM method with the additional parameter ν as MSVM_ν. For example, when $\nu = 15$, it will be referred to as MSVM_{15}. We note that $\text{MSVM}_0 \equiv$ MSVM. In this section we will present a comparative study of the performances of MSVM_ν for different values of ν ($\nu = 15^k, \quad k = 1, \ldots, 5$). The simulation study was done in the same setup as in Section 3.3.4.

3.5.2.1 Training and Cross Validation Errors

Figures 3.16 and 3.17 show the training and cross validation errors (for different values of ν) for the Hippocampi and Ellipsoid data respectively. Each curve represents a particular MSVM_ν. We observe that with increasing ν, the method behaves more like the GMD (dotted black line). This is expected, since, by construction, ν shrinks the control points towards the respective means.

In both Figures 3.16 and 3.17, MSVM_{15} has a lower minimum (across λ) cross validation error than MSVM. There are other values of ν which give lower cross validation errors than MSVM, but they are not consistent across the two examples.

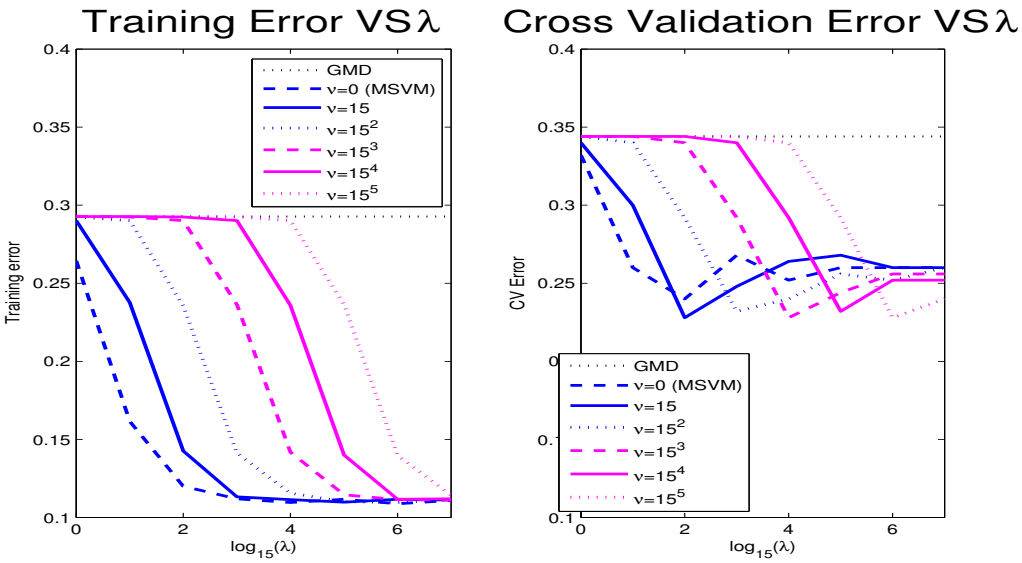


Figure 3.16: *Performance of $MSVM_\nu$'s for Hippocampi data. With increasing ν, the $MSVM_\nu$'s behave more like the GMD. Almost all $MSVM_\nu$'s have lower minimum (across λ) cross validation error than MSVM.*

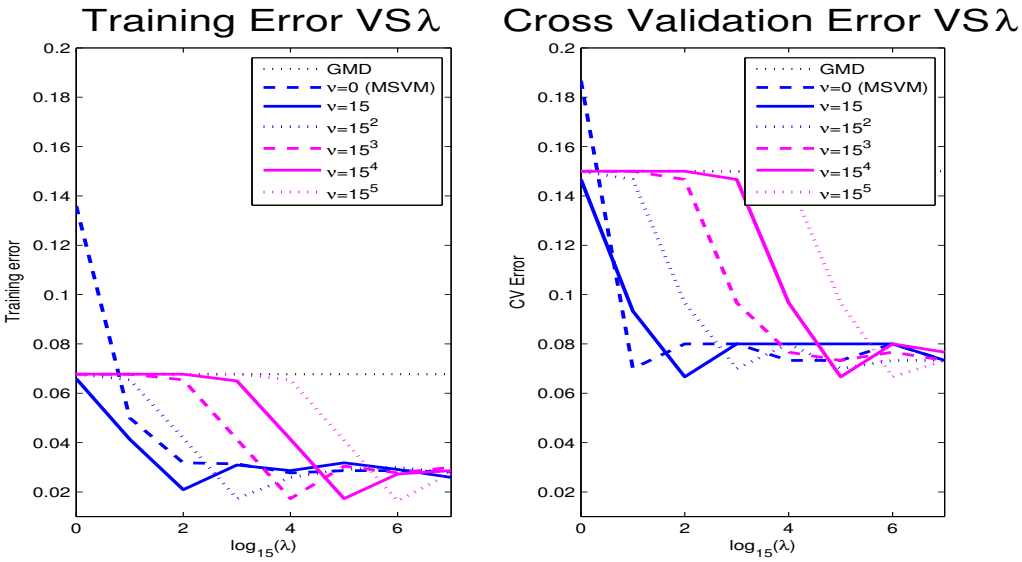


Figure 3.17: *Performance of $MSVM_\nu$'s for Ellipsoid data. With increasing ν, the $MSVM_\nu$'s behave more like the GMD. $MSVM_{15}$ has lower minimum (across λ) cross validation error than MSVM.*

3.5.2.2 Projections on Direction of Separation

Figures 3.18 and 3.19 shows the extreme projections on to the separating directions for the Hippocampus and Ellipsoid data respectively. The projections for MSVM and

MSVM_ν ($\nu = 15, 15^5$) are shown. For the other values of ν, the projected models look very similar.

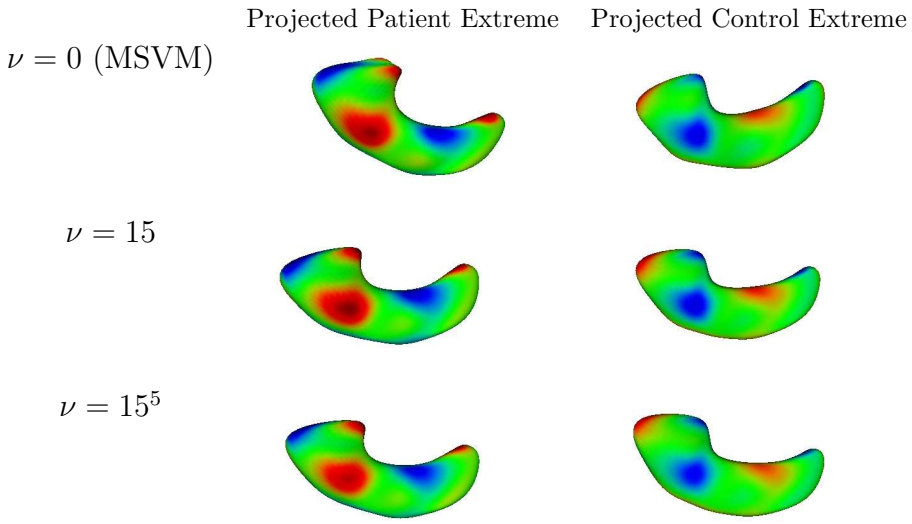


Figure 3.18: *Diagram showing the structural change captured by the MSVM_ν directions. Red, green, and blue are inward distance, zero distance, and outward distance respectively. All the directions show similar changes, and the intensity of the changes are the same too.*

In both Figures 3.18 and 3.19, there does not seem to be much difference in the projected models. To further investigate the effect of ν on the MSVM algorithm, we study the variation of training and cross validation errors over the simulation runs (for each value of ν and λ) in Section 3.5.2.3.

3.5.2.3 Sampling Variation

In this subsection, a comparative study of the variances of the MSVM_ν solutions is conducted. For each run of the simulation (as described in Section 3.3.4), we have a pair of optimum (c_1, c_{-1}) values. For each pair of ν and λ, the quantity $V = Var(c_1) + Var(c_{-1})$ is calculated. The quantity V is a measure of the sampling variation of the MSVM_ν solutions across the different simulation runs. The lower the value of V, the greater the robustness of the method.

Fig. 3.20 compares the sampling variation of the MSVM_ν method for different values of ν (represented by different curves) for the Hippocampi data set. From Fig.

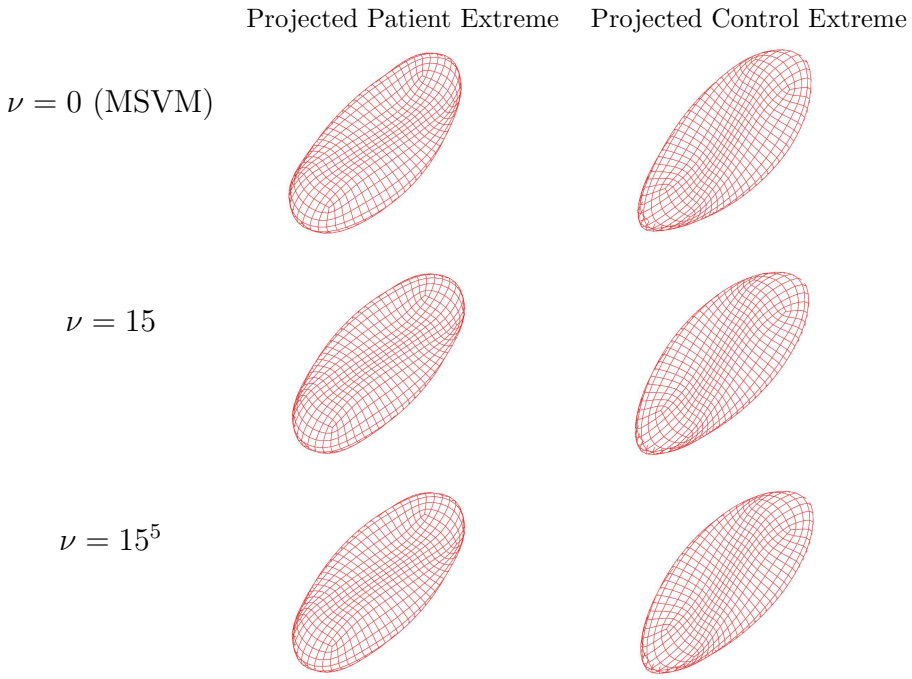


Figure 3.19: *Diagram showing the structural change captured by the MSVM$_\nu$ directions. All of them show the actual mode of difference (twisting), and there is little change for different values of ν.*

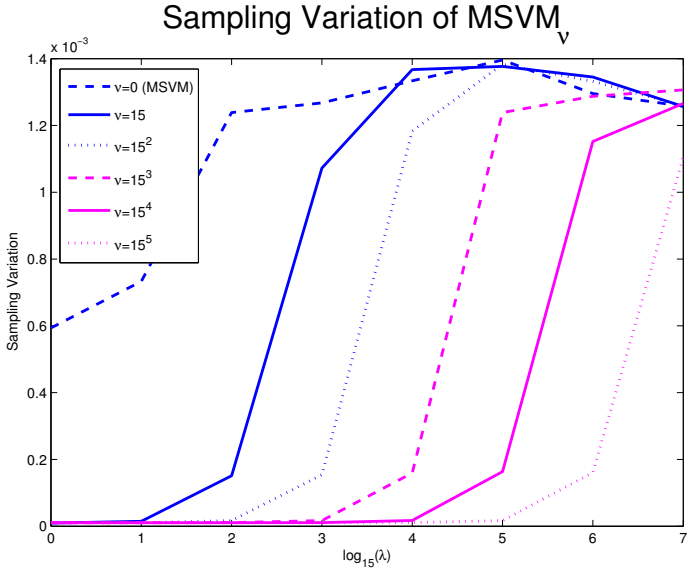


Figure 3.20: *Sampling variation of MSVM$_\nu$'s for Hippocampi data. With increasing ν, the MSVM$_\nu$'s have less variation. The change in variation is large when ν changes from 0 to 15 (for smaller values of λ).*

3.20 we note that as ν increases, the variation in the solution tends to decrease. This is not surprising, since increasing ν forces the solution to be closer to the GMD solution. We recall from previous discussions that taking $\nu = 15$ results in improved cross validation errors (Figures 3.16 and 3.17). Moreover, comparing the sampling variation of the methods (Fig. 3.20), we also note that the decrease in sampling variation is quite large when ν changes from 0 to 15 (for smaller values of λ).

Fig. 3.21 compares the sampling variation of the MSVM$_\nu$ method for different values of ν (represented by different curves) for the deformed ellipsoid data set. The

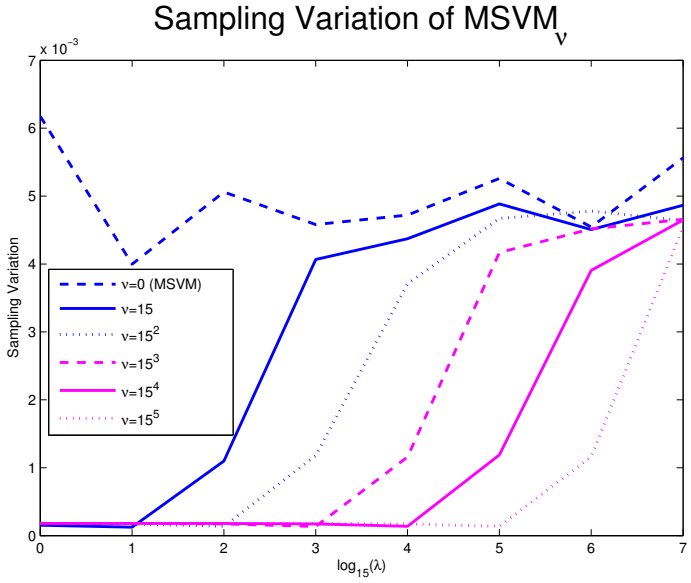


Figure 3.21: *Sampling variation of MSVM$_\nu$'s for ellipsoid data. With increasing ν, the MSVM$_\nu$'s have less variation. The change in variation is large when ν changes from 0 to 15 (for smaller values of λ).*

observations from Fig. 3.21 are similar to those from the Hippocampi data set: the sampling variation reduces when $\nu = 15$. This suggests that a small value of ν helps in improving both the cross validation error and the robustness of the MSVM method.

3.6 Summary

In this chapter, we have presented a general framework for classification of data which lie on manifolds (Section 3.1). Geodesic distance has been used to formulate the

classifier. This framework has been used to extend the methods of Mean Difference (Section 3.2), SVM (Section 3.3) and DWD (Section 3.4) for manifold data.

MSVM (Section 3.3.3) is the only method implemented in this dissertation which works intrinsically on the manifold. It seems that by virtue of this property, it brings about a nice balance of good classification power and informative separating direction. In Section 3.5, we noted that when the MSVM control points are constrained to lie close to the respective geodesic means, the sampling variation of the method reduces. From the preliminary results (Section 3.4.4), it seems that the tangent plane extensions of DWD have better generalizability properties than their SVM counterparts.

In the following chapter, we study the HDLSS asymptotics of data on manifolds and analyze the asymptotic behavior of MSVM as the dimension $d \to \infty$.

CHAPTER 4

Asymptotics of HDLSS Manifold Data

Understanding the geometric structure of HDLSS data is a challenging task due to the limitation of the human perception in visualizing data in more than three dimensions. In fact, some previous work has shown that they have quite different geometry from low dimensional data. In the next section we review the geometric representation of Euclidean HDLSS data.

4.1 Geometric Representation of Euclidean HDLSS Data

Donoho and Tanner (2005), in their asymptotic study on simplices in high dimensional space, found out that the convex hull of n Gaussian data vectors in $\Re^d$ looks like a simplex as the ratio d/n converges to $\gamma \in (0, 1)$ (as $d \to \infty$, $n \to \infty$), in the sense that all points are on the boundary of the convex hull. In our discussion, we focus on the geometry of HDLSS data using the d-asymptotic approach, letting only the dimension d tend to infinity, while fixing the sample size n. This asymptotic domain was studied by Hall *et al.* (2005) and it gives interesting insights into HDLSS situations. We review their observations in the following discussion.

