Universidad
Carlos III de Madrid

Universidad Carlos III de Madrid

PH.D. THESIS

Representing Functional Data in Reproducing Kernel Hilbert Spaces with Applications to Clustering, Classification and Time Series Problems

Author:
Javier González Hernández

Advisor:
Alberto Muñoz García

DEPARTMENT OF STATISTICS

Getafe, Madrid, May, 2010

A mis padres Juan y Clara.

Agradecimientos

Quiero comenzar mostrando mi agradecimiento a mi director y amigo Alberto Muñoz. Quiero agradecer a Alberto su tiempo, paciencia y dedicación durante estos años en los que ha conseguido guiarme hasta aquí. Si como director tengo muchísimo que agradecer a Alberto, en tanto o más estoy en deuda con él por cada uno de sus útiles consejos que me han ayudado en todos los aspectos de mi vida. Ha sido un placer aprender contigo algo nuevo cada día Alberto. Muchísimas gracias por todo.

Este trabajo no hubiera sido posible sin el apoyo y la financiación que el departamento de Estadística de la Universidad Carlos III de Madrid me ha proporcionado en todo momento. Siempre he tenido la suerte de contar con un buen lugar en el que trabajar y con todos los medios que he necesitado. En particular, quiero agradecer muy especialmente al proyecto SEJ2004-03303 por la beca BES-2005-7130 adscrita y financiada por el Ministerio de Ciencia e Innovación que pude disfrutar entre los años 2005 y 2009. Quiero agradecer también a Peter Jacko el compartir con el resto de ayudantes del departamento la plantilla de Latex con la que esta tesis ha sido escrita.

Gracias también and Fréderic Ferraty y Philippe Vieu por hacer públicos los datos y el código de sus técnicas para el análisis de datos funcionales. Han sido una gran ayuda en la sección experimental de esta tesis.

Me gustaría agradecer al Prof. Roland Fried su ayuda durante el verano de 2006 que pasé en el Departamento de Estadística de la Universidad de Dortmund. Fue toda una suerte poder trabajar en Dortmund y conocer a su grupo de investigación.

Mi más sincero agradecimiento al grupo Image del Centro Nacional de Investigaciones Atmosféricas de Estados Unidos. Pasar los veranos de 2007 y 2008 en Boulder, Colorado, fue una experiencia inolvidable. Quiero agradecer especialmente al Prof. Stephan R. Sain toda su dedicación y ayuda para que me sintiera uno más de su grupo de trabajo. Agradecer también al Prof. Luis Tenorio de la Escuela de Minas de Colorado la ayuda que me prestó en todo momento.

Con todo el cariño del mundo quiero agradecer el apoyo y paciencia de mi familia; mis padres Juan y Clara, mi hermana María y mi abuela María. Doscientos kilómetros no son una gran distancia pero son los suficientes como para extrañar a la gente que uno quiere.

Da gusto reencontrarse en cada viaje a Salamanca con amigos como Ángel y Berme. Gracias a los dos por los buenos ratos y por los que quedan por venir.

En todo estos años en Madrid he tenido la oportunidad de conocer a un montón de gente estupenda. Del periodo que pasé en la residencia universitaria Fernando de los Ríos recuerdo con especial cariño a mis amigos Nacho, Claudio, Roi, Fran, Marta, Iago, Julio, Inma y Luismi. No me puedo olvidar aquí de alguien único como Kenedy que con su sola presencia te hace sentir que el día ha merecido la pena.

Para terminar quiero agradecer a mi novia Patricia por los casi dos últimos años que hemos pasado juntos. Gracias por tu apoyo, paciencia y por hacerme sentir tan afortunado al poder compartir tanto contigo. Es un placer estar a tu lado cada día.

Gracias a todos,

Javier.

The best understanding of what one can see
comes from theories of what one can't see.
S. Smale and D. X. Zhou

Abstract

In modern data analysis areas such as Image Analysis, Chemometrics or Information Retrieval the raw data are often complex and their representation in Euclidean spaces is not straightforward. However most statistical data analysis techniques are designed to deal with points in Euclidean spaces and hence a representation of the data in some Euclidean coordinate system is always required as a previous step to apply multivariate analysis techniques. This process is crucial to guarantee the success of the data analysis methodologies and will be a core contribution of this thesis.

In this work we will develop general data representation techniques in the framework of Functional Data Analysis (FDA) for classification and clustering problems. In Chapter 1 we motivate the problems to solve, describe the roadmap of the contributions and set up the notation of this work.

In Chapter 2 we review some aspects concerning Reproducing Kernel Hilbert Spaces (RKHSs), Regularization Theory Integral Operators, Support Vector Machines and Kernel Combinations.

In Chapter 3 we propose a new methodology to obtain finite-dimensional representations of functional data. The key idea is to consider each functional curve as a point in a general function space and then project these points onto a Reproducing Kernel Hilbert Space (RKHS) with the aid of Regularization theory. We will describe the projection methods, analyze its theoretical properties and develop an strategy to select appropriate RKHSs to represent the functional data.

Following the functional data analysis approach, we develop in Chapter 4 a new procedure to deal with proximity (similarity or distance) matrices in classification problems by studying the connection between proximity measures and a certain class of integral operators. The idea is to come up with a methodology able to estimate an integral operator whose associated kernel function, evaluated at the sample, approximates the sample proximity matrix of the problem. To show the broad scope of application of the

methodology, we will apply it to three cases: (1) classification problems where the only available information about the data is an asymmetric similarity matrix (2) partially labeled classification problems and (3) classification problems where several sources of information are available and can be combined to obtain the discrimination function.

In Chapter 5 we propose an spectral framework for information fusion when the sources of information are given by a set of proximity matrices. Our approach is based on the simultaneous diagonalization of the original matrices of the problem and it represents a natural way to manage the redundant information involved in the fusion process. In particular, we define a new metric for proximity matrices and we propose a method that automatically eliminates the redundant information among a set of matrices when they are combined.

We conclude the contributions of the thesis in Chapter 6 with a battery of simulated and real examples devoted to compare the performance of the proposed methodologies with the state of the art in representation methods. Finally, in Chapter 7 we include a discussion regarding the topics described above and we propose some future lines of research we believe are the natural extensions to the work developed in this thesis.

Resumen

En áreas de análisis de datos tales como el Análisis de Imágenes, la Quimiometría o la Recuperación de Información los datos son complejos y su representación en espacios Euclídeos no es directa. Sin embargo, la mayoría de los procedimientos estadísticos están diseñados para trabajar con puntos en espacios Euclídeos. Por tanto, representar los datos en un sistema Euclídeo de coordenadas es el paso previo necesario al uso de técnicas estadísticas multivariantes. Este proceso es crucial a la hora de garantizar adecuadas soluciones a nuestros problemas y será el núcleo central de las contribuciones de esta tesis.

En este trabajo desarrollaremos técnicas generales de representación de datos en problemas de clasificación y conglomerados en el marco del Análisis Funcional de Datos. En el Capítulo 1 motivaremos los problemas a resolver, describiremos las contribuciones y fijaremos la notación utilizada en este trabajo.

En el Capítulo 2 revisamos algunos aspectos relacionados con los espacios de Hilbert de Núcleo reproductivo, la Teoría de Regularización, Operadores integrales, Máquinas de Vectores Soporte y métodos de Combinaciones de Núcleos.

En el Capítulo 3, proponemos una nueva metodología para obtener representaciones de dimensión finita de datos funcionales. La idea clave es considerar cada dato funcional como un punto en un espacio general de funciones y posteriormente proyectar estos puntos en un espacio de Hilbert de Núcleo Reproductivo con la ayuda de la teoría de Regularización. En el Capítulo 3 describiremos el método de proyección, analizaremos sus propiedades teóricas y desarrollaremos una estrategia para seleccionar un espacio apropiado en el que representar los datos funcionales.

Siguiendo el enfoque de análisis de datos funcionales, desarrollamos en el Capítulo 4 un nuevo procedimiento para trabajar con matrices de proximidades (similaridades o distancias) en problemas de clasificación y conglomerados estudiando la relación entre matrices de proximidad y cierta clase de operadores integrales. La idea es desarrollar una metodología capaz de estimar un operador integral cuya núcleo, evaluado

en la muestra, aproxime la matriz de proximidad. Para mostrar la utilidad de la meteodología propuesta la aplicaremos en tres casos: (1) problemas de clasificación donde la información disponible sobre los datos es una matriz de similaridades asimétrica, (2) problemas de clasificación parcialmente etiquetados y (3) problemas de clasificación donde varias fuentes de infomación están disponibles y pueden ser combinadas para obtener el clasificador.

En el Capítulo 5 proponemos un marco espectral para la fusión de infomación cuando las fuentes de información vienen dadas por un conjunto de matrices de proximidades. Nuestro enfoque está basado en la diagonalización simultánea de dichas matrices y representa un modo natural de tratar con la infomación redundante involucrada en el proceso de combinación. En particular, definiremos una nueva métrica para matrices de proximidades y propondremos un método que elimina automáticamente la infomación redundante de una serie de matrices cuando son combinadas.

Concluimos las contribuciones de esta tesis en el Capítulo 6 con una batería experimentos reales y simulados cuyo objetivo es comparar la metodología propuesta con el estado de arte en métodos de representatión de objetos. Finalmente, en el Capítulo 7 incluimos una discusión sobre los temas tratados en anteriormente y futuras líneas de investigación que creemos son la prolongación natural de las contribuciones de esta tesis.

Contents

List of Figures

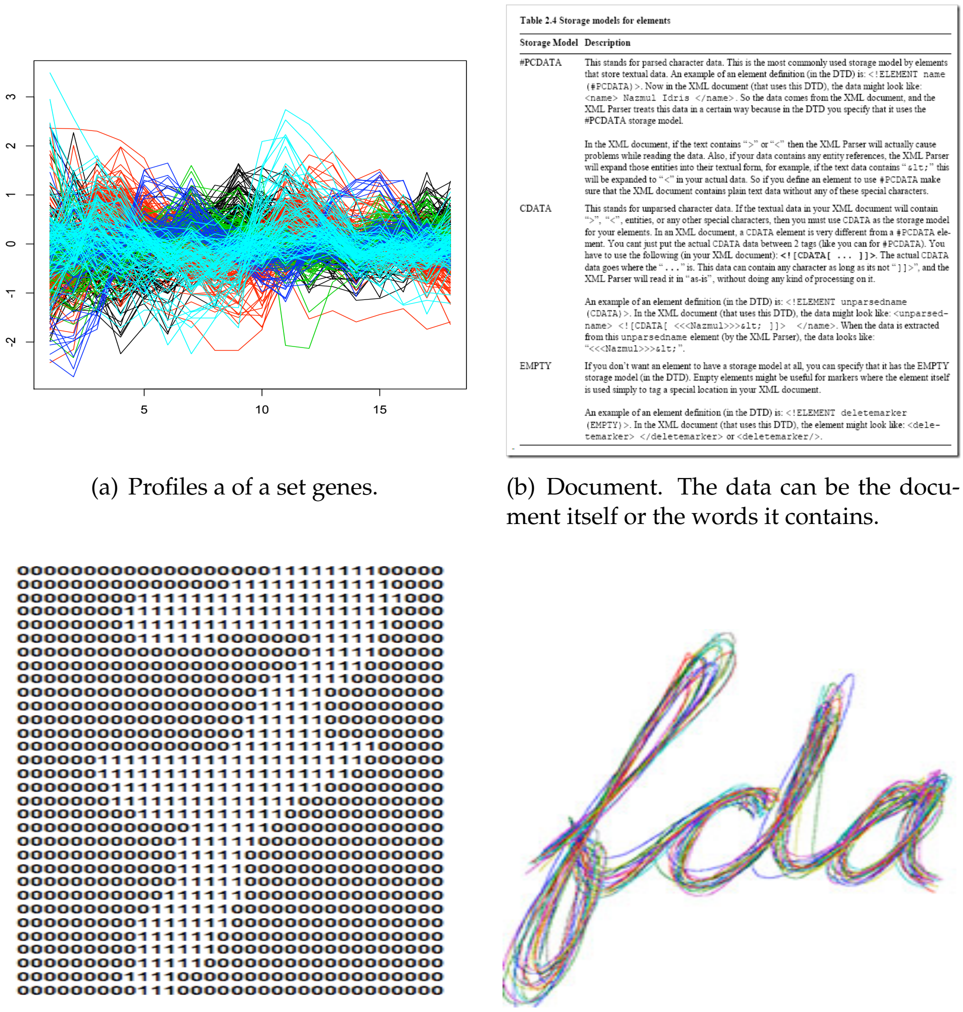
List of Tables

Chapter 1

Introduction

In modern data analysis problems the raw data are often complex and their representation in Euclidean spaces is not straightforward. For instance in Genetics the data are usually time series of cDNA micro-arrays gathered over a set of equally spaced time points (Spellman et al., 1998) (see for instance Figure 1 a)). In Textual Analysis, the data are collections of documents or web pages usually labeled by topic (Lang, 1995) (see Figure 1 b)). In problems of automatic handwriting identification the objects can be given a images of digits or characters. Such images are treated as matrices (as in Figure 1 c)) whose components play the role of the pixels of the original pictures (Frey and Slate, 1991). Another point of view in automatic handwriting identification is to process writing sequence of characters (see Figure 1 d)) as curves that reflect the horizontal and vertical position of the pen position for a set of time measurements (Ramsay and Silverman, 2006).

While in the previous fields in not straightforward to represent the data in Euclidean spaces, statistical procedures are designed to deal only with points in Euclidean coordinate systems. Therefore the data must be embedded in some Euclidean space as a previous step to apply any multivariate analysis technique.



(a) Profiles a of a set genes.