Let $\mathcal{X}(d) = \{X_1(d), \ldots, X_m(d)\}$, where $X_i(d) \in \Re^d, i = 1, \ldots, m$ be i.i.d random vectors distributed as $X(d) = (X^{(1)}, \ldots, X^{(d)})$ with $X^{(i)} \in \Re$ following standard Gaussian distribution. Therefore, each of the $X_i(d)$'s are d-dimensional random vec-

tors from the Gaussian distribution with mean zero and identity covariance matrix. Note that:

(1) The squares of the entries of $X_i(d)$ follows a χ^2_1 distribution, for all $i = 1, \ldots, m$.

(2) The squares of the entries of $\frac{X_i(d)-X_j(d)}{\sqrt{2}}$ follows a χ^2_1 distribution, for all $i, j = 1, \ldots, m; \quad i \neq j$.

Noting these two facts, and using delta method calculations, they showed that

$$\begin{aligned} \frac{||X_i(d)||}{\sqrt{d}} &= 1 + O_p(d^{-\frac{1}{2}}), \\ \frac{||X_i(d) - X_j(d)||}{\sqrt{d}} &= \sqrt{2} + O_p(d^{-\frac{1}{2}}), \\ \text{angle}(X_i(d), X_j(d)) &= \pi/2 + O_p(d^{-\frac{1}{2}}) \end{aligned} \tag{4.1}$$

for all $i, j = 1, \ldots, m; \quad i \neq j$.

Thus, they showed that as $d \to \infty$, the data tend to form an m-simplex with equal pairwise distances. Therefore, there is a deterministic structure in the data. All of the randomness is manifested only through random rotations of the simplex.

Hall *et al.* (2005) also extended the above argument to the non-Gaussian case. Suppose $\mathcal{X}(d) = \{X_1(d), \ldots, X_m(d)\}$, where $X_i(d) \in \Re^d, i = 1, \ldots, m$ are identically distributed random vectors 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,

$$\frac{1}{d}\sum_{k=1}^{d} var(X_i^{(k)}) \to \sigma^2, \text{ for all } i = 1, \ldots, m$$

(3) Viewed as a time series, $X_i^{(1)}, \ldots, X_i^{(d)}, \ldots$ is ρ-mixing for functions that are

dominated by quadratics. That is, for $k, k' = 1, \ldots, d$ with $|k - k'| > r$,

$$\sup_{|k-k'|>r} |E(X_i^{(k)} X_i^{(k')})| \leq \rho(r) \to 0 \text{ as } r \to \infty, \text{ for all } i = 1, \ldots, m. \tag{4.2}$$

If the random vectors satisfy the conditions above, then the distance between $X_i(d)$ and $X_j(d), i \neq j$, is approximately $(2\sigma^2 d)^{\frac{1}{2}}$, in the sense that

$$||X_i(d) - X_j(d)||/\sqrt{d} \xrightarrow{P} (2\sigma^2)^{\frac{1}{2}}. \tag{4.3}$$

Thus after scaling by $\sqrt{d}$, the data vectors $X_i(d)$'s are asymptotically located at the vertices of a regular m-simplex where all the edges are of length $(2\sigma^2)^{\frac{1}{2}}$.

Similar results were extended to the two sample case, where in addition to the data set $\mathcal{X}(d)$, there is another data set $\mathcal{Y}(d) = \{Y_1(d), \ldots, Y_n(d)\}$. The results are summarized below.

(A1) As $d \to \infty$, $\mathcal{X}(d)$ forms an m-simplex where the scaled pairwise distances between the sample points is a constant ($=l_1$).

(A2) As $d \to \infty$, $\mathcal{Y}(d)$ forms an n-simplex where the scaled pairwise distances between the sample points is a constant ($=l_2$).

(A3) As $d \to \infty$, the pairwise distances between sample points (one from $\mathcal{X}(d)$ and another from $\mathcal{Y}(d)$) are at a scaled distance l_{12} from each other.

The convergence holds in the sense of convergence in probability. This geometry was then used in a novel way to study the asymptotic (as $d \to \infty$) behavior of different classification methods like SVM, DWD, Mean Difference and One-Nearest-Neighbor. In the next subsection, we analyze the asymptotic behavior of the MSVM method when applied to Euclidean data with the above geometric structure.

In section 4.2, similar deterministic behavior will be sought in the case when data live in cartesian products of S^2 (the unit sphere in $\Re^3$), with the number of

such spherical components going to infinity. It appears that this idea will also be generalizable to $\mathcal{M}(d)$, as $d \to \infty$.

4.1.1 Behavior of MSVM under Euclidean HDLSS Geometric Structure

In this section we will analyze the MSVM method when applied to data sets of this particular deterministic structure. To make things simple, it is assumed that the data set has exactly the above geometrical representation (and not only in the limiting sense). Moreover, since for a given dimension d, the scaling is done by a constant factor $\sqrt{d}$, we do not show this factor in the calculations. This deterministic structure is summarized as follows:

(B1) $\mathcal{X}(d)$ forms an m-simplex where the pairwise distances between the sample points is a constant ($=l_1$).

(B2) $\mathcal{Y}(d)$ forms an n-simplex where the pairwise distances between the sample points is a constant ($=l_2$).

(B3) All pairs of sample points (one from $\mathcal{X}(d)$ and another from $\mathcal{Y}(d)$) are at a distance l_{12} from each other.

Now, some notations are introduced. Let $\mathcal{X}^{(k)} = \{X_1^{(k)}, \ldots, X_m^{(k)}\}$ denote the set containing the k^{th} component of $\mathcal{X}(d)$. Similarly, $\mathcal{Y}^{(k)}$ is defined. Let $C_x^{(k)}$ and $C_y^{(k)}$ denote the respective sample geodesic means of the k^{th} component (i.e., of $\mathcal{X}^{(k)}$ and $\mathcal{Y}^{(k)}$ respectively). Let $\underline{C_x}$ and $\underline{C_y}$ be the sample geodesic means (in this section, they are the same as the regular Euclidean means, since we are studying data in Euclidean space) of $\mathcal{X}(d), \mathcal{Y}(d)$ respectively, and $\underline{C_x} = (C_x^{(1)}, \ldots, C_x^{(d)})$ and $\underline{C_y} = (C_y^{(1)}, \ldots, C_y^{(d)})$. In the following lemma we will study the distance of any new datum from $\underline{C_x}$. S_m denotes the m-simplex formed by $\mathcal{X}(d)$, while S_n denotes the n-simplex formed by $\mathcal{Y}(d)$.

Lemma 4.1.1. *Let $\mathcal{X}(d), \mathcal{Y}(d)$ be as defined above following the deterministic structure given by (B1)-(B3). Then*

(i) The squared distance between $\underline{C_x}$ and a new point $X_N(d)$ from the X population is given by

$$d^2(\underline{C_x}, X_N(d)) = \frac{l_1^2}{2}(1 + \frac{1}{m}) \tag{4.4}$$

(ii) The squared distance between $\underline{C_x}$ and a new point $Y_N(d)$ from the Y population is given by

$$d^2(\underline{C_x}, Y_N(d)) = l_{12}^2 - \frac{l_1^2}{2}(1 - \frac{1}{m}) \tag{4.5}$$

Proof. $\underline{C_x}$ is the mean of the m-simplex S_m. Without loss of generality, let the vertices of the S_m be given by

$$\begin{aligned} X_1 &= \frac{l_1}{\sqrt{2}}(1, 0, \ldots, 0) \\ X_2 &= \frac{l_1}{\sqrt{2}}(0, 1, 0, \ldots, 0) \\ &\vdots \\ X_m &= \frac{l_1}{\sqrt{2}}(0, \ldots, 0, 1) \end{aligned}$$

This implies

$$\begin{aligned} d^2(\underline{C_x}, X_i(d)) &= \|\frac{l_1}{m\sqrt{2}}\underline{1} - X_i\|^2 \\ &= \frac{l_1^2}{2}(1 - \frac{1}{m}) \end{aligned} \tag{4.6}$$

Now, let $Z \in \Re^d$ be at distance z from each of the sample points in $\mathcal{X}(d)$. Therefore, each triangle formed by $(Z, \underline{C_x}, X_i), i = 1, \ldots, n$ is a right-angled triangle with

hypotenuse being the edge of length z (see Fig. 4.1). Therefore, by Pythagoras' theorem, we have

$$\begin{aligned} d^2(\underline{C_x}, Z) &= d^2(X_i, Z) - d^2(\underline{C_x}, X_i) \\ &= z^2 - \frac{l_1^2}{2}(1 - \frac{1}{m}) \end{aligned} \tag{4.7}$$

If Z comes from the X population, $z = l_1$. If Z comes from the Y population, $z = l_{12}$. This gives us the results (4.4) and (4.5) respectively. □

Note. When a new datum X_N is considered, the underlying geometry of the data changes. In particular, we have a regular simplex S_{m+1} (with $m+1$ vertices), which has the common pairwise distance l_1. Similarly, when a new datum Y_N is considered, the underlying geometric structure is given by a regular simplex S_{n+1}, with common edge length l_2.

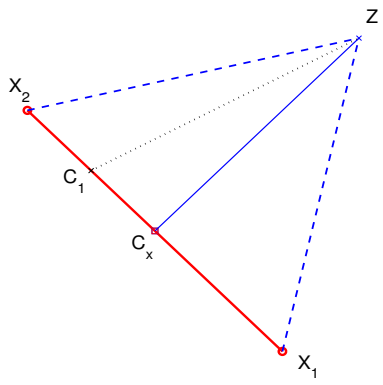


Figure 4.1: $\mathcal{X}(2) = \{X_1, X_2\}$, $X_1, X_2 \in \Re^2$. $\underline{C_x}$ *is the mean of* $\mathcal{X}(2)$. $Z \in \Re^2$ *is equidistant from* X_1 *and* X_2. $\underline{C_x}, Z, X_1$ *form a right triangle with hypotenuse* ZX_1. C_1, Z, X_1 *do not form a right angled triangle. In fact, any point* C_1 $(\neq \underline{C_x})$ *which lies on the line segment* X_1X_2 *does not form a right angled triangle with* X_1 *and* Z. *Consequently* Z *is closest to* $\underline{C_x}$ *among all points on* X_1X_2.

Remark. Fig. 4.1 shows that among all points in the simplex given by $\mathcal{X}(2)$, $\underline{C_x}$ (the mean), is closest to a point Z which is equidistant from the sample points in $\mathcal{X}(2)$. This holds true in the d-dimensional space.

The following lemma states results similar to those of Lemma 4.1.1. Here we study distances from $\underline{C_y}$, the mean of $\mathcal{Y}(d)$.

Lemma 4.1.2. *Let $\mathcal{X}(d), \mathcal{Y}(d)$ be as defined above following the deterministic structure given by (B1)-(B3). Then*

(i) The squared distance between $\underline{C_y}$ and a new point $X_N(d)$ from the X population is given by

$$d^2(\underline{C_y}, X_N(d)) = l_{12}^2 - \frac{l_2^2}{2}(1 - \frac{1}{n}) \tag{4.8}$$

(ii) The squared distance between $\underline{C_y}$ and a new point $Y_N(d)$ from the Y population is given by

$$d^2(\underline{C_y}, Y_N(d)) = \frac{l_2^2}{2}(1 + \frac{1}{n}) \tag{4.9}$$

Using the above two lemmas, we can calculate the distance between $\underline{C_x}$ and $\underline{C_y}$. The following corollary gives us the result.

Corollary 4.1.3. *Let $\mathcal{X}(d), \mathcal{Y}(d)$ be as defined above following the deterministic structure given by (B1)-(B3). Then*

$$d^2(\underline{C_x}, \underline{C_y}) = l_{12}^2 - \frac{l_2^2}{2}(1 - \frac{1}{m}) - \frac{l_2^2}{2}(1 - \frac{1}{n}) \tag{4.10}$$

Proof. From Eq. (4.8), $\underline{C_y}$ is equidistant from each of the vertices of the m-simplex given by $\mathcal{X}(d)$ (the common distance being $\sqrt{l_{12}^2 - \frac{l_2^2}{2}(1 - \frac{1}{n})}$). Then, by Eq. (4.7), the squared distance between $\underline{C_x}$ and $\underline{C_y}$ is given by $l_{12}^2 - \frac{l_2^2}{2}(1 - \frac{1}{m}) - \frac{l_2^2}{2}(1 - \frac{1}{n})$. Note, that here we used $Z = \underline{C_y}$ and therefore $z = \sqrt{l_{12}^2 - \frac{l_2^2}{2}(1 - \frac{1}{n})}$. □

As a remark to Lemma 4.1.1, we noted that if a point Z is equidistant from all the vertices of S_m, then $\underline{C_x}$ is the point (among all points in S_m) which is closest to

Z (see Fig. 4.1). Similar results are studied when there are two simplices (one given by $\mathcal{X}(d)$, the other by $\mathcal{Y}(d)$) in the following lemma.

Lemma 4.1.4. *Let $\mathcal{X}(d), \mathcal{Y}(d)$ be as defined above following the deterministic structure given by (B1)-(B3). Let $C_1 (\neq \underline{C_x})$ be any point in the m-simplex S_m formed by $\mathcal{X}(d)$. Then*

(i) Each vertex of the S_n (given by $\mathcal{Y}(d)$) is at equal distance from C_1. In other words,

$$d(C_1, Y_i(d)) = U = U(C_1), \forall i = 1, \ldots, n \tag{4.11}$$

(ii) The common distance U(C_1) is larger than the common distance between $\underline{C_x}$ and any of the vertices of S_n. Or,

$$d(\underline{C_x}, Y_i(d)) = \sqrt{l_{12}^2 - \frac{l_1^2}{2}(1 - \frac{1}{m})} < U(C_1), \forall i = 1, \ldots, n \tag{4.12}$$

Proof. **Part(i).** For $i = 1, \ldots, n$, consider the $(m+1)$-hedrons created by Y_i and vertices of $\mathcal{X}(d)$.The corresponding edges of these $(m+1)$-hedrons are of the same length. In particular, there is the common base constituting the m-simplex (due to $\mathcal{X}(d)$). In addition, the distance from Y_i to the vertices of $\mathcal{X}(d)$ are the same $(= l_{12})$. Therefore, the n $(m+1)$-hedrons are congruent to each other.

[Let us use Fig 4.2 as an example. Here $\mathcal{X}(3) = \{X_1, X_2\}, \mathcal{Y}(3) = \{Y_1, Y_2\}$, for $X_1, X_2, Y_1, Y_2 \in \Re^3$. The 3-hedrons (here, triangles) $Y_1X_1X_2$ and $Y_2X_1X_2$ are congruent with common base given by the simplex X_1X_2.]

Consequently, the angle between the line segments joining any Y_i and any X_j, and the simplex due to $\mathcal{X}(d)$ are the same. In particular, for any point C_1 ($\in$ m-simplex), $\angle Y_iX_jC_1$ is same for all i, j. (In Fig. 4.2, $\angle Y_1X_1C_1 = \angle Y_1X_2C_1 = \angle Y_2X_1C_1 = \angle Y_2X_2C_1$)

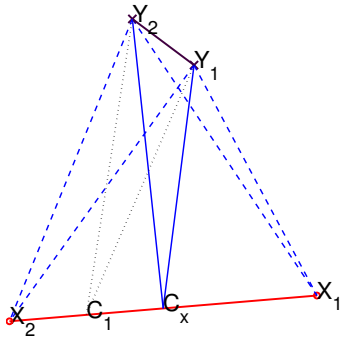


Figure 4.2: *Diagram showing two data sets* $\mathcal{X}(3) = \{X_1, X_2\}$ *and* $\mathcal{Y}(3) = \{Y_1, Y_2\}$, *for* $X_1, X_2, Y_1, Y_2 \in \Re^3$. $\underline{C_x}$ *is the mean of* $\mathcal{X}(3)$. *Note that triangles* $Y_1X_1C_1$ *and* $Y_2X_1C_1$ *are congruent and therefore,* $d(C_1, Y_1) = d(C_1, Y_2)$.