Table 2.4 Storage models for elements

Storage Model	Description
#PCDATA	This stands for parsed character data. This is the most commonly used storage model by elements that store textual data. An example of an element definition (in the DTD) is: <!ELEMENT name (#PCDATA)>. Now in the XML document (that uses this DTD), the data might look like: <name> Nazmul Idris </name>. So the data comes from the XML document, and the XML Parser treats this data in a certain way because in the DTD you specify that it uses the #PCDATA storage model. In the XML document, if the text contains ">" or "<" then the XML Parser will actually cause problems while reading the data. Also, if your data contains any entity references, the XML Parser will expand those entities into their textual form, for example, if the text data contains "&lt;" this will be expanded to "<" in your actual data. So if you define an element to use #PCDATA make sure that the XML document contains plain text data without any of these special characters.
CDATA	This stands for unparsed character data. If the textual data in your XML document will contain ">", "<", entities, or any other special characters, then you must use CDATA as the storage model for your elements. In an XML document, a CDATA element is very different from a #PCDATA element. You cant just put the actual CDATA data between 2 tags (like you can for #PCDATA). You have to use the following (in your XML document): <![CDATA[...]]>. The actual CDATA data goes where the "..." is. This data can contain any character as long as its not "]]>", and the XML Parser will read it in "as-is", without doing any kind of processing on it. An example of an element definition (in the DTD) is: <!ELEMENT unparsedname (CDATA)>. In the XML document (that uses this DTD), the data might look like: <unparsedname> <![CDATA[<<<Nazmul>>>&lt;]]> </name>. When the data is extracted from this unparsedname element (by the XML Parser), the data looks like: "<<<Nazmul>>>&lt;".
EMPTY	If you don't want an element to have a storage model at all, you can specify that it has the EMPTY storage model (in the DTD). Empty elements might be useful for markers where the element itself is used simply to tag a special location in your XML document. An example of an element definition (in the DTD) is: <!ELEMENT deletemarker (EMPTY)>. In the XML document (that uses this DTD), the element might look like: <deletemarker> </deletemarker> or <deletemarker/>.

(b) Document. The data can be the document itself or the words it contains.

```
00000000000000000001111111100000
00000000000000001111111111110000
00000000011111111111111111111000
00000000111111111111111111110000
00000000111111111111111111110000
00000000011111100000001111100000
00000000000000000000001111100000
00000000000000000000011111000000
00000000000000000001111110000000
00000000000000000011111000000000
00000000000000000011111000000000
00000000000000000111111000000000
00000000000000000111111000000000
00000000000000001111111111100000
00000011111111111111111111000000
00000011111111111111111110000000
00000001111111111111111000000000
00000001111111111111100000000000
00000000011111111111000000000000
00000000000001111110000000000000
00000000000011111100000000000000
00000000000011111000000000000000
00000000000011111000000000000000
00000000000011111000000000000000
00000000000111111000000000000000
00000000001111110000000000000000
00000000011111110000000000000000
00000000011111100000000000000000
00000000011111100000000000000000
00000000011111000000000000000000
00000000111100000000000000000000
00000000011100000000000000000000
```

(c) Binary matrix representing a digit.

(d) Writing sequences of the chain of characters "fda".

Figure 1.1: Four real examples of data with different nature.

In fields like Genetic Data Analysis, Control Quality or Chemometrics the data have very high (or intrinsically infinite) dimensionality and they are very difficult to manage for many traditional statistical techniques. The reason is that the data are functions and most Multivariate Analysis algorithms are designed to work with vectors. In this cases, the data must be analyzed in a Functional Data Analysis (FDA) context (Ferraty and Vieu, 2006; Ramsay and Silverman, 2006). The key idea of FDA is to represent each curve as a point in some space of functions $B = \{\phi_1, \dots \phi_d\}$ where $d \in \mathbb{N}$ and each ϕ_j explains some feature of the curves of the problem.

In other applications the objects can be images, shapes of objects in 3D, points on a manifold, tree structured data, etc. In such cases, a generalized approach to represent the objects can be done when a proximity (similarity or disimilarity) matrix is available. In this cases it is always possible to estimate Euclidean coordinates for the data via some Multidimensional Scaling (MDS) procedure (Cox and Cox, 2001).

In all the previous approaches, to find a "good" representation system means to obtain a coordinate system where the distance between two objects, viewed as points, reflects the "right" notion of dissimilarity between them. Two similar objects (for instance, two similar genes profiles (see Figure 1 a)) should be represented by two points that are close together, and two dissimilar objects (two different genes profiles) should be represented by two points that are far apart. When the data set is labeled (classification problems) the underlying hypothesis for a good representation is that all the points in an small enough neighborhood must belong to the same class (have the same labels) (Martín de Diego et al., 2009).

So far, the literature has already considered a wide variety of representations methods. Common examples are Principal Components Analysis (PCA) (Jolliffe, 2002), Partial Least Squares (PLS) (de Jong, 1993) or Independent components Analysis (ICA) (Hyvärinen et al., 2001). These procedures work well in some cases but their success depends on the problem at hand. The main goal of this thesis is to develop a general data representation techniques in classification and clustering problems that takes into account the properties described above. To achieve this task we will develop a methodology capable to solve problems where the available data have some of the following characteristics:

i) *The objects are naturally represented by functions.*

We are mainly interested in problems involving functional data. That is problems where each datum is given by a large vector where some covariance structure among the variables exists. This includes chemometrics data, time series, genetic data profiles etc.

ii) *The information about the data set is a proximity matrix.*

In some cases the only available information about the objects is a proximity matrix. We are specially interested problems where such proximity matrix is not symmetric (Martín-Merino and Muñoz, 2005). This happens, for instance, when dealing with similarities in genetic or textual data analysis.

iii) *Classification problems with partially labeled data.*

We are also interested in discrimination problems where the training data set consists of some labeled points and the remaining unlabeled (Chapelle et al., 2006). The idea is to use the structure of the unlabeled data to define a mapping that acts over the labeled data and that helps to improve the final classification performance.

iv) *Several information sources are available for the objects.*

In the last years, increasing interest has focused in the development of statistical techniques able to combine several sources of information to solve the problem at hand. In this context, we will pay an special attention to classification problems where two or more proximity matrices are available (Martín de Diego et al., 2009; Lanckriet et al., 2004; Kittler et al., 1998). For instance, when a set of web pages has to be classified we generally can make use of two different matrices to discriminate the web pages: the co-citation matrix and the terms by document matrix. In cases like this, such matrices convey complementary information and they should be combined. In this thesis we are interested in the combination process itself and in the problems derived from the bad use of the common information of the matrices.

Next, we give an overview of the thesis including its original contributions. In Section 1.2 we set up the notation of this work.

1.1 Overview of the thesis and roadmap of the contributions

This thesis is divided in six chapters. Chapter 2 we review some aspects concerning Reproducing Kernel Hilbert Spaces (RKHSs), Regularization Theory Integral Operators, Support Vector Machines and Kernel Combinations.

The theoretical contributions of this thesis are developed in Chapters 3, 4 and 5. In Chapter 3 we propose a new methodology to obtain finite-dimensional representations of functional data. We will describe the proposed method, analyze its theoretical properties and develop an strategy to select RKHSs appropriate to represent the functional data.

Following the functional data analysis approach, we develop in Chapter 4 a new procedure to deal with proximity (similarity or distance) matrices in classification problems by studying the connection between proximity measures and a certain class of integral operators. To show the broad scope of application of the methodology, we will apply it in three cases: classification problems where the only available information about the data is an asymmetric similarity matrix, in partially labeled classification problems and in classification problems where several sources of information must be combined to obtain the discrimination function.

In Chapter 5 we propose an spectral framework for information fusion when the sources of information are given by a set of proximity matrices. In particular, we define a new metric for proximity matrices and we propose a method that automatically eliminate the redundant information among a set of matrices when they are combined.

Since the nature of the problem to study is quite data-oriented we have included a chapter specifically devoted to experimentation. In Chapter 6 we include simulated experiments to check the performance of the proposed techniques in controlled situations. In addition we compare our proposals with the state of the art in data representation techniques in a wide range of real examples.

In Chapter 7 we show some general conclusions and motivate some future work. Next we describe roadmap the most significant contributions of this thesis.

Our contributions start in Chapter 3 which concerns on the the problem of representing functional data in classification and cluster problems. The two main contributions of this chapter are:

(1) Representation systems for functional data
Problem: Most FDA approaches choose an orthogonal basis of functions $B = \{\phi_1, \ldots \phi_d\}$ ($d \in \mathbb{N}$), where each ϕ_j belongs to a general function space (usually $L^2(X)$) and then represent each functional datum by means of a linear combination in $Span(B)$. Different choices of B induce different distance for the curves (viewed as points in B) and hence a good election of the basis of functions where represent the curves is crucial to guarantee the success of classification and clustering techniques.

Contribution: We propose new techniques to obtain finite-dimensional representations of functional data. The idea is to consider each functional data as a point in a general function space and then project these points onto a Reproducing Kernel Hilbert Space (RKHS) with the aid of Regularization theory. In Chapter 3 we will show how to implement this idea in practical cases and we will derive some geometrical and statistical properties of the previous projection method.

(2) Model selection criteria for functional data
Problem: In (Sugiyama and Ogawa, 2001) the Subspace information Criterion (SIC) is proposed as a methodology for model selection in general regularization methods. As we will study in Chapter 3 it represents the natural way to select the RKHS to represent

the functional data in problem (1). However, as we will show in Section 3.4, the SIC fail choosing the best RKHS is some relevant problems.

Contribution: We propose the Modified Subspace Information Criteria (MSIC), as an alternative to the SIC in the context of Statistical Learning Theory (Vapnik, 1995). We show the theoretical and practical benefits of the new criterion in Section 3.4.3.

The next stop is Chapter 4 where we propose a Functional Data Analysis (FDA) approach to deal with proximity (similarity or distance) matrices in classification problems. In particular we study the precise connection between proximity matrices and certain class of self-adjoint, positive, compact integral operators and we apply the previous idea to solve the following problems.

(3) Classification Problems with partially labeled data: Classification with partially labeled data (Chapelle et al., 2006) are a class of classification problems where the data consist of some labeled points and the remaining unlabeled. The objective is to use the structure of the unlabeled data to improve the classification of test points.

Contribution: We propose a methodology to solve partially data classification problems. First, we define a particular similarity matrix that take into account the geometrical information of both, labeled and unlabeled data. Second, we estimate a particular integral operator associated to such matrix to extend its components for out-of-sample points.

(4) Classification problems with asymmetric information: In some classification problems the available proximity matrix between the objects is asymmetric (Martín-Merino and Muñoz, 2005). In this cases there is no immediate way to obtain Euclidean coordinates and thus apply standard classification procedures.

Contribution: We propose a methodology to estimate an integral operator whose eigenfunctions define a data embedding induced by the original asymmetric proximity matrix. This will allow to map the data into a Euclidean space as a previous step to the use of any classification algorithm.

(5) Combination of multiple proximity matrices in classification problems: In classification problems, the best alternatives to combine a set of proximity matrices make use of the labels of the training points (Martín de Diego et al., 2009; Lanckriet et al., 2004; Kittler et al., 1998). This strategy, that work well in real cases presents two serious drawbacks: The final combination matrix it is not necessarily positive definite and its components for out-of-sample points are unknown.

Contribution: We propose a methodology to solve the two problems described above within the integral operators framework. The idea is, first, to project the matrix of the combination onto the cone of positive definite matrices. Finally, use this projected matrix to estimate an integral operator able to extend the matrix combination component for out-of-sample points.

The last stop is in Chapter 5 where we focus on the analysis the redundant information in proximity matrices combination procedures. Next we detail the contributions of this chapter:

(6) Metrics for proximity matrices

Problem: With the emergence of data fusion techniques, the task of comparing proximity (distance/similarity/kernel) matrices is becoming increasingly relevant (Martín de Diego and Muñoz, 2006; Cristianini and Shawe-Taylor, 2002). The choice of appropriate metrics for matrices involved in classification or clustering problems is far from trivial. However it is crucial to guarantee the success in proximity combinations procedures.

Contribution: We propose a general spectral framework to build metrics for matrix spaces. It can be used to reinterpret the most common metrics in the literature and allows to define some new alternatives. In particular, we propose the Pencil Dissimilarity which is proven to work well in real situations.

(7) Redundancies in Information Fusion Techniques

Problem: Any technique developed to combine several sources of information should take into account the redundant information of the system (Muñoz and González, 2008). To illustrate this issue, consider a data set with three variables, and two data representations given by two projections on two pairs of principal axes: (x, y) and (x, z), where the x variable is present in both representations. If we use the direct sum of the corresponding spaces as solution for the combination problem, we will have the representation (x, y, x, z). Thus, the weight of the x variable will be doubled when using the Euclidean distance and the results of the classification or regression algorithms will be distorted.

Contribution: When the sources of information of the problem are given by a set of proximity matrices the direct sum of the matrices common in kernel combinations is affected by the problem described above. We propose a new technique for information fusion based on the Joint Diagonalization of matrices able to produce a new data representation in a Euclidean space eliminating the redundant information among the input matrices.

1.2 Some notation and conventions

In this section we introduce some basic notational conventions. Throughout this work vectors are denoted by lowercase bold letters and matrices by bold uppercase letters. Individual entries of vectors are denoted by the corresponding subindex while for matrices we use parenthesis. For instance, $\mathbf{x} \in \mathbb{R}^n$ is a vector whose coefficients are given by x_i and the matrix $\mathbf{A}$ has entries $(\mathbf{A})_{ij}$. Vector and matrix transpose is denoted by $\mathbf{x}^T$.

Given a matrix $\mathbf{A}$ of dimensions $n \times n$ we will denote its eigenvalues and eigenvectors by l_j and $\mathbf{v}_j$ respectively, ard we will always assume that the eigenvalues are sorted in non-decreasing order, that is $l_1 \leq l_2 \leq \cdots \leq l_n$.