Now, consider any two triangles $Y_1X_1C_1$ and $Y_2X_1C_1$. They are congruent (since (a) $d(Y_1, X_1) = d(Y_2, X_1) = l_{12}$, (b) C_1X_1 is the common edge, and (c) $\angle Y_1X_1C_1 = \angle Y_2X_1C_1$). Therefore, $d(C_1, Y_1) = d(C_1, Y_2)$ (being corresponding edges of congruent triangles). Since Y_1 and Y_2 were arbitrary choices from $\mathcal{Y}(d)$, the relation holds for all the n sample points in $\mathcal{Y}(d)$. Consequently, since the distance from C_1 to any of n vertices only depend on C_1, we can write

$$d(C_1, Y_i(d)) = U = U(C_1), \forall i = 1, \ldots, n$$

Thus Eq. (4.11) is proved.

Part(ii). From Eq. (4.5), it has been proved that $d(\underline{C_x}, Y_i) = \sqrt{l_{12}^2 - \frac{l_1^2}{2}(1 - \frac{1}{m})}$. Note that each of the Y_i's are equidistant from $\mathcal{X}(d)$. By Lemma 4.1.1 and a remark following it, we know that $\underline{C_x}$ is the point in the m-simplex which is closest to each of the points in $\mathcal{Y}(d)$. In other words, $d(\underline{C_x}, Y_i) < d(C_1, Y_i) \quad \forall i = 1, \ldots, n$. Combining these two facts along with part(i) we have

$$d(\underline{C_x}, Y_i(d)) = \sqrt{l_{12}^2 - \frac{l_1^2}{2}(1 - \frac{1}{m})} < U(C_1), \quad \forall i = 1, \ldots, n$$

□

Remark. It can be also proved that each vertex of S_m (given by $\mathcal{X}(d)$) is at equal distance from $C_2(\neq \underline{C_y}) \in S_n$. Also, this distance is greater than the common distance from $\underline{C_y}$ to each of the vertices of the m-simplex.

Corollary 4.1.3 gives the distance between $\underline{C_x}, \underline{C_y}$. The following theorem shows that the solution of the MSVM algorithm (discussed earlier) is $(\underline{C_x}, \underline{C_y})$, when the deterministic structure in the data holds true. We consider the simple situation, where the two data sets $\mathcal{X}(d), \mathcal{Y}(d)$ are separable. We also assume that the permissible candidates for the solution are restricted to the two simplices given by $\mathcal{X}(d)$ and $\mathcal{Y}(d)$. Hall *et al.* (2005) pointed out that for Euclidean data, the SVM separating hyperplane is the perpendicular bisector of the nearest points of the two convex hulls formed by the data sets $\mathcal{X}(d)$ and $\mathcal{Y}(d)$. From that point of view, restricting the choice of optimum control points for MSVM to the simplices S_n and S_m seems to be natural.

Theorem 4.1.5. *Suppose that $\mathcal{X}(d)$ and $\mathcal{Y}(d)$ are data sets as defined above following the deterministic structure given by (B1)-(B3). Let the data sets be separable. In other words, there exists a pair of control points (c_1, c_{-1}) such that $H(c_1, c_{-1})$ separates them. Then, the pair of control points (restricted to the two respective simplices) which define the MSVM separating surface are the means $\underline{C_x}, \underline{C_y}$.*

Proof. Recall, the MSVM algorithm searches for a pair of control points $(\tilde{c_1}, \tilde{c}_{-1})$ such that

$$(\tilde{c_1}, \tilde{c}_{-1}) = \underset{c_1, c_{-1} \in \Re^d}{\operatorname{argmin}} \{d^2(c_1, c_{-1}) + \frac{\lambda}{n} \sum_{i=1}^{n} \Big[k - y_i \{d^2(x_i, c_{-1}) - d^2(x_i, c_1)\} \Big]_+ \}$$

where λ is the penalty parameter for violating the constraints. Since, here we are considering separable data sets, and the candidate solutions are restricted to the

simplices, the MSVM problem formulation reduces to

$$(\tilde{c}_1, \tilde{c}_{-1}) = \underset{c_1 \in S_m \& c_{-1} \in S_n}{\operatorname{argmin}} d^2(c_1, c_{-1}) \tag{4.13}$$

Therefore, in order to prove the statement of the theorem, it is sufficient to prove that

$$(\underline{C_x}, \underline{C_y}) = \underset{c_1 \in S_m \& c_{-1} \in S_n}{\operatorname{argmin}} d^2(c_1, c_{-1}) \tag{4.14}$$

In other words, it would be sufficient to prove that, of all pairs of points (one from the m-simplex and the other from n-simplex) $(\underline{C_x}, \underline{C_y})$ is the pair which is closest to each other. Now, let $C_1 (\neq \underline{C_x})$ belong to S_m and $C_2 (\neq \underline{C_y})$ belong to S_n. Note that $\underline{C_y}$ is equidistant from all the points in $\mathcal{X}(d)$ (by Lemma 4.1.2). Therefore, by Lemma 4.1.4, we can say that

$$d(\underline{C_y}, \underline{C_x}) < d(\underline{C_y}, C_1) \tag{4.15}$$

Again, note that C_1 is equidistant from all the points in $\mathcal{Y}(d)$. Therefore, By Lemma 4.1.4, we can say that

$$d(\underline{C_y}, C_1) < d(C_2, C_1) \tag{4.16}$$

Using Eq. (4.15) and (4.16) gives

$$d(\underline{C_y}, \underline{C_x}) < d(C_2, C_1)$$

This proves the theorem. □

Remark. Theorem 4.1.5 implies that if there is a deterministic structure (given by (B1)-(B3)) in the data, then the separating hyperplanes given by the Mean Difference

method and the MSVM method are the same.

Though all calculations were done assuming conditions (B1)-(B3) holds exactly true, we recall, in reality these conditions hold only in the limiting sense (in probability, as $d \to \infty$). In the next subsection, it is shown that the MSVM solution converges to the Mean Difference solution as $d \to \infty$.

4.1.2 Asymptotic Behavior of MSVM for Euclidean Data

First, we show that the solution of MSVM can be interpreted as an *M-estimate*. Then, the asymptotic behavior of MSVM is studied as $d \to \infty$.

Suppose we are interested in a parameter θ related to the distribution of observations $X_1, \ldots, X_n$. A popular method of finding an estimator $\hat{\theta_n} = \hat{\theta_n}(X_1, \ldots, X_n)$ is to maximize a function of the type

$$M_n(\theta) = \frac{1}{n} \sum_{i=1}^{n} m_\theta(X_i), \tag{4.17}$$

where m_θ is a known function. The estimator which maximizes $M_n(\theta)$ over the parameter space Θ is called an M-estimator (Huber, 1981; Hampel *et al.*, 1986). For example, a very frequently used M-estimator is the Maximum Likelihood Estimator (MLE) where the m_θ's are the loglikelihood functions.

For example, let $X_1, \ldots, X_n$ be independent samples from $N(\theta, 1)$. Then the MLE $\hat{\theta_n}$ of $\theta \in \Re$ can be defined as a sequence such that

$$\hat{\theta_n} = \underset{\theta \in \Re}{\operatorname{argmax}} \sum_{i=1}^{n} \{-(x_i - \theta)^2\}. \tag{4.18}$$

In the asymptotic study of MSVM, we recall that the choice of control points have been restricted to convex combinations of the data. In other words, any candidate

solution (c_1^d, c_{-1}^d) can be written as

$$\begin{aligned} c_1^d &= \sum_{i=1}^{m} \alpha_i X_i(d), \text{ and} \\ c_{-1}^d &= \sum_{i=1}^{n} \beta_i Y_i(d), \end{aligned} \tag{4.19}$$

where $[\underline{\alpha}, \underline{\beta}] \in \Theta$, and

$$\Theta = \{\underline{\alpha}, \underline{\beta} : \alpha_i, \beta_j \in [0,1] \text{ and } \sum_{i=1}^{m} \alpha_i = \sum_{i=1}^{n} \beta_i = 1\}. \tag{4.20}$$

Note: For fixed sample sizes m and n, the dimension of the vectors $\underline{\alpha}$ and $\underline{\beta}$ do not change with d. Throughout our discussion, $\underline{\alpha}$ is of dimension m, while $\underline{\beta}$ is of dimension n.

Therefore, for the separable case, the optimal solution $(\tilde{c}_1^d, \tilde{c}_{-1}^d)$ of MSVM can be written as

$$\begin{aligned} \tilde{c}_1^d &= \sum_{i=1}^{m} \tilde{\alpha_i}^d X_i(d), \text{ and} \\ \tilde{c}_{-1}^d &= \sum_{i=1}^{n} \tilde{\beta_i}^d Y_i(d), \end{aligned} \tag{4.21}$$

where $[\underline{\tilde{\alpha}}^d, \underline{\tilde{\beta}}^d] \in \Theta$ is such that $d^2_{\Re^d}(\tilde{c}_1^d, \tilde{c}_{-1}^d)$ is minimum among all choices of (c_1^d, c_{-1}^d) defined in (4.19).

Note: Here we use the superscript d to indicate that the values of $[\underline{\tilde{\alpha}}^d, \underline{\tilde{\beta}}^d]$ will depend on the dimension d. Again, we note that the dimensions of the vectors $\underline{\alpha}^d$ and $\underline{\beta}^d$ remain m and n throughout.

Now, using (4.19), and recalling that $X^{(k)} \in \Re$ is the k^{th} component of a d-

dimensional vector $X(d) \in \Re^d$, we can write

$$\begin{aligned} d^2_{\Re^d}({c_1}^d, {c_{-1}}^d) &= d^2_{\Re^d}(\sum_{i=1}^{m} \alpha_i X_i(d), \sum_{i=1}^{n} \beta_i Y_i(d)) \\ &= \sum_{k=1}^{d} d^2_{\Re}(\sum_{i=1}^{m} \alpha_i X_i^{(k)}, \sum_{i=1}^{n} \beta_i Y_i^{(k)}). \end{aligned} \tag{4.22}$$

Writing $\underline{\tilde{\theta}^d} = [\underline{\tilde{\alpha}^d}, \underline{\tilde{\beta}^d}]$, we can say $\underline{\tilde{\theta}^d} \in \Theta$ is a sequence of M-estimates, since it maximizes a function $M_d(\underline{\theta})$ given by

$$\begin{aligned} M_d(\underline{\theta}) &= -\frac{1}{d} d^2_{\Re^d}({c_1}^d, {c_{-1}}^d) \\ &= -\frac{1}{d} \sum_{k=1}^{d} d^2_{\Re}(\sum_{i=1}^{m} \alpha_i X_i^{(k)}, \sum_{i=1}^{n} \beta_i Y_i^{(k)}) \\ &= \frac{1}{d} \sum_{k=1}^{d} m_{\underline{\theta}}(\mathcal{X}^{(k)}, \mathcal{Y}^{(k)}), \end{aligned} \tag{4.23}$$

where,

$$m_{\underline{\theta}}(\mathcal{X}^{(k)}, \mathcal{Y}^{(k)}) = -d^2_{\Re}(\sum_{i=1}^{m} \alpha_i X_i^{(k)}, \sum_{i=1}^{n} \beta_i Y_i^{(k)}) \tag{4.24}$$

and $\mathcal{X}^{(k)}, \mathcal{Y}^{(k)}$ are the collections of the k^{th} components of the data $\mathcal{X}(d), \mathcal{Y}(d)$ respectively.

Lemma 4.1.6. *Let $m_{\underline{\theta}}(\cdot, \cdot)$ be as defined in (4.24). Let the following conditions hold:*

(1) For a constant σ,

$$\frac{1}{d} \sum_{k=1}^{d} var(X_i^{(k)}) \to \sigma^2, \text{ for all } i = 1, \ldots, m \tag{4.25}$$

(2) For a constant τ,

$$\frac{1}{d}\sum_{k=1}^{d} var(Y_j^{(k)}) \to \tau^2, \text{ for all } j = 1, \ldots, n \tag{4.26}$$

Then $m_{\underline{\theta}}(\mathcal{X}^{(k)}, \mathcal{Y}^{(k)})$ is dominated by an integrable function for all k.

The proof is given in Section 4.4.

Lemma 4.1.7. *Suppose that $\mathcal{X}(d)$ and $\mathcal{Y}(d)$ are data sets as defined above following the deterministic structure given by (A1)-(A3). Let $M_d(\underline{\theta})$ be as defined in (4.23). Then, we have*

$$M_d(\underline{\theta}) \xrightarrow{P} M(\underline{\theta}), \tag{4.27}$$

as $d \to \infty$, where

$$M(\underline{\theta}) = -[l_{12}^2 - \frac{l_1^2}{2}(1 - \sum_{i=1}^{m} \alpha_i^2) - \frac{l_2^2}{2}(1 - \sum_{i=1}^{n} \beta_i^2)], \tag{4.28}$$

for all $\underline{\theta} \in \Theta$.

The proof is given in Section 4.4. We note that $M(\underline{\theta})$ is maximized by $\underline{\theta} = \underline{\theta}_0 = (\frac{1}{m}, \ldots, \frac{1}{m}, \frac{1}{n}, \ldots, \frac{1}{n})$. In other words, the distance between any two convex combinations (one from the $\mathcal{X}(d)$, another from $\mathcal{Y}(d)$) is minimum when the corresponding points are the means. Recall, that in Theorem 4.1.5, we have proved this fact following a geometric argument also.

In the following theorem, it is shown that the sequence of estimates $\underline{\tilde{\theta}^d} \in \Theta$, defined in 4.23, converges in probability to $\underline{\theta}_0 = (\frac{1}{m}, \ldots, \frac{1}{m}, \frac{1}{n}, \ldots, \frac{1}{n})$, as $d \to \infty$. In other words, the MSVM solution asymptotically behaves like the Mean Difference method as dimension increases.

Theorem 4.1.8. *Let $\mathcal{X}(d)$ and $\mathcal{Y}(d)$ be separable data sets such that the following conditions hold:*

(1) As $d \to \infty$, the data has an asymptotic deterministic structure given by the conditions (A1)-(A3).

(2) Conditions given by Eq. (4.25) and (4.26) are true.

Let $\underline{\tilde{\theta}^d} = [\underline{\tilde{\alpha}^d}, \underline{\tilde{\beta}^d}]$ be the sequence of estimates which defines the MSVM solution (as defined above in (4.21)). Then,

$$\underline{\tilde{\theta}^d} \xrightarrow{P} \underline{\theta}_0 = (\frac{1}{m}, \ldots, \frac{1}{m}, \frac{1}{n}, \ldots, \frac{1}{n}), \tag{4.29}$$

as $d \to \infty$.

The proof is given in Section 4.4. The above theorem states that under the conditions in which the data asymptotically lie in the Euclidean HDLSS deterministic geometric structure, the solution of the MSVM algorithm asymptotically behaves like the Mean Difference method. This is not a purely new observation. Hall *et al.* (2005) showed the equivalence of several classification methods when the dimension increases and the data tends to follow a deterministic pattern. They have pointed out that Euclidean SVM is equivalent to the Mean Difference method when the deterministic structure holds. In this discussion, Theorem 4.1.8 studies the same phenomenon from the viewpoint of *control points*. This approach will be useful when the asymptotic behavior of the MSVM method is studied in the manifold setup. See next section for details.

4.2 Geometric Representation of Manifold HDLSS Data

In the previous subsection, we studied the asymptotic behavior (as $d \to \infty$) of MSVM in the Euclidean case using the geometric structure in Euclidean HDLSS data. Here, we first study the geometry of manifold HDLSS data and then study some asymptotic properties of MSVM when data lie on a manifold. In particular, we will consider data, the components of which are lying in S^2.

Let $\mathcal{X}(d) = \{X_1(d), \ldots, X_m(d)\}$, where $X_i(d), i = 1, \ldots, m$ are i.i.d random vectors distributed as $X(d) = (X^{(1)}, \ldots, X^{(d)})$ with $X^{(i)} \in S^2$. Similarly, let $\mathcal{Y}(d) = (Y_1(d), \ldots, Y_n(d))$. Therefore, $X_i(d), Y_j(d) \in (S^2)^d \quad \forall i = 1, \ldots, m; \quad j = 1, \ldots, n$.