We use the standard norms on finite dimensional vector spaces. Let $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{A} \in \mathbb{R}^{n \times n}$. Then

$$\|x\| = \sqrt{\sum_{j=1}^{n} x_i^2} \quad and \quad \|\mathbf{A}\|_F = \sqrt{\sum_{i,j=1}^{n} (\mathbf{A})_{ij}^2} \tag{1.1}$$

where $\|\cdot\|_F$ indicates the Frobenius norm.

In this thesis we will consider subsets of $\mathbb{R}^n$ as probability spaces and we will quietly assume the existence of an associated Borel-algebra. Let X be a compact subset of $\mathbb{R}^n$ and ν a Borel probability measure defined on X. Then if ξ is a random variable on X, we will denote as $\mathsf{E}_\nu(\xi)$ the expected value of of ξ and $\mathsf{Var}_\nu(\xi)$ its variance with respect to the measure ν.

In Table 1.2 we summarize the most relevant symbol used in this thesis and a brief description.

Table 1.1: Summary table of the symbols used in this thesis.

Symbol	**Description**
X	Compact space or manifold.
Y	Space of label samples (classification) or $Y = \mathbb{R}$ (regression).
$C(X)$	Banach space functions on X.
$H(X)$	Hilbert space of functions on X.
$\nu(X)$	Borel measure on X.
P	Probability.
E_ν	Expectation over the measure ν.
Var_ν	Variance over the measure ν
n	Sample size.
L	Loss function.
s_n	Random sample of n observations.
$x_1, \ldots, x_n$	Object samples.
$\mathbf{X}$	Matrix whose columns are the object samples
$\mathbf{x}$	Sample vector $(x_1, \ldots, x_n)^T$.
$y_1, \ldots, y_n$	Label samples.
$\mathbf{y}$	Labels vector $(y_1, \ldots, y_n)^T$.
$f_\mathbf{x}$	Vector $f_\mathbf{x}$ = $(f(x_1),\ldots, f(x_n))^T$.
$R_\nu(f)$	Generalization error of f.
$R_{s_n}(f)$	Empirical error of f for the sample s_n.
$\mathcal{H}$	Hypothesis space.
$f_\mathcal{H}$	Target function.
$f_{s_n,\mathcal{H}}$	Empirical target function.
K	Kernel function.
L_K	Integral operator (associated to K).
$\mathcal{H}_K$	Reproducing Kernel Hilbert Space of kernel K.
R_γ	Regularized Generalization Risk.
R_{γ,s_n}	Regularized Empirical Risk.
$\mathbf{S}$	Similarity or proximity matrix.
$\mathbf{D}$	Distance matrix.
$K\big\rvert_\mathbf{x}$	Kernel matrix where $(K\big\rvert_\mathbf{x})_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$.
$\mathbf{K_x}$	Kernel matrix where $(\mathbf{K_x})_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$.
λ_j	j-th eigenvalue of the integral operator L_K.
ϕ_j	j-th eigenfunction of the integral operator L_K..
$\boldsymbol{\phi}_{j,\mathbf{x}}$	Vector $\boldsymbol{\phi}_{j,\mathbf{x}}$ = $(\phi_j(x_1),\ldots, \phi_j(x_n))^T$.
l_j	j-th eigenvalue of the kernel matrix $K\big\rvert_\mathbf{x}$ (or $\mathbf{K_x}$).
$\mathbf{v}_j$	j-th eigenvector of the kernel matrix $K\big\rvert_\mathbf{x}$ (or $\mathbf{K_x}$).
$rank(\mathbf{A})$	Rank of the matrix $\mathbf{A}$.

Chapter 2

Background

Abstract

In this chapter we give a general overview of the theoretical background we will need in the sequel. We will explain with some level of detail the theory concerning Integral Operators, Reproducing Kernel Hilbert Spaces, Regularization Theory, Support Vector Machines and Kernel Combination methods since a good understanding of these topics is essential for the development of the techniques proposed in this thesis.

Keywords: Statistical Learning Theory, Inverse Problems, Reproducing Kernel Hilbert Spaces, Hypothesis Space, Information Fusion.

2.1 Introduction

The main objective of many statistical techniques is to learn some function from data samples (generally perturbed with some noise). To this aim, we need to make use of a variety of subjects. First, Statistical Inference whose purpose is precisely to infer information from random samples. Approximation theory will also play a crucial role to choose appropriate functions from data samples in some functions spaces. Finally some algorithmic considerations have to be done in the learning process. The estimation of the function will always be the output of some algorithmic procedure and its efficiency and stability are crucial in practice.

Our aim in this chapter is to give a theoretical overview of the statistical learning process blending ideas from the three previous fields and emphasizing the relationship of the learning process with the mainstream of some mathematical theories in the core of the contributions of this work. We start by showing some general learning examples.

Example 2.1 (Classification). Consider the task of automatic digit identification where the elements of the problem are: X, the space of binary matrices that represent such digits (see Figure 1 c)) and Y the labels (name) of the digits. The learning problem to solve here is the approximation of a function $f : X \to Y$ that relates each digit $\mathbf{x} \in X$ with its corresponding label $y \in Y$.

• • •

Example 2.2 (Regression). Another classical learning problem is the approximation of a surface by estimating a function $f : X \to \mathbb{R}$ that fits the best a set of n data points $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ where $\mathbf{x}_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}$.

• • •

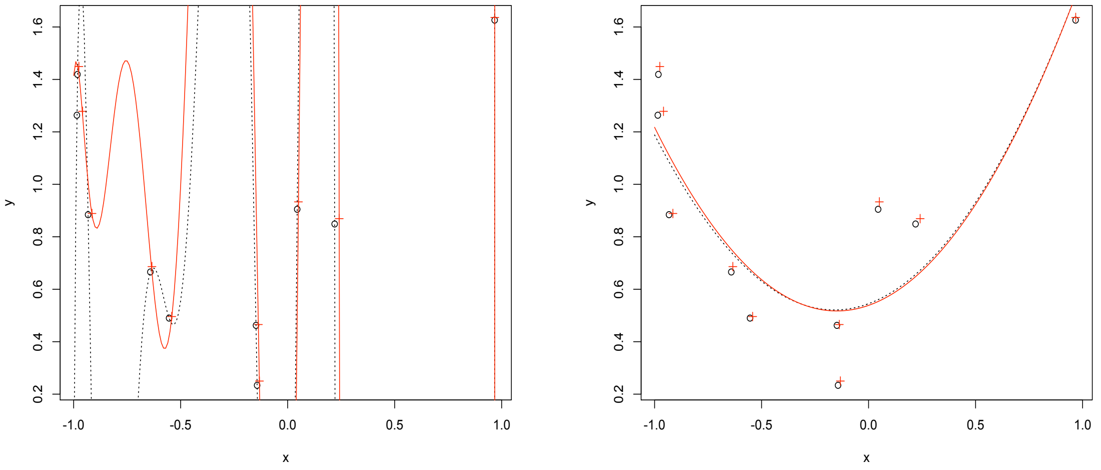
Example 2.3 (Level sets estimation). Another example of learning is the task of estimating high density regions from data samples (Muñoz and Moguerza, 2006). Assume $\mathbf{x}$ is a random variable a with density function $p(\mathbf{x})$ defined on $\mathbb{R}^d$. Let $\mathbf{x}_1, \dots, \mathbf{x}_n$ a set of independent identically distributed (iid) samples of size n (drawn from p). The problem is stated as the estimation of minimum volume sets of the form $S_\alpha(p) = \{\mathbf{x} : p(\mathbf{x}) \geq \alpha\}$, such that $P(S_\alpha(p)) = \mu$, where $0 < \mu < 1$. The learning task is to find $f(\mathbf{x})$ such that $f(\mathbf{x}_i) = 1$ if $\mathbf{x} \in S_\alpha(f)$ and $f(\mathbf{x}) = -1$ otherwise.