First, we recall some notational conventions. We denote the geodesic distance between two points $X^{(1)}, X^{(2)} \in S^2$ by $d_{S^2}(X^{(1)}, X^{(2)})$. The distance between two points $X_1(d), X_2(d) \in (S^2)^d$ is denoted by $d_{\underline{S^2}}(X_1(d), X_2(d))$. In short, the geodesic distance defined on S^2 is denoted by $d_{S^2}(\cdot, \cdot)$, while the geodesic distance defined on $(S^2)^d$ is denoted by $d_{\underline{S^2}}(\cdot, \cdot)$. The relation between these two distance measures is given by

$$d^2_{\underline{S^2}}(X_1(d), X_2(d)) = \sum_{k=1}^{d} d^2_{S^2}(X_1^{(k)}, X_2^{(k)}), \tag{4.30}$$

where $X_i(d) = \{X_i^{(1)}, \ldots, X_i^{(d)}\} \in (S^2)^d$ and $X_i^{(k)} \in S^2$, $i = 1, 2$ and $k = 1, \ldots, d$.

Now, we state some results which will be used in our discussion.

Lemma 4.2.1. *(1) Let Z_d be a sequence of i.i.d. random variables with $EZ_d^2 = \mu^2$ and $EZ_d^4 = \sigma^2 < \infty$. Then*

$$\frac{\sqrt{\sum_{i=1}^{d} Z_i^2}}{\sqrt{d}} = \mu + O_p(d^{-\frac{1}{2}}). \tag{4.31}$$

(2) Let l be a constant. If Z_d is a sequence of random variables such that the relation

$$\frac{Z_d}{\sqrt{d}} = l + O_p(d^{-\frac{1}{2}}) \tag{4.32}$$

holds, then the following is true:

$$\frac{Z_d^2}{d} = l^2 + O_p(d^{-\frac{1}{2}}). \tag{4.33}$$

(3) If Z_d is a sequence of random variables such that relation (4.33) holds, then relation (4.32) holds true.

The proof is given in Section 4.4.

The following lemma studies some asymptotic properties of pairwise distances of data $\in (S^2)^d$ as $d \to \infty$. We restrict our analysis to the case where the entries $X^{(k)}$'s in $X(d)$ are i.i.d random variables. Similarly, $Y^{(k)}$'s are assumed to be i.i.d. random variables.

Lemma 4.2.2. *Let $\mathcal{X}(d) = \{X_1(d), \ldots, X_m(d)\}$ and $\mathcal{Y}(d) = \{Y_1(d), \ldots, Y_n(d)\}$ be as defined above. Moreover, assume that the entries of the vectors in $X_i(d)$'s and $Y_j(d)$'s are i.i.d. Define $l_1^2 = Ed_{S^2}^2(X_1^{(k)}, X_2^{(k)})$, $l_2^2 = Ed_{S^2}^2(Y_1^{(k)}, Y_2^{(k)})$ and $l_{12}^2 = Ed_{S^2}^2(X_1^{(k)}, Y_1^{(k)})$. Then, the following results hold:*

(i) The scaled geodesic distance between any two points in $\mathcal{X}(d)$ is given by

$$d_{\underline{S^2}}(X_1(d), X_2(d))/\sqrt{d} = l_1 + O_p(d^{-\frac{1}{2}}). \tag{4.34}$$

(ii) The scaled geodesic distance between any two points in $\mathcal{Y}(d)$ is given by

$$d_{\underline{S^2}}(Y_1(d), Y_2(d))/\sqrt{d} = l_2 + O_p(d^{-\frac{1}{2}}). \tag{4.35}$$

(iii) The scaled geodesic distance between any two points (one from $\mathcal{X}(d)$, another from $\mathcal{Y}(d)$) is given by

$$d_{\underline{S^2}}(X_1(d), Y_1(d))/\sqrt{d} = l_{12} + O_p(d^{-\frac{1}{2}}). \tag{4.36}$$

The proof of the lemma is given in Section 4.4.

Lemma 4.2.2 states that under the given conditions, the geodesic distance (scaled by $\sqrt{d}$) between pairs of points from $\mathcal{X}(d)$ (or, $\mathcal{Y}(d)$) is asymptotically a constant, which equals l_1 (or, l_2). Moreover, the scaled pairwise distance between one point in $\mathcal{X}(d)$ and one from $\mathcal{Y}(d)$ is also asymptotically a constant ($= l_{12}$). This implies that a deterministic structure, similar to that in the Euclidean case (Hall *et al.*, 2005), also exists when data lie on $(S^2)^d$ as $d \to \infty$.

We will approach the analysis of the asymptotic behavior of MSVM by studying the structure of the data on tangent planes at particular points on the manifold. We develop the theory in the following discussion. Given a point $p \in S^2$, and the tangent plane T_p at p, we will denote by $d_{T_p}(X, Y)$ the Euclidean distance between $\text{Log}_p(X)$ and $\text{Log}_p(Y) \in T_p$.

In general,

$$d_{S^2}(X, Y) \neq d_{T_p}(X, Y). \tag{4.37}$$

Equality holds true when $p = X$ or Y.

Though it is pointed out that equality in Eq. (4.37) does not hold in general, we can write

$$d^2_{T_p}(X, Y) \;\; = \;\; d^2_{S^2}(X, Y) + \delta(p, X, Y), \tag{4.38}$$

where $\delta(p, X, Y)$ is the error committed in approximating the squared distance on T_p

by the squared distance on S^2. Note that $\delta(p, X, Y)$ is a function of $p, X, Y \in S^2$.

We recall some notations. $\mathcal{X}^{(k)} = \{X_1^{(k)}, \ldots, X_m^{(k)}\}$ denotes the set containing the k^{th} component of $\mathcal{X}(d)$. Similarly, $\mathcal{Y}^{(k)}$ is defined. Let $C_x^{(k)}$ and $C_y^{(k)}$ denote the respective sample geodesic means of the k^{th} component (i.e., of $\mathcal{X}^{(k)}$ and $\mathcal{Y}^{(k)}$ respectively). Let $\underline{C_x}$ and $\underline{C_y}$ be the sample geodesic means of $\mathcal{X}(d), \mathcal{Y}(d)$ respectively. $\underline{C_x} = (C_x^{(1)}, \ldots, C_x^{(d)})$ and $\underline{C_y} = (C_y^{(1)}, \ldots, C_y^{(d)})$.

The following lemma describes a set of conditions under which the deterministic structure exists in the tangent planes $\underline{T_{C_x}}$ and $\underline{T_{C_y}}$.

Lemma 4.2.3. *Let $\mathcal{X}(d) = \{X_1(d), \ldots, X_m(d)\}$ and $\mathcal{Y}(d) = \{Y_1(d), \ldots, Y_n(d)\}$ be as defined above. Let the definitions of Lemma 4.2.2 hold. Define the following quantities:*

$$l_{x0.1\delta} = E_{\mathcal{X}^{(k)}} \delta(C_x^{(k)}, X_1^{(k)}, X_2^{(k)}),$$

$$l_{x0.2\delta} = E_{\mathcal{X}^{(k)}, \mathcal{Y}^{(k)}} \delta(C_x^{(k)}, Y_1^{(k)}, Y_2^{(k)}),$$

$$l_{x0.12\delta} = E_{\mathcal{X}^{(k)}, \mathcal{Y}^{(k)}} \delta(C_x^{(k)}, X_1^{(k)}, Y_1^{(k)}),$$

$$l_{y0.1\delta} = E_{\mathcal{X}^{(k)}, \mathcal{Y}^{(k)}} \delta(C_y^{(k)}, X_1^{(k)}, X_2^{(k)}),$$

$$l_{y0.2\delta} = E_{\mathcal{Y}^{(k)}} \delta(C_y^{(k)}, Y_1^{(k)}, Y_2^{(k)}) \text{ and}$$

$$l_{y0.12\delta} = E_{\mathcal{X}^{(k)}, \mathcal{Y}^{(k)}} \delta(C_y^{(k)}, X_1^{(k)}, Y_1^{(k)}), \text{ for all } k.$$

The following relations hold:

(i) On the tangent plane $\underline{T_{C_x}}$, the scaled distance between any two points in $\mathcal{X}(d)$ is given by

$$d_{\underline{T_{C_x}}}(X_1(d), X_2(d))/\sqrt{d} = \sqrt{l_1^2 + l_{x0.1\delta}} + O_p(d^{-\frac{1}{2}}), \tag{4.39}$$

(ii) On the tangent plane $\underline{T_{C_x}}$*, the scaled distance between any two points in* $\mathcal{Y}(d)$ *is given by*

$$d_{\underline{T_{C_x}}}(Y_1(d), Y_2(d))/\sqrt{d} = \sqrt{l_2^2 + l_{x0.2\delta}} + O_p(d^{-\frac{1}{2}}), \tag{4.40}$$

(iii) On the tangent plane $\underline{T_{C_x}}$*, the scaled distance between any two points (one from* $\mathcal{X}(d)$*, another from* $\mathcal{Y}(d)$*) is given by*

$$d_{\underline{T_{C_x}}}(X_1(d), Y_1(d))/\sqrt{d} = \sqrt{l_{12}^2 + l_{x0.12\delta}} + O_p(d^{-\frac{1}{2}}), \tag{4.41}$$

(iv) On the tangent plane $\underline{T_{C_y}}$*, the scaled distance between any two points in* $\mathcal{X}(d)$ *is given by*

$$d_{\underline{T_{C_y}}}(X_1(d), X_2(d))/\sqrt{d} = \sqrt{l_1^2 + l_{y0.1\delta}} + O_p(d^{-\frac{1}{2}}), \tag{4.42}$$

(v) On the tangent plane $\underline{T_{C_y}}$*, the scaled distance between any two points in* $\mathcal{Y}(d)$ *is given by*

$$d_{\underline{T_{C_y}}}(Y_1(d), Y_2(d))/\sqrt{d} = \sqrt{l_2^2 + l_{y0.2\delta}} + O_p(d^{-\frac{1}{2}}), \tag{4.43}$$

(vi) On the tangent plane $\underline{T_{C_y}}$*, the scaled distance between any two points (one from* $\mathcal{X}(d)$*, another from* $\mathcal{Y}(d)$*) is given by*

$$d_{\underline{T_{C_y}}}(X_1(d), Y_1(d))/\sqrt{d} = \sqrt{l_{12}^2 + l_{y0.12\delta}} + O_p(d^{-\frac{1}{2}}). \tag{4.44}$$

The proof is given in Section 4.4.

Lemma 4.2.3 shows that on the tangent planes at $\underline{C_x}$ and $\underline{C_y}$, there exists a deterministic relation between data, similar to the Euclidean case (Hall *et al.*, 2005).

On the tangent plane at $\underline{C_x}$, we will call the simplex formed by $\mathcal{X}(d)$ and $\mathcal{Y}(d)$ as $S_{m,\underline{C_x}}$ and $S_{n,\underline{C_x}}$ respectively. Similarly, the simplices at $T_{\underline{C_y}}$ will be called $S_{m,\underline{C_y}}$ and $S_{n,\underline{C_y}}$ respectively.

Now, let us concentrate on the tangent plane at $\underline{C_x}$. From results (i)-(iii) in the previous lemma, we can conclude that the mean of the m-simplex S_{m,C_x} on $\underline{T_{C_x}}$ is nearest to every point on the n-simplex S_{n,C_x}. Note, here $\underline{C_x}$ is the mean of the points in $S_{m,C_x} \in \underline{T_{C_x}}$. This implies that, on $\underline{T_{C_x}}$, $\underline{C_x}$ is the point (among all other points in S_{m,C_x}) which is closest to all points in S_{n,C_x} (by Lemma 4.1.4). In addition, Lemma 4.1.1 gives the scaled distance of $\underline{C_x}$ from any vertex $X_i \in S_{m,C_x}$ and $Y_i \in S_{n,C_x}$. The relations are given below:

$$\begin{aligned} d_{\underline{T_{C_x}}}(\underline{C_x}, X_i(d))/\sqrt{d} &= \sqrt{\frac{1}{2}(l_1^2 + l_{x0.1\delta})(1 - \frac{1}{m})} + O_p(d^{-\frac{1}{2}}) \text{ and,} \qquad (4.45) \\ d_{\underline{T_{C_x}}}(\underline{C_x}, Y_j(d))/\sqrt{d} &= \sqrt{\frac{d^2_{\underline{T_{C_x}}}(X_i(d), Y_j(d)) - d^2_{\underline{T_{C_x}}}(\underline{C_x}, X_i(d))}{d}} \\ \Rightarrow d_{\underline{T_{C_x}}}(\underline{C_x}, Y_j(d))/\sqrt{d} &= \sqrt{l_{12}^2 + l_{x0.12\delta} - \frac{1}{2}(l_1^2 + l_{x0.1\delta})(1 - \frac{1}{m})} + O_p(d^{-\frac{1}{2}}), \\ & \qquad (4.46) \end{aligned}$$

for all $i = 1, \ldots, m$ and $j = 1, \ldots, n$. This implies

$$d_{\underline{S^2}}(\underline{C_x}, X_i(d))/\sqrt{d} = \sqrt{\frac{1}{2}(l_1^2 + l_{x0.1\delta})(1 - \frac{1}{m})} + O_p(d^{-\frac{1}{2}}) \text{ and,} \qquad (4.47)$$

$$d_{\underline{S^2}}(\underline{C_x}, Y_j(d))/\sqrt{d} = \sqrt{l_{12}^2 + l_{x0.12\delta} - \frac{1}{2}(l_1^2 + l_{x0.1\delta})(1 - \frac{1}{m})} + O_p(d^{-\frac{1}{2}}). \qquad (4.48)$$

for all $i = 1, \ldots, m$ and $j = 1, \ldots, n$.

Similarly, the scaled distance of $\underline{C_y}$ from any point $Y_i \in \mathcal{Y}(d)$ is given by

$$d_{\underline{S^2}}(\underline{C_y}, Y_j(d))/\sqrt{d} = \sqrt{\frac{1}{2}(l_2^2 + l_{y0.2\delta})(1 - \frac{1}{n})} + O_p(d^{-\frac{1}{2}}), \qquad (4.49)$$

and the scaled distance of $\underline{C_y}$ from any point $X_i \in \mathcal{X}(d)$ is given by

$$d_{\underline{S^2}}(\underline{C_y}, X_i(d))/\sqrt{d} = \sqrt{l_{12}^2 + l_{y0.12\delta} - \frac{1}{2}(l_2^2 + l_{y0.2\delta})(1 - \frac{1}{n})} + O_p(d^{-\frac{1}{2}}), \qquad (4.50)$$

for all $i = 1, \ldots, m$ and $j = 1, \ldots, n$.