• • •

These three situations constitute examples of inverse problems (Tikhonov and Arsenin, 1977). Inverse problems occur in many branches of science and mathematics where the values of some model parameters have to be estimated from the observed data. Inverse problems are usually stated as solving $Af = y$ where A is generally a linear operator and both, f (the target of the problem) and y are functions that belong to some metric space. For instance, in multiple regression problems (Example 2.2 when the solution is restricted to be linear), $f \in \mathbb{R}^{d+1}$ can be identified with the regression vector parameters, A is the linear operator induced by the data matrix of dimension $n \times (d+1)$ and $y = (y_1, \dots, y_n)^T \in \mathbb{R}^n$ is the vector of response variables.

Inverse problems are hard to solve. Different values of the model parameters may be consistent with the sample data and to estimate them may require the exploration of a huge parameter space (Tarantola, 2005). Moreover inverse problems represent a broad cast of ill-posed problems (O'Sullivan, 1986).

Definition 2.1. *A problem is well-posed in the sense of Hadamard if its solutions is exists, is unique and depends continuously on the data sample.*



(a) Smoothest interpolating polynomial of degree 10 for the original data and their perturbations.

(b) Interpolating polynomial of degree 2 for the original data and their perturbations.

Figure 2.1: Example of problem originally ill-posed and its tranfomation to well posed by restringing the interpolation function to the family of polynomials of degree 2.

A problem is ill-posed if some of the previous conditions is not satisfied. For instance in classification, regression or density estimation, a parametric approach to find the target function is not always possible. In the three examples above, f requires an infinite number of observations to be perfectly described while the sample size is always finite. Hence, to obtain well posed-ness necessarily implies to seek f from a finite family of models. We illustrate this in the following example.

Example 2.4. Consider 10 data points in $\mathbb{R}^2$. We can always find a smoothest interpolating polynomial of high degree (10 in this case) as as it is done in Figure 2.1a) (circles and dotted curve). However if we slightly perturb the points (red crosses) the solution of the interpolation problem changes a lot. Therefore the problem is ill-posed since the solution is not stable on the input data. Nevertheless, if we force the interpolation function to be a polynomial of degree 2, the solution only varies a small amount when the data are perturbed (see Figure 2.1 b)) and the problem is now well-posed.

• • •

In this chapter we will detail a theoretical framework to achieve well posedness in a broad cast of learning problems. To this aim we need two main ingredients. First, we need a manner to measure the difference between the correct model which produces the data and our estimation. This is generally done by using some loss functions. Second, we need a function space where the solution to the learning problem must be sought. In example 2.4 such space is the space of polynomials up to degree 2. In a general context our choice will be a Reproducing Kernel Hilbert space.

This chapter is organized as follows. In Section 2.2 we study the learning process and convergence of the sample error. In Section 2.3 we introduce the concept of hypothesis space and we study, with some level of detail Reproducing Kernel Hilbert Spaces an the hypothesis spaces derived from them. In Section 2.4 we review some class of integral operators defined by kernels. We will focus in Regularization methods in Section 2.5. In particular we will analyze with some level of detail Ivanov and Tikhonov Regularization. We will study the Support Vector Machines as a particular case of Regularization in Section 2.6. In Section 2.7 we review the concepts of similarity and dissimilarity and their connections. We conclude in Section 2.8 with an analysis of the most relevant kernel combinations techniques in classification.

2.2 Learning process and convergence of the sample error

Let X be a compact space or manifold in an Euclidean Space and $Y = \mathbb{R}$. Let ν be a Borel probability measure defined on $Z = X \times Y$ whose regularity conditions will be assumed as needed.

Given $f : X \to Y$, we define the *Generalization Error* of f as

$$R_\nu(f) = \int_Z (f(x) - y)^2 d_\nu(x, y). \tag{2.1}$$

For each $x \in X$ and $y \in Y$, the function[1] $(f(x) - y)^2$ measures the error of using f to predict y when we observe x. The integral over Z averages this error for all the possible pairs (x, y).

The learning problem is posed as finding f that minimizes $R_\nu(f)$ and a natural way proceed is to decompose $R_\nu(f)$ into a sum. For every $x \in X$ let $\nu(y \,|\, x)$ the conditional probability measure on Y and $\nu(x)$ the marginal probability measure of ν on X. Let $\gamma : X \times Y \to \mathbb{R}$ an integrable function on $X \times Y$. Then

$$\int_{X \times Y} \gamma(x, y) d_\nu(x, y) = \int_X \left(\int_Y \gamma(x, y) d_\nu(y \,|\, x) \right) d_\nu(x). \tag{2.2}$$

[1] Any convex lower-bounded function can be used through this analysis.

Hence it is possible to break the measure ν into the measures $\nu(x \,|\, y)$ and $\nu(x)$ looking at ν as a product of an input domain and an output set.

Define $f_\nu : X \to Y$ by

$$f_\nu(x) = \int_Y y d_\nu(y|x). \tag{2.3}$$

This function is called *Regression function*. For each $x \in X$, $f_\nu(x)$ is the average of the y coordinate (in $\{x\} \times Y$). We will assume that f_ν is an bounded function.

Fix $x \in X$. Then the expectation of $(f_\nu(x) - y)$ is $\int_Y (f_\nu(x) - y) d_\nu(y \,|\, x) = 0$ and its variance

$$\int_Y (y - f_\nu(x))^2 d_\nu(y \,|\, x). \tag{2.4}$$

Then averaging over X and using eq. (2.2)

$$\int_X \left(\int_Y (f_\nu(x) - y)^2 d_\nu(y \,|\, x) \right) d_\nu(x) = \int_Z (f_\nu(x) - y)^2 d_\nu(x, y) = R_\nu(f_\nu). \tag{2.5}$$

Next we decompose the *Generalization Error* $R_\nu(f)$ into two independent terms as follows. For every $f : X \to Y$,

$$\begin{aligned} R_\nu(f) &= \int_Z (f(x) - f_\nu(x) + f_\nu(x) - y)^2 d_\nu(x, y) \\ &= \int_X (f(x) - f_\nu(x))^2 d_\nu(x) + \int_Z (f_\nu(x) - y)^2 d_\nu(x, y) \\ &+ \int_Z (f(x) - f_\nu(x))(f_\nu(x) - y) d_\nu(x, y) \\ &= \int_X (f(x) - f_\nu(x))^2 d_\nu(x) + R_\nu(f_\nu) \end{aligned}$$

Notice that $\int_Z (f(x) - f_\nu(x))(f_\nu(x) - y) d_\nu(x, y) = 0$ since $\int_Y (y - f_\nu(x)) d_\nu(y \,|\, x) = 0$. The term $\int_X (f(x) - f_\nu(x))^2 d_\nu(x)$ provides an average of the error when we use f instead of f_ν. The term $R(f_\nu)$ is independent of f and therefore f_ν is the function with the minimum error among the functions $f : X \to Y$. Then the goal of learning reduces *to find a good approximation to* f_ν what will be done from random samples on Z drawn from the probability measure ν.

Let $s_n = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\} \in Z^n$ be a random sample independently obtained from ν. Define the *Empirical Error* as

$$R_{s_n}(f) = \frac{1}{n} \sum_{i=1}^{n} (f(\mathbf{x}_i) - y_i)^2, \tag{2.6}$$

quantity that gives a measure of the error for a function f in the training sample s_n. In practice we will use $R_{s_n}(f)$ to approximate $R_\nu(f)$. Next we study how this approximation makes sense.

For any function $f : X \to Y$ define $f_Y : X \times Y \to Y$ as $f_Y(x) = f(x) - y$. Following the previous notation we can write:

$$\mathsf{E}_\nu(f_Y^2) = \int_Z (f(x) - y)^2 d_\nu(x, y) = R_\nu(f),$$

$$\mathsf{E}_{s_n}(f_Y^2) = \frac{1}{n}\sum_{i=1}^{n}(f(\mathbf{x}_i) - y_i)^2 = R_{s_n}(f),$$

where E_ν denotes the expectation over the measure ν and E_{s_n} denotes the expectation over the sample. In addition the variance of f_Y^2 is

$$\mathsf{Var}_\nu(f_Y^2) = \int_Z \left(f_Y^2 - \mathsf{E}_\nu(f_Y^2)\right)^2 d_\nu(x, y). \tag{2.7}$$

Define $F_{s_n}(f) = R_\nu(f) - R_{s_n}(f)$. The following theorem (Cucker and Smale, 2001) states a bound for $P\{|F_{s_n}(f)| \leq \epsilon\}$.

Theorem 2.1. *Let $M > 0$ and $f : X \to Y$ be such that $|f(x) - y| \leq M$ almost everywhere. Then for all $\epsilon > 0$,*

$$Prob\{|F_{s_n}(f)| \geq \epsilon\} \leq 1 - 2e^{\frac{n\epsilon^2}{2\left(\sigma^2 + \frac{1}{3}M\epsilon\right)}} \tag{2.8}$$

where $\sigma^2 = \mathsf{Var}_\nu(f_Y^2)$.

Theorem 2.1 ensures the convergence of the Empirical Risk to the Generalization Error when $n \to \infty$. Notice that since X is a compact space the condition $|f(x) - y| \leq M$ holds for any $x \in X$. Therefore it makes sense to approximate $R_\nu(f)$ by $R_{s_n}(f)$ in the learning process. Remark as well that the right hand side in the inequality above approaches 1 exponentially fast with n what guarantees a fast convergence of the Empirical Error to the Generalization error.

2.3 Hypothesis spaces

In any learning process some structure has to be assumed. This structure can be imposed by choosing some space of functions where the best approximation to the function f_ν should be sought. For instance, in Example 2.4 we constrain the regression function to be polynomials of at most degree 2.

We start by considering $C(X)$ the Banach space of continuous functions in X endowed with the norm

$$\|f\|_\infty = \underset{\mathbf{x}\in X}{sup}\, |f(\mathbf{x})|.$$

Consider a compact set $\mathcal{H}$ of $C(X)$, namely the *hypothesis space*: we will seek the best approximation for f_ν in $\mathcal{H}$.

Define the *target function* $f_{\mathcal{H}}$ as a function minimizing $R_\nu(f)$ over $f \in \mathcal{H}$ that is, an optimizer of

$$\min_{f\in\mathcal{H}} \int_Z (f(x)-y)^2 d_\nu(x,y). \tag{2.9}$$

Since the measure ν is unknown, problem in eq. (2.9) cannot be solved directly. Given a sample$s_n = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\} \in Z^n$ we define the *empirical target function* $f_{s_n,\mathcal{H}}$ as the function $f \in \mathcal{H}$ that optimizes

$$\min_{f\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n}(f(\mathbf{x}_i)-y_i)^2, \tag{2.10}$$

Imposing compactness on the hypothesis space $\mathcal{H}$ assures well-posedness of the problem in eq. (2.10). In Section 2.5 we will detail how a possible manner to ensure this is by minimizing the variational

$$F(f) = \frac{1}{n}\sum_{i=1}^{n}(f(\mathbf{x}_i)-y_i)^2 + \gamma\Omega(f)\,, \tag{2.11}$$

where $\gamma > 0$ and $\Omega(f)$ is a convex positive functional. Our choice will be $\Omega(f) = \|f\|_K^2$, being $\|f\|_K$ the norm of f in a Reproducing Kernel Hilbert Space.

2.3.1 Reproducing Kernel Hilbert Spaces

Reproducing Kernel Hilbert Spaces (RKHSs) (Aroszajn, 1950; Cucker and Smale, 2001; Wahba, 2003) represent a theoretical framework that can be used in a wide variety of problems such as time series (Parzen, 1970), independence of random variables (Bach and Jordan, 2002) smoothing surface estimation (Wahba, 1990), classification and regression problems (Moguerza and Muñoz, 2006), etc. We start with a formal definition of RKHS.

Definition 2.2. *A Hilbert space of functions H defined on a compact domain X is a Reproducing Kernel Hilbert Space (RKHS) if every linear evaluation functional $\mathcal{F}_x : H \to \mathrm{I\!R}$ is bounded: there exists a $M > 0$ such that*

$$|\mathcal{F}_x(f)| = |f(x)| \leq M\|f\| \quad \textit{for all } f \textit{ in the RKHS and } x \in X$$

where $\|\cdot\|$ is the norm in the Hilbert space.

Definition 2.3 (Mercer Kernel (Mercer, 1909)). *Let X a metric space and $K : X \times X \to \mathbb{R}$ a continuous and symmetric function. If we assume that K is positive definite, that is, for any set $\boldsymbol{x} = \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n\} \subset X$ the matrix $K\big|_{\boldsymbol{x}}$ with components $(\boldsymbol{K}\big|_{\boldsymbol{x}})_{ij} = K(x_i, x_j)$ is positive definite, then K is a Mercer Kernel.*

The Moore-Aronszajn theorem (Aroszajn, 1950) states that there exists a biunivocal correspondence between kernels and RKHSs. For each Reproducing Kernel Hilbert space of functions on X there exists a unique reproducing kernel K which is positive definite. Conversely, any Reproducing Kernel Hilbert Space can be characterized by a Mercer kernel. Next we explicitly state this point of the theorem since we will need in the sequel

Theorem 2.2 (Generation of RKHSs form kernels (Aroszajn, 1950)). *Let X be a compact domain or manifold and ν a Borel measure on X. Let $K : X \times X \to \mathbb{R}$ a continuous, symmetric and positive definite function. Define $K_{\boldsymbol{x}} : X \to \mathbb{R}$ the function given by $K_{\boldsymbol{x}}(\boldsymbol{t}) = K(\boldsymbol{x}, \boldsymbol{t})$. Then for every K there exists a unique Reproducing Kernel Hilbert Space $(\mathcal{H}_K, \langle,\rangle_{\mathcal{H}_K})$ of functions on X satisfying that:*

(i) For all $\boldsymbol{x} \in X$, $K_{\boldsymbol{x}} \in \mathcal{H}_K$.