Equation (4.48) states that on $(S^2)^d$, the distance between $\underline{C_x}$ and any $Y_i(d) \in \mathcal{Y}(d)$ is asymptotically a constant. Equation (4.50) states that on $(S^2)^d$, the distance between $\underline{C_y}$ and any $X_i(d) \in \mathcal{X}(d)$ is also asymptotically a constant. Now, using equation (4.38), we can write

$$d^2_{T_{C_y^{(k)}}}(C_x^{(k)}, Y_1^{(k)}) \quad = \quad d^2_{S^2}(C_x^{(k)}, Y_1^{(k)}) + \delta(C_y^{(k)}, C_x^{(k)}, Y_1^{(k)})$$

for all $k = 1, \ldots, n$. This implies

$$d^2_{T_{\underline{C_y}}}(\underline{C_x}, Y_1(d)) = d^2_{\underline{S^2}}(\underline{C_x}, Y_1(d)) + \sum_{k=1}^{d} \delta(C_y^{(k)}, C_x^{(k)}, Y_1^{(k)}). \qquad (4.51)$$

Define $l_{y0.x02\delta} = E_{\mathcal{X}^{(k)},\mathcal{Y}^{(k)}} \delta(C_y^{(k)}, C_x^{(k)}, Y_1^{(k)}) \quad \forall k = 1, \ldots, d$. Then, by the *Central Limit Theorem*, the following result holds:

$$\frac{1}{d} \sum_{k=1}^{d} \delta(C_y^{(k)}, C_x^{(k)}, Y_1^{(k)}) = l_{y0.x02\delta} + O_p(d^{-\frac{1}{2}}). \qquad (4.52)$$

Using equations (4.48), (4.51), (4.52) and Lemma 4.2.1, we have

$$d^2_{T_{\underline{C_y}}}(\underline{C_x}, Y_1(d))/d = l_{12}^2 + l_{x0.12\delta} - \frac{1}{2}(l_1^2 + l_{x0.1\delta})(1 - \frac{1}{m}) + l_{y0.x02\delta} + O_p(d^{-\frac{1}{2}}). \qquad (4.53)$$

The above equation, along with equation (4.49) states that on the tangent plane at $\underline{C_y}$, the points of $\mathcal{Y}(d)$ form a regular simplex S_{n,C_y} (with their mean as $\underline{C_y}$). In addition, the geodesic mean $\underline{C_x}$ (of $\mathcal{X}(d)$) is equidistant (w.r.t. $d_{T_{\underline{C_y}}}$) to each of the

vertices of S_{n,C_y}. Therefore, we can use the Euclidean results proved in the previous subsection. In particular, we observe that on the tangent plane $T_{\underline{C_y}}$, $\text{Log}_{\underline{C_y}}(\underline{C_x})$ is normal to S_{n,C_y}.

An important consequence of the normality of $\text{Log}_{\underline{C_y}}(\underline{C_x})$ to S_{n,C_y} is the fact that of all points $\in S_{n,\underline{C_y}}$, the geodesic mean $\underline{C_y}$ is the closest to $\underline{C_x}$ (by Lemma 4.1.1 and 4.1.4). In other words,

$$\underline{C_y} = \underset{v \in S_{n,\underline{C_y}}}{\text{argmin}}\, d^2_{T_{\underline{C_y}}}(v, \underline{C_x}) \tag{4.54}$$

Similar analysis on the tangent plane at $\underline{C_x}$ will prove that of all points $\in S_{m,\underline{C_x}}$, the geodesic mean $\underline{C_x}$ is the closest to $\underline{C_y}$. In other words,

$$\underline{C_x} = \underset{v \in S_{m,\underline{C_x}}}{\text{argmin}}\, d^2_{T_{\underline{C_x}}}(v, \underline{C_y}) \tag{4.55}$$

The above two results will now be used to prove that among *restricted* pairs of points (c_1, c_{-1}), the pair given by the geodesic means $(\underline{C_x}, \underline{C_y})$ are closest (w.r.t. the geodesic distance) to each other. In particular, for c_1, the restricted set is taken to be the image (via Exp) of the convex hull of the $\text{Log}_{\underline{C_x}} X_i(d)$'s. There is a similar restricted set for c_{-1}. The sets are defined as follows:

$$\mathcal{E}_{\underline{C_x}}(\mathcal{X}(d)) = \{p \in (S^2)^d : p = \text{Exp}_{\underline{C_x}}(\sum_{i=1}^{m} \alpha_i \text{Log}_{\underline{C_x}}(X_i(d))); \alpha_i \geq 0; \sum_{i=1}^{m} \alpha_i = 1\}. \tag{4.56}$$

$$\mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d)) = \{p \in (S^2)^d : p = \text{Exp}_{\underline{C_y}}(\sum_{i=1}^{n} \beta_i \text{Log}_{\underline{C_y}}(Y_i(d))); \beta_i \geq 0; \sum_{i=1}^{n} \beta_i = 1\}. \tag{4.57}$$

This representation of control points offers a convenient extension of the idea of a convex hull for manifold data. It allows us to use Euclidean results on tangent planes at $\underline{C_x}$ and $\underline{C_y}$, as will be seen in the following discussion.

The following proposition states the results.

Proposition 4.2.4. *Let $\mathcal{X}(d), \mathcal{Y}(d)$ be as defined before such that the conditions of Lemma 4.2.2 hold. Let the sets $\mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))$ and $\mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))$ be as defined in Eq. (4.56) and (4.57) . Assume that the data $\mathcal{X}(d) \cup \mathcal{Y}(d)$ belong to a small neighborhood.*

(i) Among all points in $\mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))$, $\underline{C_x}$ is closest to $\underline{C_y}$. In other words

$$\underline{C_x} = \underset{c_1 \in \mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))}{\operatorname{argmin}} d^2_{\underline{S^2}}(c_1, \underline{C_y}). \tag{4.58}$$

(ii) Among all points in $\mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))$, $\underline{C_y}$ is closest to $\underline{C_x}$. In other words

$$\underline{C_y} = \underset{c_{-1} \in \mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))}{\operatorname{argmin}} d^2_{\underline{S^2}}(c_{-1}, \underline{C_x}). \tag{4.59}$$

Proof. **Part(i).** We note that on T_{C_x}, the projection of the set $\mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))$ (denoted by $\operatorname{Log}_{\underline{C_x}} \mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))$) is the convex hull of the $\operatorname{Log}_{\underline{C_x}}(X_i)$'s. Therefore, by construction,

$$\operatorname{Log}_{\underline{C_x}} \mathcal{E}_{\underline{C_x}}(\mathcal{X}(d)) = S_{m,\underline{C_x}}$$

Since $\mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))$ is the restricted set in which c_1 can lie, note that the allowable set of points in $T_{\underline{C_x}}$ is given by $S_{m,\underline{C_x}}$. Therefore, to prove that $\underline{C_x}$ is a critical point of the function $d^2_{\underline{S^2}}(c_1, \underline{C_y})$, we will have to show that the directional derivative (along any direction in $S_{m,\underline{C_x}}$) of $d^2_{\underline{S^2}}(c_1, \underline{C_y})$ is equal to zero.

The directional derivative of a function $f(u)$ along a given direction v is given by

$$D_v f(u) = \nabla_u f(u)' v, \tag{4.60}$$

where $\nabla_u f(u)$ is the derivative of $f(u)$ w.r.t. u.

When data lies in a small (convex) neighborhood, we have

$$\nabla_c d^2_{\underline{S^2}}(c, Z) = -2\operatorname{Log}_c(Z) \tag{4.61}$$

Now, let us consider a direction $v \in S_{m,\underline{C_x}}$. Let $c = \underline{C_x}$ and $Z = \underline{C_y}$ in equation (4.61).

Therefore, using (4.60) and (4.61), we have the directional derivative (along any direction in $S_{m,\underline{C_x}}$) of the function $d^2_{\underline{S^2}}(c, \underline{C_y})$ at $c = \underline{C_x}$ as

$$\begin{aligned} D_v d^2_{\underline{S^2}}(c, \underline{C_y})|_{c=\underline{C_x}} &= -2\text{Log}_{\underline{C_x}}(\underline{C_y})'v \\ &= 0, \end{aligned} \tag{4.62}$$

since, we have earlier shown that $\text{Log}_{\underline{C_x}}(\underline{C_y})$ is normal to $S_{m,\underline{C_x}}$, and hence to all $v \in S_{m,\underline{C_x}}$.

This proves that $\underline{C_x}$ is a critical point. However, under the same assumption of data lying in a sufficiently small neighborhood, we can say that $\underline{C_x}$ is the minimizer.

Part(ii). The proof follows similar logic as in part (i). Considering the tangent plane $T_{\underline{C_y}}$ at $\underline{C_y}$, we have to note that $\text{Log}_{\underline{C_y}}\mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d)) = S_{n,\underline{C_y}}$ and $\text{Log}_{\underline{C_y}}(\underline{C_x})$ is normal to $S_{n,\underline{C_y}}$. □

The next set of results study the deterministic structure on the tangent planes at any point in $\mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))$ or $\mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))$. For notational convenience, any point in $\mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))$ is denoted by $\underline{C_{x\alpha}}$, while a point in $\mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))$ is denoted by $\underline{C_{y\beta}}$.

Lemma 4.2.5. *Let $\mathcal{X}(d) = \{X_1(d), \ldots, X_m(d)\}$ and $\mathcal{Y}(d) = \{Y_1(d), \ldots, Y_n(d)\}$ be as defined above. Let the definitions of Lemma 4.2.2 hold. Define the following quantities:*

$$l_{x\alpha.1\delta} = E_{\mathcal{X}^{(k)}}\delta(C^{(k)}_{x\alpha}, X^{(k)}_1, X^{(k)}_2),$$

$$l_{x\alpha.2\delta} = E_{\mathcal{X}^{(k)},\mathcal{Y}^{(k)}}\delta(C^{(k)}_{x\alpha}, Y^{(k)}_1, Y^{(k)}_2),$$

$$l_{x\alpha.12\delta} = E_{\mathcal{X}^{(k)},\mathcal{Y}^{(k)}}\delta(C^{(k)}_{x\alpha}, X^{(k)}_1, Y^{(k)}_1),$$

$$l_{y\beta.1\delta} = E_{\mathcal{X}^{(k)},\mathcal{Y}^{(k)}}\delta(C^{(k)}_{y\beta}, X^{(k)}_1, X^{(k)}_2),$$

$l_{y\beta.2\delta} = E_{\mathcal{Y}^{(k)}}\delta(C_{y\beta}^{(k)}, Y_1^{(k)}, Y_2^{(k)})$ *and*

$l_{y\beta.12\delta} = E_{\mathcal{X}^{(k)},\mathcal{Y}^{(k)}}\delta(C_{y\beta}^{(k)}, X_1^{(k)}, Y_1^{(k)})$, *for all* k.

The following results hold:

(i) On the tangent plane $\underline{T_{C_{x\alpha}}}$, *the scaled distance between any two points in* $\mathcal{X}(d)$ *is given by*

$$d_{\underline{T_{C_{x\alpha}}}}(X_1(d), X_2(d))/\sqrt{d} = \sqrt{l_1^2 + l_{x\alpha.1\delta}} + O_p(d^{-\frac{1}{2}}), \tag{4.63}$$

(ii) On the tangent plane $\underline{T_{C_{x\alpha}}}$, *the scaled distance between any two points in* $\mathcal{Y}(d)$ *is given by*

$$d_{\underline{T_{C_{x\alpha}}}}(Y_1(d), Y_2(d))/\sqrt{d} = \sqrt{l_2^2 + l_{x\alpha.2\delta}} + O_p(d^{-\frac{1}{2}}), \tag{4.64}$$

(iii) On the tangent plane $\underline{T_{C_{x\alpha}}}$, *the scaled distance between any two points (one from* $\mathcal{X}(d)$, *another from* $\mathcal{Y}(d)$*) is given by*

$$d_{\underline{T_{C_{x\alpha}}}}(X_1(d), Y_1(d))/\sqrt{d} = \sqrt{l_{12}^2 + l_{x\alpha.12\delta}} + O_p(d^{-\frac{1}{2}}), \tag{4.65}$$

(iv) On the tangent plane $\underline{T_{C_{y\beta}}}$, *the scaled distance between any two points in* $\mathcal{X}(d)$ *is given by*

$$d_{\underline{T_{C_{y\beta}}}}(X_1(d), X_2(d))/\sqrt{d} = \sqrt{l_1^2 + l_{y\beta.1\delta}} + O_p(d^{-\frac{1}{2}}), \tag{4.66}$$

(v) On the tangent plane $\underline{T_{C_{y\beta}}}$, *the scaled distance between any two points in* $\mathcal{Y}(d)$ *is given by*

$$d_{\underline{T_{C_{y\beta}}}}(Y_1(d), Y_2(d))/\sqrt{d} = \sqrt{l_2^2 + l_{y\beta.2\delta}} + O_p(d^{-\frac{1}{2}}), \tag{4.67}$$

(vi) On the tangent plane $\underline{T_{C_{y\beta}}}$*, the scaled distance between any two points (one from* $\mathcal{X}(d)$*, another from* $\mathcal{Y}(d)$*) is given by*

$$d_{\underline{T_{C_{y\beta}}}}(X_1(d), Y_1(d))/\sqrt{d} = \sqrt{l_{12}^2 + l_{y\beta.12\delta}} + O_p(d^{-\frac{1}{2}}). \tag{4.68}$$

The proof is given in Section 4.4.

The next proposition extends the results of Proposition 4.2.4 to any pair of points from $\mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))$ and $\mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))$.

Proposition 4.2.6. *Let* $\mathcal{X}(d), \mathcal{Y}(d)$ *be as defined before such that the conditions of Lemma 4.2.2 hold. Let the sets* $\mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))$ *and* $\mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))$ *be as defined in Eq. (4.56) and (4.57) . Assume that the data* $\mathcal{X}(d) \cup \mathcal{Y}(d)$ *belong to a small neighborhood.*

(i) Among all points in $\mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))$*,* $\underline{C_x}$ *is closest to any* $\underline{C_{y\beta}} \in \mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))$*. In other words*

$$\underline{C_x} = \underset{\underline{C_{x\alpha}} \in \mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))}{\operatorname{argmin}} d^2_{\underline{S^2}}(\underline{C_{x\alpha}}, \underline{C_{y\beta}}). \tag{4.69}$$

(ii) Among all points in $\mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))$*,* $\underline{C_y}$ *is closest to any* $\underline{C_{x\alpha}} \in \mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))$*. In other words*

$$\underline{C_y} = \underset{\underline{C_{y\beta}} \in \mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))}{\operatorname{argmin}} d^2_{\underline{S^2}}(\underline{C_{x\alpha}}, \underline{C_{y\beta}}). \tag{4.70}$$

The proof uses similar arguments as in Proposition 4.2.4 and is given in Section 4.4.

The following proposition states that under the deterministic conditions, the MSVM method (when the choice of the control points (c_1, c_{-1}) is restricted to the sets $\mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))$ and $\mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))$ respectively) is equivalent to the GMD method.

Proposition 4.2.7. *Suppose that $\mathcal{X}(d)$ and $\mathcal{Y}(d)$ are data sets as defined above following the deterministic structure discussed above. Let the data sets be separable. In other words, there exists a pair of control points (c_1, c_{-1}) such that $H(c_1, c_{-1})$ separates them. Then, the pair of control points (restricted to the sets $\mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))$ and $\mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))$ respectively) which define the MSVM separating surface are the geodesic means $\underline{C_x}, \underline{C_y}$. In other words, MSVM is equivalent to GMD under the stated conditions.*

Proof. Recall the MSVM algorithm searches for a pair of control points $(\tilde{c_1}, \tilde{c}_{-1})$ such that

$$(\tilde{c_1}, \tilde{c}_{-1}) = \underset{c_1, c_{-1} \in (S^2)^d}{\operatorname{argmin}} \{ d^2_{\underline{S^2}}(c_1, c_{-1}) + \frac{\lambda}{n} \sum_{i=1}^{n} \Big[k - y_i \{ d^2_{\underline{S^2}}(x_i, c_{-1}) - d^2_{\underline{S^2}}(x_i, c_1) \} \Big]_+ \} \tag{4.71}$$

where λ is the penalty parameter for violating the constraints. Since, here we are considering separable data sets, and the candidate solutions are restricted to the sets $\mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))$ and $\mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))$ respectively, the MSVM problem formulation reduces to

$$(\tilde{c_1}, \tilde{c}_{-1}) = \underset{c_1 \in \mathcal{E}_{\underline{C_x}}(\mathcal{X}(d)) \& c_{-1} \in \mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))}{\operatorname{argmin}} d^2_{\underline{S^2}}(c_1, c_{-1}) \tag{4.72}$$

Therefore, in order to prove the statement of the proposition, it is sufficient to prove that

$$(\underline{C_x}, \underline{C_y}) = \underset{c_1 \in \mathcal{E}_{\underline{C_x}}(\mathcal{X}(d)) \& c_{-1} \in \mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))}{\operatorname{argmin}} d^2_{\underline{S^2}}(c_1, c_{-1}) \tag{4.73}$$

In other words, it would be sufficient to prove that, of all pairs of points (one from the $\mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))$ and the other from $\mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))$), $(\underline{C_x}, \underline{C_y})$ is the pair which is closest to each other. Now, let $C_1 (\neq \underline{C_x})$ belong to $\mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))$ and $C_2 (\neq \underline{C_y})$ belong to $\mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))$.

By Proposition 4.2.6, we can say that

$$d^2_{\underline{S^2}}(\underline{C_x}, \underline{C_y}) < d^2_{\underline{S^2}}(\underline{C_y}, C_1) \tag{4.74}$$

Again, note that $C_1 \in \mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))$. Therefore, by Proposition 4.2.6,

$$d^2_{\underline{S^2}}(\underline{C_y}, C_1) < d^2_{\underline{S^2}}(C_2, C_1) \tag{4.75}$$

Using Eq. (4.74) and (4.75) gives

$$d^2_{\underline{S^2}}(\underline{C_x}, \underline{C_y}) < d^2_{\underline{S^2}}(C_2, C_1)$$

This proves the proposition. □

Proposition 4.2.7 shows that when the deterministic geometric structure exists in the data, the MSVM solution and the GMD solutions are the same.

4.2.1 Asymptotic Behavior of MSVM for Manifold Data

In this subsection, it is shown that the MSVM solution asymptotically behaves like the GMD solution as data tend to follow the deterministic structure with increasing dimension. We proceed in a way similar to Section 4.1.2, where we interpret the MSVM solution as a sequence of estimates as the dimension $d \to \infty$.

Recall, that the choice of control points has been restricted to the sets $\mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))$ and $\mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))$. Therefore, similar to the Euclidean case, we can represent the control points with the vector $\underline{\theta} = [\underline{\alpha}, \underline{\beta}]$, using the definition of $\mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))$ and $\mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))$. In other words, any candidate solution (c_1^d, c_{-1}^d) can be written as

$$c_1^d \;=\; \mathrm{Exp}_{\underline{C_x}}(\sum_{i=1}^{m} \alpha_i \mathrm{Log}_{\underline{C_x}}(X_i(d))), \text{ and}$$

$$c_{-1}^{d} = \text{Exp}_{\underline{C_y}}(\sum_{i=1}^{n} \beta_i \text{Log}_{\underline{C_y}}(Y_i(d))), \tag{4.76}$$

where $[\underline{\alpha}, \underline{\beta}] \in \Theta$, and

$$\Theta = \{\underline{\alpha}, \underline{\beta} : \alpha_i, \beta_j \in [0,1] \text{ and } \sum_{i=1}^{m} \alpha_i = \sum_{i=1}^{n} \beta_i = 1\}. \tag{4.77}$$

Note: For fixed sample sizes m and n, the dimension of the vectors $\underline{\alpha}$ and $\underline{\beta}$ do not change with d. Throughout our discussion, $\underline{\alpha}$ is of dimension m, while $\underline{\beta}$ is of dimension n.

Therefore, for the separable case, the optimal solution $(\tilde{c}_1^d, \tilde{c}_{-1}^d)$ of MSVM can be written as

$$\begin{aligned} \tilde{c}_1^d &= \text{Exp}_{\underline{C_x}}(\sum_{i=1}^{m} \tilde{\alpha}_i{}^d \text{Log}_{\underline{C_x}}(X_i(d))), \text{ and} \\ \tilde{c}_{-1}^d &= \text{Exp}_{\underline{C_y}}(\sum_{i=1}^{n} \tilde{\beta}_i{}^d \text{Log}_{\underline{C_y}}(Y_i(d))), \end{aligned} \tag{4.78}$$

where $[\underline{\tilde{\alpha}}^d, \underline{\tilde{\beta}}^d] \in \Theta$ is such that $d^2_{\underline{S^2}}(\tilde{c}_1^d, \tilde{c}_{-1}^d)$ is minimum among all choices of (c_1^d, c_{-1}^d) defined in (4.76).

Note: Here, we use the superscript d to indicate that the values of $[\underline{\tilde{\alpha}}^d, \underline{\tilde{\beta}}^d]$ will depend on the dimension d. Again, we note that the dimensions of the vectors $\underline{\alpha}^d$ and $\underline{\beta}^d$ remain m and n throughout.

Now, using (4.76), and recalling that $X^{(k)} \in S^2$ is the k^{th} component of $X(d) \in (S^2)^d$, we can write

$$\begin{aligned} d^2_{\underline{S^2}}(c_1{}^d, c_{-1}{}^d) &= d^2_{\underline{S^2}}(\text{Exp}_{\underline{C_x}}(\sum_{i=1}^{m} \alpha_i \text{Log}_{\underline{C_x}}(X_i(d))), \text{Exp}_{\underline{C_y}}(\sum_{i=1}^{n} \beta_i \text{Log}_{\underline{C_y}}(Y_i(d)))) \\ &= \sum_{k=1}^{d} d^2_{S^2}(\text{Exp}_{C_x^{(k)}}(\sum_{i=1}^{m} \alpha_i Log_{C_x^{(k)}}(X_i^{(k)})), \text{Exp}_{C_y^{(k)}}(\sum_{i=1}^{n} \beta_i Log_{C_y^{(k)}}(Y_i^{(k)}))), \end{aligned} \tag{4.79}$$

Writing $\underline{\tilde{\theta}^d} = [\underline{\tilde{\alpha}^d}, \underline{\tilde{\beta}^d}]$, we can say $\underline{\tilde{\theta}^d} \in \Theta$ is a sequence of M-estimates, since it maximizes the function $M_d(\underline{\theta})$ given by

$$\begin{aligned} M_d(\underline{\theta}) &= -\frac{1}{d} d^2_{\underline{S^2}}({c_1}^d, {c_{-1}}^d)) \\ &= -\frac{1}{d}\sum_{k=1}^{d} d^2_{S^2}(\mathrm{Exp}_{C_x^{(k)}}(\sum_{i=1}^{m}\alpha_i Log_{C_x^{(k)}}(X_i^{(k)})), \mathrm{Exp}_{C_y^{(k)}}(\sum_{i=1}^{n}\beta_i Log_{C_y^{(k)}}(Y_i^{(k)}))) \\ &= \frac{1}{d}\sum_{k=1}^{d} m_{\underline{\theta}}(\mathcal{X}^{(k)}, \mathcal{Y}^{(k)}), \end{aligned} \tag{4.80}$$

where,

$$m_{\underline{\theta}}(\mathcal{X}^{(k)}, \mathcal{Y}^{(k)}) = -d^2_{S^2}(\mathrm{Exp}_{C_x^{(k)}}(\sum_{i=1}^{m}\alpha_i Log_{C_x^{(k)}}(X_i^{(k)})), \mathrm{Exp}_{C_y^{(k)}}(\sum_{i=1}^{n}\beta_i Log_{C_y^{(k)}}(Y_i^{(k)}))) \tag{4.81}$$

and $\mathcal{X}^{(k)}, \mathcal{Y}^{(k)}$ are the collections of the k^{th} components of the data $\mathcal{X}(d), \mathcal{Y}(d)$ respectively.

Lemma 4.2.8. *Suppose that $\mathcal{X}(d)$ and $\mathcal{Y}(d)$ are data sets such that the conditions of Lemma 4.2.2 hold. Let $M_d(\underline{\theta})$ be as defined in (4.80). Then, we have*

$$M_d(\underline{\theta}) \xrightarrow{P} M(\underline{\theta}), \tag{4.82}$$

as $d \to \infty$, where

$$\begin{aligned} M(\underline{\theta}) &= -[l_{12}^2 + l_{x\alpha.12\delta} - \frac{1}{2}(l_1^2 + l_{x\alpha.1\delta})(1 - \sum_{i=1}^{n}\alpha_i^2) + \\ &\quad + l_{y\beta.x\alpha 2\delta} - \frac{1}{2}(l_2^2 + l_{y\beta.2\delta})(1 - \sum_{i=1}^{m}\beta_i^2)] \end{aligned} \tag{4.83}$$

for all $\underline{\theta}$, where $l_{y\beta.x\alpha 2\delta} = E_{\mathcal{X}^{(k)},\mathcal{Y}^{(k)}}\delta(C_{y\beta}^{(k)}, C_{x\alpha}^{(k)}, Y_1^{(k)})$.

The proof is given in Section 4.4.

We note that $M(\underline{\theta})$ is maximized by $\underline{\theta} = \underline{\theta}_0 = (\frac{1}{m}, \ldots, \frac{1}{m}, \frac{1}{n}, \ldots, \frac{1}{n})$ (proved in Proposition 4.2.7), under the condition that data lie in a small neighborhood. In other words, the geodesic distance between any two points (one from $\mathcal{E}_{\underline{C_x}}(\mathcal{X}(d))$, another from $\mathcal{E}_{\underline{C_y}}(\mathcal{Y}(d))$) is minimum when the corresponding points are the geodesic means.

The following proposition states that the sequence of estimates $\underline{\tilde{\theta}^d} \in \Theta$, defined in 4.78, converges in probability to $\underline{\theta}_0$, as $d \to \infty$. In other words, the MSVM solution behaves asymptotically like the GMD method as the dimension increases.

Proposition 4.2.9. *Suppose that $\mathcal{X}(d)$ and $\mathcal{Y}(d)$ are data sets such that the conditions of Lemma 4.2.2 hold. Let $\underline{\tilde{\theta}^d} = [\underline{\tilde{\alpha}^d}, \underline{\tilde{\beta}^d}]$ be the sequence of estimators which defines the MSVM solution (as defined above in (4.78)). If the data lie in a small neighborhood, then*

$$\underline{\tilde{\theta}^d} \xrightarrow{P} \underline{\theta}_0 = (\frac{1}{m}, \ldots, \frac{1}{m}, \frac{1}{n}, \ldots, \frac{1}{n}), \tag{4.84}$$

as $d \to \infty$.

The proof is given in Section 4.4.

By Proposition 4.2.9, we note that when there tends to be a deterministic structure (see Lemma 4.2.2) in the data lying in $(S^2)^d$ with increasing dimension, the MSVM solution asymptotically behaves like the GMD solution. This is an extension of the analysis of the asymptotic behavior of MSVM in the Euclidean case (see Theorem 4.1.8) for manifold data.

4.3 Summary

In this chapter, we studied the asymptotic behavior the MSVM method for both Euclidean data (Section 4.1) and data in $(S^2)^d$ (Section 4.2). We observed that the MSVM solution behaves like the GMD solution with increasing dimension when data

tend to follow the HDLSS deterministic pattern. Hall *et al.* (2005) has shown that methods such as DWD, Nearest Neighbor Classifier have similar behavior for Euclidean data. In particular, they showed that all methods asymptotically behave like the Mean Difference method. In order to draw similar conclusions for manifold data, we also need to study the asymptotic properties of MDWD and Nearest Neighbor Classifier for manifolds. This is an interesting area of future research.

4.4 Technical Details

Proof of Lemma 4.1.6. Using Jensen's inequality, and noting that $0 \leq \alpha_i, \beta_i \leq 1$, we can write

$$\begin{aligned}
|m_{\underline{\theta}}(\mathcal{X}^{(k)}, \mathcal{Y}^{(k)})| &= d^2_{\Re}(\sum_{i=1}^{m} \alpha_i X_i^{(k)}, \sum_{i=1}^{n} \beta_i Y_i^{(k)}) \\
&= (\sum_{i=1}^{m} \alpha_i X_i^{(k)} - \sum_{i=1}^{n} \beta_i Y_i^{(k)})^2 \\
&\leq 2[(\sum_{i=1}^{m} \alpha_i X_i^{(k)})^2 + (\sum_{i=1}^{n} \beta_i Y_i^{(k)})^2] \\
&\leq 2[\sum_{i=1}^{m} \alpha_i (X_i^{(k)})^2 + \sum_{i=1}^{n} \beta_i (Y_i^{(k)})^2] \\
&\leq 2[\sum_{i=1}^{m} (X_i^{(k)})^2 + \sum_{i=1}^{n} (Y_i^{(k)})^2] \qquad (4.85)
\end{aligned}$$

Conditions given by (4.25) and (4.26) imply that $E(X_i^{(k)})^2, E(Y_j^{(k)})^2 < \infty$ for all k and for all $i = 1, \ldots, m$ and $j = 1, \ldots, n$. This implies that $E|m_{\underline{\theta}}(\mathcal{X}^{(k)}, \mathcal{Y}^{(k)})| < \infty$ for all k and this completes the proof. □

Proof of Lemma 4.1.7. Without loss of generality, by appealing to the asymptotic behavior of the pairwise distances between points (as given by (A1)-(A3)), the data can be represented as

$$X_1/\sqrt{d} = \frac{l_1}{\sqrt{2}}(1, 0, \ldots, 0)$$

$$
\begin{aligned}
X_2/\sqrt{d} &= \frac{l_1}{\sqrt{2}}(0,1,0,\ldots,0) \\
&\ldots \\
X_m/\sqrt{d} &= \frac{l_1}{\sqrt{2}}(0,\ldots,0,1,0,\ldots,....)
\end{aligned}
$$

and

$$
\begin{aligned}
Y_1/\sqrt{d} &= \frac{l_2}{\sqrt{2}}(0,\ldots,0,1,0,\ldots,0) + \underline{\mu} \\
Y_2/\sqrt{d} &= \frac{l_2}{\sqrt{2}}(0,\ldots,0,0,1,0,\ldots,0) + \underline{\mu} \\
&\ldots \\
Y_n/\sqrt{d} &= \frac{l_2}{\sqrt{2}}(0,\ldots,0,0,\ldots,0,1,0,\ldots,0) + \underline{\mu},
\end{aligned}
$$

where $\underline{\mu} = \mu(1,\ldots,1,0,\ldots,0)$. Noting that μ is such that $d_{\Re^d}(X_1,Y_1)/\sqrt{d} \longrightarrow^P l_{12}$, we have

$$
l_1^2/2 + l_2^2/2 + \mu(l_1 - l_2)/\sqrt{2} + \mu^2(m+n) = l_{12}^2. \tag{4.86}
$$

Using this representation, for any $\underline{\theta} \in \Theta$ we have

$$
\begin{aligned}
\sum_{i=1}^{m} \alpha_i X_i(d)/\sqrt{d} &= \frac{l_1}{\sqrt{2}}(\alpha_1,\ldots,\alpha_m,0,\ldots,0), \text{ and} \\
\sum_{i=1}^{n} \beta_i Y_i(d)/\sqrt{d} &= \frac{l_2}{\sqrt{2}}(0,\ldots,0,\beta_1,\ldots,\beta_n,0,\ldots,0) + \underline{\mu},
\end{aligned} \tag{4.87}
$$

where the relations hold asymptotically. Using Equations (4.86) and (4.87), we have

$$
\begin{aligned}
M_d(\underline{\theta}) &= -\frac{1}{d} d^2_{\Re^d}(\sum_{i=1}^{m} \alpha_i X_i(d), \sum_{i=1}^{n} \beta_i Y_i(d)) \\
&\longrightarrow^P \sum_{i=1}^{m} (l_1\alpha_i/\sqrt{2} - \mu)^2 + \sum_{i=j}^{n} (l_2\beta_j/\sqrt{2} + \mu)^2
\end{aligned}
$$

$$
\begin{aligned}
&= -[l_{12}^2 - \frac{l_1^2}{2}(1 - \sum_{i=1}^{m} \alpha_i^2) - \frac{l_2^2}{2}(1 - \sum_{i=1}^{n} \beta_i^2)] \\
&= M(\underline{\theta}). \qquad (4.88)
\end{aligned}
$$

This completes the proof. □

The following is a theorem which will be used to prove Theorem 4.1.8.

Theorem 4.4.1 (Theorem 5.7 of van der Vaart (1998))**.** *Let M_d be random functions and let M be a fixed function of $\underline{\theta}$ such that for every $\epsilon > 0$*

$$\sup_{\underline{\theta} \in \Theta} |M_d(\underline{\theta}) - M(\underline{\theta})| \xrightarrow{P} 0, \qquad (4.89)$$

$$\sup_{\underline{\theta}: d(\underline{\theta}, \underline{\theta}_0) \geq \epsilon} M(\underline{\theta}) < M(\underline{\theta}_0), \qquad (4.90)$$

Then any sequence of estimators $\hat{\underline{\theta}}^d$ with

$$M_d(\hat{\underline{\theta}}^d) \geq M_d(\underline{\theta}_0) - o_p(1) \qquad (4.91)$$

converges in probability to $\underline{\theta}_0$.

The above theorem gives us a set of conditions under which a sequence of estimates converge to a particular value.

Proof of Theorem 4.1.8. We shall verify that the conditions of Theorem 4.4.1 holds true. First, it should be noted that $\underline{\theta}_0 = (\frac{1}{m}, \ldots, \frac{1}{m}, \frac{1}{n}, \ldots, \frac{1}{n})$ is the unique minimizer of $M(\underline{\theta})$ (defined in Eq. (4.28)). Therefore, we have the condition (4.90) is satisfied.

The condition (4.91) ensures that $\underline{\tilde{\theta}^d}$'s nearly maximize the M_d's. By definition of $\underline{\tilde{\theta}^d} = [\underline{\tilde{\alpha}^d}, \underline{\tilde{\beta}^d}]$ in Eq. (4.21), we have that satisfied.

A sufficient set of conditions for Eq. (4.89) to hold true is given below (van der Vaart (1998)):

(a) Θ is compact.

(b) $m_{\underline{\theta}} : \Re^{m+n} \longmapsto \Re$ is continuous.

(c) $m_{\underline{\theta}}$ is dominated by an integrable function.

We note that $\Theta = \{\underline{\alpha}, \underline{\beta} : \alpha_i, \beta_j \in [0,1] \text{and} \sum_{i=1}^{m} \alpha_i = \sum_{i=1}^{n} \beta_i = 1\}$. Therefore, Θ is compact and thus (a) holds. Since $m_{\underline{\theta}}$ is a quadratic polynomial, it is continuous, and thus (b) holds. By Lemma 4.1.6, we have condition (c) satisfied.

All conditions for Theorem 4.4.1 has been verified and hence $\underline{\tilde{\theta}^d} \longrightarrow^P \underline{\theta}_0$. This completes the proof. □

Proof of Lemma 4.2.1.

Part (1). Given the conditions, the *Central Limit Theorem* can be applied to the sequence of variables Z_d^2. This implies

$$\sqrt{d}(\frac{1}{d}\sum_{i=1}^{d} Z_i^2 - \mu^2)/\sigma_1 \sim N(0,1)$$

as $d \to \infty$, where $\sigma_1^2 = \sigma^2 - \mu^4$. Therefore, using the *Delta* method we can write,

$$2\mu\sqrt{d}(\sqrt{\tfrac{1}{d}\textstyle\sum_{i=1}^{d} Z_i^2} - \mu)/\sigma_1 \sim N(0,1) \text{ as } d \to \infty,$$

$$\text{or,} \qquad 2\mu\sqrt{d}(\sqrt{\tfrac{1}{d}\textstyle\sum_{i=1}^{d} Z_i^2} - \mu)/\sigma_1 = O_p(1)$$

$$\text{or,} \qquad \frac{\sqrt{\sum_{i=1}^{d} Z_i^2}}{\sqrt{d}} = \mu + O_p(d^{-\frac{1}{2}}).$$

Hence, relation (4.31) holds.

Part (2). It is given that Z_d is a sequence of random variables such that

$$\frac{Z_d}{\sqrt{d}} = l + O_p(d^{-\frac{1}{2}}).$$

This implies

$$\begin{aligned} & Z_d^2 = l^2 d + O_p(1)^2 + 2l\sqrt{d}O_p(1) \\ \Rightarrow \quad & \frac{Z_d^2}{d} = l^2 + O_p(d^{-1}) + 2lO_p(d^{-\frac{1}{2}}) \\ \Rightarrow \quad & \frac{Z_d^2}{d} = l^2 + O_p(d^{-\frac{1}{2}}). \end{aligned}$$

Hence, relation (4.33) holds.

Part (3). It is given that the relation (4.33) holds. This implies

$$\begin{aligned} \frac{Z_d^2}{\sqrt{d}} &= l^2\sqrt{d} + O_p(1) \\ \text{or, } \frac{Z_d}{d^{1/4}} &= \{l^2\sqrt{d} + O_p(1)\}^{\frac{1}{2}} \\ &= ld^{1/4} + \frac{1}{2ld^{1/4}}O_p(1) - \frac{1}{8l^3d^{3/4}}O_p(1) + \dots \\ \text{or, } \frac{Z_d}{\sqrt{d}} &= l + \frac{1}{2ld^{1/2}}O_p(1) - \frac{1}{8l^3d}O_p(1) + \dots \\ &= l + O_p(d^{-\frac{1}{2}}). \end{aligned}$$

Hence, relation (4.32) holds.

□

Proof of Lemma 4.2.2. Lemma 4.2.1 is used to prove the statements of the three parts.

For part (i), Z_k is substituted by $d_{S^2}(X_1^{(k)}, X_2^{(k)})$ and μ by l_1. We note that the finite fourth moment condition is satisfied since $|d_{S^2}(\cdot,\cdot)| \leq \pi$. Therefore using part (1) of the lemma and the relation (4.30) gives us

$$\begin{aligned} \frac{\sqrt{\sum_{k=1}^{d} d_{S^2}^2(X_1^{(k)}, X_2^{(k)})}}{\sqrt{d}} &= l_1 + O_p(d^{-\frac{1}{2}}) \\ \text{or, } \sqrt{\frac{d_{S^2}^2(X_1(d), X_2(d))}{d}} &= l_1 + O_p(d^{-\frac{1}{2}}) \end{aligned}$$

$$\text{or, } \underline{d_{S^2}}(X_1(d), X_2(d))/\sqrt{d} \;=\; l_1 + O_p(d^{-\frac{1}{2}})$$

This proves part (i).

For part (ii), Z_k is substituted by $d_{S^2}(Y_1^{(k)}, Y_2^{(k)})$ and μ by l_2.

For part (iii), Z_k is substituted by $d_{S^2}(X_1^{(k)}, Y_2^{(k)})$ and μ by l_{12}. □

Proof of Lemma 4.2.3. Lemma 4.2.1 is used to prove the statements of this lemma.

For part (i), Z_k is substituted by $d_{T_{C_x^{(k)}}}(X_1^{(k)}, X_2^{(k)})$ in part (1) of Lemma 4.2.1. Therefore, using (4.38)

$$\begin{aligned}
\mu^2 &= EZ_k^2 \\
&= E_{\mathcal{X}^{(k)}} d^2_{T_{C_x^{(k)}}}(X_1^{(k)}, X_2^{(k)}) \\
&= E_{\mathcal{X}^{(k)}} d^2_{S^2}(X_1^{(k)}, X_2^{(k)}) + E_{\mathcal{X}^{(k)}} \delta(C_x^{(k)}, X_1^{(k)}, X_2^{(k)}) \\
&= l_1^2 + l_{x0.1\delta}.
\end{aligned}$$

Now, using part (1) of Lemma 4.2.1 we have,

$$\begin{aligned}
\frac{\sqrt{\sum_{k=1}^{d} d^2_{T_{C_x^{(k)}}}(X_1^{(k)}, X_2^{(k)})}}{\sqrt{d}} &= \sqrt{l_1^2 + l_{x0.1\delta}} + O_p(d^{-\frac{1}{2}}) \\
\text{or, } \sqrt{\frac{d^2_{\underline{T_{C_x}}}(X_1(d), X_2(d))}{d}} &= \sqrt{l_1^2 + l_{x0.1\delta}} + O_p(d^{-\frac{1}{2}}) \\
\text{or, } \underline{d_{T_{C_x}}}(X_1(d), X_2(d))/\sqrt{d} &= \sqrt{l_1^2 + l_{x0.1\delta}} + O_p(d^{-\frac{1}{2}})
\end{aligned}$$

This proves part (i).

Parts (ii)-(vi) can be proved using similar arguments. □

Proof of Lemma 4.2.5. Lemma 4.2.1 is used to prove the statements of this lemma.

For part (i), Z_k is substituted by $d_{T_{C_{x\alpha}^{(k)}}}(X_1^{(k)}, X_2^{(k)})$ in part (1) of Lemma 4.2.1.

Therefore, using (4.38)

$$\begin{aligned}
\mu^2 &= EZ_k^2 \\
&= E_{\mathcal{X}^{(k)}} d^2_{T_{C^{(k)}_{x\alpha}}}(X_1^{(k)}, X_2^{(k)}) \\
&= E_{\mathcal{X}^{(k)}} d^2_{S^2}(X_1^{(k)}, X_2^{(k)}) + E_{\mathcal{X}^{(k)}} \delta(C^{(k)}_{x\alpha}, X_1^{(k)}, X_2^{(k)}) \\
&= l_1^2 + l_{x\alpha.1\delta}.
\end{aligned}$$

Now, using part (1) of Lemma 4.2.1 we have,

$$\begin{aligned}
\frac{\sqrt{\sum_{k=1}^{d} d^2_{T_{C^{(k)}_{x\alpha}}}(X_1^{(k)}, X_2^{(k)})}}{\sqrt{d}} &= \sqrt{l_1^2 + l_{x\alpha.1\delta}} + O_p(d^{-\frac{1}{2}}) \\
\text{or, } \sqrt{\frac{d^2_{\underline{T_{C_{x\alpha}}}}(X_1(d), X_2(d))}{d}} &= \sqrt{l_1^2 + l_{x\alpha.1\delta}} + O_p(d^{-\frac{1}{2}}) \\
\text{or, } d_{\underline{T_{C_{x\alpha}}}}(X_1(d), X_2(d))/\sqrt{d} &= \sqrt{l_1^2 + l_{x\alpha.1\delta}} + O_p(d^{-\frac{1}{2}})
\end{aligned}$$

This proves part (i).

Parts (ii)-(vi) can be proved using similar arguments. □

Proof of Proposition 4.2.6. Using the results in Lemma 4.2.5, and the resulting geometric representation of the data (see the proof of Lemma 4.1.7), we have

$$\begin{aligned}
d_{\underline{T_{C_{x\alpha}}}}(\underline{C_{x\alpha}}, Y_1(d))/\sqrt{d} &= \sqrt{l_{12}^2 + l_{x\alpha.12\delta} - \frac{1}{2}(l_1^2 + l_{x\alpha.1\delta})(1 - \sum_{i=1}^{n} \alpha_i^2)} + O_p(d^{-\frac{1}{2}}) \\
\text{or }, d_{\underline{S^2}}(\underline{C_{x\alpha}}, Y_1(d))/\sqrt{d} &= \sqrt{l_{12}^2 + l_{x\alpha.12\delta} - \frac{1}{2}(l_1^2 + l_{x\alpha.1\delta})(1 - \sum_{i=1}^{n} \alpha_i^2)} + O_p(d^{-\frac{1}{2}})
\end{aligned}$$

Therefore, we have

$$d_{\underline{T_{C_y}}}(\underline{C_{x\alpha}}, Y_1(d))/\sqrt{d} = \sqrt{l_{12}^2 + l_{x\alpha.12\delta} - \frac{1}{2}(l_1^2 + l_{x\alpha.1\delta})(1 - \sum_{i=1}^{n} \alpha_i^2) + l_{y0.x\alpha 2\delta}} + O_p(d^{-\frac{1}{2}}),$$

where $l_{y0.x\alpha 2\delta} = E_{\mathcal{X}^{(k)},\mathcal{Y}^{(k)}}\delta(C_y^{(k)}, C_{x\alpha}^{(k)}, Y_1^{(k)})$.

Recall, that $\mathcal{Y}(d)$ forms the regular simplex $S_{n,\underline{C_y}}$ and by the above equation, each vertex Y_i is equidistant from $\underline{C_{x\alpha}}$. Therefore, using Lemma 4.1.1, we can say that $\text{Log}_{\underline{C_y}}(\underline{C_{x\alpha}})$ is normal to $S_{n,\underline{C_y}}$. This implies,

$$\begin{aligned} D_v d^2_{\underline{S^2}}(c, \underline{C_{x\alpha}})|_{c=\underline{C_y}} &= -2\text{Log}_{\underline{C_y}}(\underline{C_{x\alpha}})'v \\ &= 0, \end{aligned}$$

where $v \in S_{n,\underline{C_y}}$. This, along with the assumption that data the lie in a small neighborhood, implies part (ii) of the lemma.

For part (i), we study the data at $T_{\underline{C_x}}$ and note that $\text{Log}_{\underline{C_x}}(\underline{C_{y\beta}})$ is normal to $S_{m,\underline{C_x}}$. This completes the proof.

□

Proof of Lemma 4.1.7. Using the results in Lemma 4.2.5, and the resulting geometric representation of the data (see the proof of Lemma 4.1.7), we have

$$\begin{aligned} d_{\underline{T_{C_{x\alpha}}}}(\underline{C_{x\alpha}}, Y_1(d))/\sqrt{d} &= \sqrt{l_{12}^2 + l_{x\alpha.12\delta} - \frac{1}{2}(l_1^2 + l_{x\alpha.1\delta})(1 - \sum_{i=1}^{n}\alpha_i^2)} + O_p(d^{-\frac{1}{2}}) \\ \text{or }, d_{\underline{S^2}}(\underline{C_{x\alpha}}, Y_1(d))/\sqrt{d} &= \sqrt{l_{12}^2 + l_{x\alpha.12\delta} - \frac{1}{2}(l_1^2 + l_{x\alpha.1\delta})(1 - \sum_{i=1}^{n}\alpha_i^2)} + O_p(d^{-\frac{1}{2}}) \end{aligned}$$

Therefore, we have

$$d_{\underline{T_{C_{y\beta}}}}(\underline{C_{x\alpha}}, Y_1(d))/\sqrt{d} = \sqrt{l_{12}^2 + l_{x\alpha.12\delta} - \frac{1}{2}(l_1^2 + l_{x\alpha.1\delta})(1 - \sum_{i=1}^{n}\alpha_i^2) + l_{y\beta.x\alpha 2\delta}} + O_p(d^{-\frac{1}{2}}),$$

where $l_{y\beta.x\alpha 2\delta} = E_{\mathcal{X}^{(k)},\mathcal{Y}^{(k)}}\delta(C_{y\beta}^{(k)}, C_{x\alpha}^{(k)}, Y_1^{(k)})$.

From Lemma 4.2.5, part (v), we have

$$d_{\underline{T_{C_{y\beta}}}}(Y_1(d), Y_2(d))/\sqrt{d} = \sqrt{l_2^2 + l_{y\beta.2\delta}} + O_p(d^{-\frac{1}{2}}).$$

Therefore, using the resulting geometric representation of the data (see the proof of Lemma 4.1.7), we have

$$\begin{aligned} d_{\underline{T_{C_{y\beta}}}}(\underline{C_{x\alpha}}, \underline{C_{y\beta}})/\sqrt{d} &= [l_{12}^2 + l_{x\alpha.12\delta} - \frac{1}{2}(l_1^2 + l_{x\alpha.1\delta})(1 - \sum_{i=1}^{n} \alpha_i^2) + \\ &\quad + l_{y\beta.x\alpha 2\delta} - \frac{1}{2}(l_2^2 + l_{y\beta.2\delta})(1 - \sum_{i=1}^{m} \beta_i^2)]^{\frac{1}{2}} + O_p(d^{-\frac{1}{2}}). \end{aligned} \tag{4.92}$$

Therefore, noting that $d_{\underline{T_{C_{y\beta}}}}(\underline{C_{x\alpha}}, \underline{C_{y\beta}}) = d_{\underline{S^2}}(\underline{C_{x\alpha}}, \underline{C_{y\beta}})$ and using Eq. (4.92), we have

$$\begin{aligned} M_d(\underline{\theta}) &= -\frac{1}{d} d_{\underline{S^2}}^2(\underline{C_{x\alpha}}, \underline{C_{y\beta}}) \\ &\longrightarrow^P -[l_{12}^2 + l_{x\alpha.12\delta} - \frac{1}{2}(l_1^2 + l_{x\alpha.1\delta})(1 - \sum_{i=1}^{n} \alpha_i^2) + \\ &\quad + l_{y\beta.x\alpha 2\delta} - \frac{1}{2}(l_2^2 + l_{y\beta.2\delta})(1 - \sum_{i=1}^{m} \beta_i^2)]. \end{aligned}$$

This completes the proof. □

Proof of Proposition 4.2.9. We shall verify that the conditions of Theorem 4.4.1 holds true. First, it should be noted that $\underline{\theta}_0 = (\frac{1}{m}, \ldots, \frac{1}{m}, \frac{1}{n}, \ldots, \frac{1}{n})$ is the unique minimizer of $M(\underline{\theta})$ (defined in Eq. (4.83)). Therefore, we have the condition (4.90) is satisfied.

The condition (4.91) ensures that $\underline{\tilde{\theta}^d}$'s nearly maximize M_d's. By definition of $\underline{\tilde{\theta}^d} = [\underline{\tilde{\alpha}^d}, \underline{\tilde{\beta}^d}]$ in Eq. (4.78), we have that satisfied.

A sufficient set of conditions for Eq. (4.89) to hold true is given below (van der Vaart (1998)):

(a) Θ is compact.

(b) $m_{\underline{\theta}} : \Re^{m+n} \mapsto \Re$ is continuous.

(c) $m_{\underline{\theta}}$ is dominated by an integrable function.

We note that $\Theta = \{\underline{\alpha}, \underline{\beta} : \alpha_i, \beta_j \in [0,1] \text{and} \sum_{i=1}^{m} \alpha_i = \sum_{i=1}^{n} \beta_i = 1\}$. Therefore, Θ is compact and thus (a) holds.

When data lie in a small neighborhood, $d_{S^2}(\cdot,\cdot)$ is a continuous function. Noting that $\sum \alpha_i \text{Log}_{\underline{C_x}} X_i(d)$ and $\sum \beta_i \text{Log}_{\underline{C_y}} Y_i(d)$ are continuous functions of $\underline{\alpha}$ and $\underline{\beta}$ respectively, we can say that $m_{\underline{\theta}}$ is a continuous function of $\underline{\theta}$. Thus (b) holds.

Note that $|d_{S^2}(\cdot,\cdot)| \leq \pi$. Therefore, $m_{\underline{\theta}}$ is dominated by an integrable function and thus condition (c) holds.

All conditions for Theorem 4.4.1 has been verified and hence $\underline{\tilde{\theta}^d} \longrightarrow^P \underline{\theta}_0$. This completes the proof. □

CHAPTER 5

Discussion and Future Work

In this chapter, some avenues of future work is discussed, which involves unresolved questions and possible application of the developed methods in new areas.

5.1 Implementing MDWD

In Section 3.4.3, we extended the method of DWD for manifold data. The resulting optimization problem was given by Eq. (3.33). Our attempt to solve the optimization problem via a negative gradient descent approach (described in 3.3.3) failed. It seems a more sophisticated nonlinear optimization technique needs to be employed. Results in Section 3.4.4 suggest that the tangent plane methods, ITanDWD and TDWD work better than their SVM counterparts. Therefore, it will be of interest to implement MDWD by solving the optimization problem given by Eq. (3.33).

5.2 Role of the Parameter k in MSVM

In Section 3.3.3, we extended SVM to manifold data by presenting an optimization problem which minimizes the objective function given by

$$g_\lambda(c_1, c_{-1}) = d^2(c_1, c_{-1}) + \frac{\lambda}{n}\sum_{i=1}^{n}\Big[k - y_i\{d^2(x_i, c_{-1}) - d^2(x_i, c_1)\}\Big]_+,$$

where the parameter k is such that

$$\hat{y}_{c_1,c_{-1}} f_{c_1,c_{-1}}(\hat{x}_{(c_1,c_{-1})}) = k.$$

Here $\hat{x}_{c_1,c_{-1}}$ is the training point nearest to $H(c_1, c_{-1})$ and $\hat{y}_{c_1,c_{-1}}$ is its class label. It will be of interest to see how different choices of k and λ interact with each other.

5.3 Asymptotic Behavior of Manifold Data under Milder Conditions

In our study of the asymptotic behavior of data lying on $(S^2)^d$ (Section 4.2), we assumed that the entries in each of the dimensions are independent and identically distributed. This is a strong assumption. There is scope for relaxing these conditions. For example, the assumption about identically distributed entries can be relaxed if the *Lindeberg condition* is imposed on the moments of $d^2_{S^2}(X_1^{(k)}, X_2^{(k)})$ and $d^2_{S^2}(Y_1^{(k)}, Y_2^{(k)})$.

Moreover, it will be of interest to see how assumptions similar to those used by Hall *et al.* (2005) (described in Section 4.1) can be used to study the geometric structure of the data. This approach treats the entries of the vectors as a time series and requires them to be almost independent. However, these are much weaker assumptions than requiring the entries to be i.i.d. random variables.

It should be noted that in order to relax the i.i.d. condition, we will also need several additional assumptions on the error term $\delta()$ (defined in Eq. (4.38)). This is because our treatment of the asymptotic behavior involves studying the pairwise distances both on the manifold $(S^2)^d$ and on several tangent planes. Every time we use properties of the data on the manifold to study the behavior on a tangent plane (or vice versa), the variable $\delta()$ is used.

5.4 Application to DT-MRI

Diffusion tensor magnetic resonance imaging (DT-MRI) is emerging as an important tool in medical image analysis of the brain. DT-MRI, developed by Basser *et al.* (1994), measures the random 3D motion of water molecules, i.e., the diffusion of water. It produces a 3D diffusion tensor, i.e., a 3×3, symmetric, positive-definite matrix, at each voxel of a 3D imaging volume.

Fletcher and Joshi (2004) show that the space of diffusion tensors is a type of curved manifold known as a Riemannian symmetric space. They expanded the method of principal geodesic analysis to symmetric spaces and applied it to the computation of the variability of diffusion tensor data.

The classification methods proposed in this dissertation can be developed for DT-MRI data using the mathematical foundation due to Fletcher and Joshi (2004).

5.5 Generalizing Proposed Methods to Multiclass

In this study, the classification methods developed are applicable to data from just two classes. There is scope to extend these methods to multi-class situations (K being the number of classes). In general, instead of having two control points c_1 and c_{-1}, we will have a set $c = \{c_1, c_2, \ldots, c_K\}$ of control points, representing the K classes. Given a set of control points, a new datum x will be assigned to that class, whose corresponding control point is closest (in the geodesic sense) to x. In other words, the datum x will be assigned to class l_x if

$$l_x = \underset{l \in \{1,\ldots,K\}}{\operatorname{argmin}} \, d_M(c_l, x),$$

where $d_M(\cdot, \cdot)$ is the geodesic distance on the manifold M. The challenge lies in identifying the criteria which produces control points with desirable properties.

BIBLIOGRAPHY

Basser P.J., Mattiello J. and Le Bihan D. (1994). MR diffusion tensor spectroscopy and imaging. *Biophysics Journal* **66**, 259–267.

Benito M., J. P., Du Q., Wu J., Xiang D., Perou C. and Marron J.S. (2004). Adjustment of systematic microarray data biases. *Bioinformatics* **20**, 105–114.

Bhattacharya R. and Patrangenaru V. (2002). Nonparametric estimation of location and dispersion on riemannian manifolds. *Journal for Statistical Planning and Inference* **108**.

Bloomenthal J. and Shoemake K. (1991). Convolution surfaces. *Computer Graphics (SIGGRAPH 91 Proceedings)* **25**, 251–256.

Blum H. (1967). A transformation for extracting new descriptors of shape, In W.Wathen-Dunn, editor. *Models for the Perception of Speech and Visual Form. MIT Press, Cambridge MA* pp. 363–380.

Blum H. and Nagel R. (1978). Shape description using weighted symmetric axis features. *Pattern Recognition,* **10(3)**, 167180.

Bookstein F.L. (1978). The measurement of biological shape and shape change. *Number 24 in Lecture Notes in Biomathematics. Springer-Verlag* .

Bookstein F.L. (1986). Size and shape spaces for landmark data in two dimensions (with discussion). *Statistical Science* **1**, 181–242.

Boothby W.M. (1986). *An Introduction to Differentiable manifold and Riemannian Geometry*. New York: Academic.

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

Chikuse Y. (2003). *Statistics on Special Manifolds*. Lecture Notes in Statistics. Springer-Verlag New York, LLC.

Cootes T.F., Hill A., Taylor C.J. and Haslam J. (1993). In h. h. barrett and a. f. gmitro, editors. *Proceedings of Information Processing in Medical Imaging, volume of Lecture Notes in Computer Science, pages . Springer-Verlag* **687**, 3347.

Donoho D. and Tanner J. (2005). Neighborliness of randomly-projected simplices in high dimensions. *Proceedingsof the National Academy of Sciences of the United States of America* **102**, 9452–9457.

Duda R., Hart P.E. and Stork D.G. (2001). *Pattern Classification*. Wiley-Interscience.

Fisher R. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics* **7**, 179–188.

Fletcher P.T. and Joshi S. (2004). Principal geodesic analysis on symmetric spaces: statistics of diffusion tensors. *Lecture Notes in Computer Science; Springer-Verlag* **3117**, 87–98.

Fletcher P.T., Lu C. and Joshi. S. (2003). Statistics of shape via principal geodesic analysis on lie groups. *In Proceedings IEEE Conference on Computer Vision and Pattern Recognition* **1**, 95101.

Fletcher P.T., Lu C., Pizer S.M. and Joshi. S. (2004). Principal geodesic analysis for the study of nonlinear statistics of shape. *IEEE Transactions on Medical Imaging* **23**, 995–1005.

Fletcher T. (2004). Statistical variability in nonlinear spaces: Application to shape analysis and dt-mri. *PhD thesis, The University of North Carolina at Chapel Hill* .

Ge N. and Simpson D.G. (1998). Correlation and high-dimensional consistency in pattern recognition. *Journal of the American Statistical Association* **93**, 995–1006.

Grenander U. (1963). *Probabilities on Algebraic Structures*. John Wiley and Sons.

Hall P., Marron J.S. and Neeman A. (2005). Geometric representation of high dimension, low sample size data **67(3)**, 427–444.

Hampel F.R., Ronchetti E.M., Rousseeuw P.J. and Stahel W.A. (1986). *Robust Statistics*. Mathematics. Wiley.

Hastie T., Tibshirani R. and Friedman J. (2001). *The Elements of Statistical Learning*. New York: Springer.

Helgason S. (1978). *Differential Geometry, Lie Groups, and Symmetric Spaces*. New York: Academic.

Hotelling H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology* **24:417-441**, 498–520.

Huber P.J. (1981). *Robust Statistics*. Probability and Statistics. Wiley.

Hunt G.A. (1956). Semi-groups of measures on lie groups. *Transactions of the American Mathematical Society* **81**, 264–293.

Karcher H. (1977). Riemannian center of mass and mollifier smoothing. *Commun. Pure Appl. Math.* **30**, 509–541.

Kendall D.G. (1984). Shape manifolds, procrustean metrics, and complex projective spaces. *Bull. Lond. Math. Soc.* **16**, 18–121.

Klassen E., Srivastava A., Mio W. and Joshi S. (2004). Analysis of planar shapes using geodesic paths on shape space. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **26**, 372–383.

Lee T.S., Mumford D. and Schiller P.H. (1995). Neuronal correlates of boundary and medial axis representations in primate striate cortex. *In Investigative Ophthalmology and Visual Science* **36**, 477.

Leyton M. (1992). *Symmetry, Causality, Mind*. MIT Press, Cambridge, MA.

Mardia K.V. (1999). Directional statistics. *John Wiley and Sons,* .

Mardia K.V. and Dryden I.L. (1989). Shape distributions for landmark data. *Advances in Applied Probability* **21**, 742–755.

Marron J.S., Todd M. and Ahn J. (2004). Distance weighted discrimination. *Operations Research and Industrial Engineering, Cornell University, Technical Report 1339* .

Nackman L.R. and Pizer S.M. (1985). Three-dimensional shape description using the symmetric axis transform, I: theory. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **7(2)**, 187–202.

Olsen N.H. (2003). Morphology and optics of human embryos from light microscopy. *PhD thesis, University of Copenhagen, Denmark* .

Pearson K. (1901). On lines and planes of closest fit to points in space. *Philosophical Magazine* **2**, 609–629.

Pennec X. (1999). Probabilities and statistics on riemannian manifolds: Basic tools for geometric measurements. *In IEEE Workshop on Nonlinear Signal and Image Processing* .

Pizer S., Fritsch D., Yushkevich P., Johnson V. and Chaney E. (1999). Segmentation, registration, and measurement of shape variation via image object shape. *IEEE Transactions on Medical Imaging* **18**.

Pizer S.M., Fletcher P.T., Joshi S., Thall A., Chen J.Z., Fridman Y., Fritsch D.S., Gash A.G., Glotzer J.M., Jiroutek M.R., Lu C., Muller K.E., Tracton G., Yushkevich P. and Chaney E.L. (2003). Deformable m-reps for 3d medical image segmentation. *International Journal of Computer Vision* **55(2-3)**, 85–106.

Rao C.R. (1958). Some statistical methods for comparison of growth curves. *Biometrics* **14**, 1–17.

Ruymgaart F.H. (1989). Strong uniform convergence of density estimators on spheres. *J. Statist. Pl. Inf.* **23**, 45–52.

Ruymgaart F.H., Hendriks W. and Janssen H. (1992). A cramr-rao type inequality for random variables in euclidean manifolds. *Sankhya* **54**, 387–401.

Sch ö lkopf B. and Smola A. (2002). *Learning with Kernels.*. Cambridge: MIT Press.

Sen S.K., Foskey M., Marron J.S. and Styner M.A. (2008). Support vector machine for data on manifolds: An application to image analysis. *IEEE Symposium on Biomedical Imaging, ISBI* .

Siddiqi K. and Pizer S.M. (2007). *Medial Representations: Mathematics, Algorithms and Applications*. Springer.

Storti D.W., Turkiyyah G.M., Ganter M.A., Lim C.T. and Stat D.M. (1997). Skeletonbased modeling operations on solids. *In Proceedings of the Fourth Symposium on Solid Modeling and Applications (SSMA 97)* pp. 141–154.

Styner M., Lieberman J., Pantazis D. and Gerig G. (2004). Boundary and medial shape analysis of the hippocampus in schizophrenia. *MedIA* **197-203**.

Swann A. and Olsen N.H. (2003). Linear transformation groups and shape space. *Journal of Mathematical Imaging and Vision* **19**, 49–62.

Terriberry T.B. and Gerig G. (2006). A continuous 3-d medial shape model with branching. *Proc. of the International Workshop on Mathematical Foundations of Computational Anatomy (MFCA)* .

Thompson D. (1942). *On Growth and Form*. Cambridge University Press, second edition.

Toh K.C., Tutuncu R.H. and Todd M.J. (2006). Sdpt3 url. *www.math.nus.edu.sg/mattohkc/sdpt3.html* .

van der Vaart A.W. (1998). *Asymptotic Statistics*. Cambridge University Press.

Vapnik V. (1995). *The Nature of Statistical Learning Theory*. Springer-Verlag, New York.

Vapnik V., Golowich S. and Smola A. (1996). Support vector method for function approximation, regression estimation, and signal processing. *Advances in Neural Information Processing Systems* **9**, 281–287.

Vapnik V.N. (1982). *Estimation of Dependencies Based on Empirical Data*. Springer, New York.

Wehn D. (1959). Limit distributions on lie groups. *PhD thesis, Yale University* .

Wehn D. (1962). Probabilities on lie groups. *Proceedings of the National Academy of Sciences of the United States of America* **48**, 791–795.

Yushkevich P. (2003). Statistical shape characterization using the medial representation. *PhD thesis, The University of North Carolina at Chapel Hill* .

Yushkevich P., Fletcher P.T., Joshi S., Thall A. and Pizer. S.M. (2003). Continuous medial representations for geometric object modeling in 2d and 3d. *Image and Vision Computing* **21**, 17–28.