(ii) The span of $\{K_{\boldsymbol{x}} : \boldsymbol{x} \in X\}$ is dense in $\mathcal{H}_K$.

(iii) For all $f \in \mathcal{H}_K$ then $\langle K_{\boldsymbol{x}}, f\rangle_{\mathcal{H}_K} = f(\boldsymbol{x})$.

In addition (Cucker and Smale, 2001) $\mathcal{H}_K$ consists of continuous functions and the inclusion $I_K : \mathcal{H}_K \to C(X)$ is bounded with $\|I_K\| \leq \sup_{x,t \in X} \sqrt{K(x,t)}$.

A particular way to generate $\mathcal{H}_K$ from K is next described. Let $\mathcal{H}'$ be space set of functions spanned by finite linear combinations of the form $f = \sum_{i=1}^n \alpha_i K(\mathbf{x}_i, \mathbf{x})$ where $n \in \mathbb{N}$, $\mathbf{x}_i \in X$ and $\alpha_i, \in \mathbb{R}$ equipped with the inner product

$$\langle f, g\rangle = \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \beta_j K(\mathbf{x}_i, \mathbf{x}_j), \tag{2.12}$$

for $f = \sum_{i=1}^n \alpha_i K(\mathbf{x}_i, \mathbf{x})$, $g = \sum_{j=1}^n \beta_j K(\mathbf{x}_j, \mathbf{x})$. Then $\mathcal{H}_K$ is the completion of $\mathcal{H}'$ with the associated inner product. That is we adjoint to $\mathcal{H}'$ all limits of Cauchy sequences (Wahba, 1990; Cucker and Smale, 2001).

Regarding the other sense of the implication of the Moore-Aronszajn theorem, if $\mathcal{H}'$ is an RKHS, by the Riesz representation theorem there exists a unique a function $K_{\mathbf{x}} \in \mathcal{H}'$ such that $\langle K_{\mathbf{x}}, f \rangle_{\mathcal{H}'} = f(\mathbf{x})$ for all $\mathbf{x} \in X$. The function $K_{\mathbf{x}}$ is called the point-evaluation functional at the point $\mathbf{x}$. Since $\mathcal{H}'$ is an space of functions, $K_{\mathbf{x}}$ is a function as well and it can be evaluated in any $\mathbf{x} \in X$. Define $K(\mathbf{x}, \mathbf{t}) = K_{\mathbf{x}}(\mathbf{t})$ for all $\mathbf{x}, \mathbf{t} \in X$. It can be proven that the function K is unique, symmetric and positive definite, that is, a Mercer kernel.

2.3.2 Hypothesis spaces associated to RKHSs

To ensure the existence and the unicity of the problem in eq. (2.10) we will define a compact set $\mathcal{H}$ based on $\mathcal{H}_K$. Next two propositions (Cucker and Zhou, 2007) are the key for this purpose.

Proposition 2.1. *Let K be a Mercer Kernel on a compact space X, and $\mathcal{H}_K$ its associated RKHS. For all $R > 0$ the ball $B_R := \{f \in \mathcal{H}_K \ : \ \|f\|_{\mathcal{H}_K} \leq R\}$ is a closed subset of C(X).*

Proposition 2.2. *Let K be a Mercer Kernel on a compact space X and $\mathcal{H}_K$ be its RKHS. For all $R > 0$, the set $I_K(B_R)$ is compact.*

Define $\mathcal{H} = I_K(B_R)$ for $R > 0$. By propositions 2.1 and 2.2, $\mathcal{H}$ is compact and therefore it makes sense to use it in eq. (2.10) as hypothesis space where $f_{s_n,\mathcal{H}}$ must be sought. Nevertheless, before affording this problem, we will study in next section a broad cast of integral operators defined by kernel functions that will be helpful to give a new characterization of RKHSs.

2.4 Operators defined by a kernel

Let $L^2_\nu(X)$ the space of squared integrable functions in X where ν is a Borel measure. Let $K : X \times X \to \mathbb{R}$ a continuous function. Then the (linear) map $L_K : L^2_\nu(X) \to C(X)$ defined by the operator

$$(L_K f)(\mathbf{x}) = \int K(\mathbf{x}, \mathbf{t}) f(\mathbf{t}) d_\nu(\mathbf{t}), \tag{2.13}$$

is well defined and the function K is called the kernel of L_K. Several properties of L_K can be obtained from the properties of K (Cucker and Smale, 2001). For instance, if K is continuous then L_K is compact and

$$\|L_K\| \leq \sqrt{\nu(X)} \sup_{x,t \in X} |K(x,t)|, \tag{2.14}$$

where is $\nu(X)$ the measure of X. In the sequel we will exclusively concentrate on Mercer's kernels.

If K is a Mercer kernel then L_K is self-adjoint, positive, compact and the Spectral theorem applies (Hochstadt, 1973; Conway, 1990): There exists an orthogonal basis $\{\phi_1, \phi_2, \dots\}$ of $L^2_\nu(X)$ consisting on eigenfunctions of L_K where each ϕ_j is given by

$$\phi_j(x) = \frac{1}{\lambda_j} \int_X K(x,t)\phi_j(t) \; d_\nu(t), \tag{2.15}$$

being λ_j its corresponding eigenvalue (See (Conway, 1990) for details). Thus, given ϕ_j, ϕ_i any two eigenfunctions of L_K, then $\|\phi_j\|_{L^2_\nu(X)} = 1$, $\langle \phi_j, \phi_i \rangle_{L^2_\nu(X)} = 0$ for $i \neq j$ and for any $f \in L^2_\nu(X)$, $f = \sum_{j=1}^{\infty} \langle f, \phi_j \rangle \phi_j$. In addition the set $\{\lambda_j\}$ is either finite or $\lambda_j \to 0$ when $j \to \infty$.

If $\lambda_j > 0$ then the eigenfunction ϕ_j is continuous and also lies in the RKHS $\mathcal{H}_K$. Then it belong to the span of $\{K_{\mathbf{x}} \,|\, \mathbf{x} \in X\}$. Additionally it can be proven (Cucker and Zhou, 2007) that $\|\phi_j\|_{\mathcal{H}_K} \leq \frac{1}{\lambda_j} \sqrt{\nu(X)} \sup_{x,t \in X} |K(x,t)|$ for $\| \cdot \|_{\mathcal{H}_K}$ the norm in $\mathcal{H}_K$.

Assuming that $\lambda_j \geq \lambda_{j+1}$ next theorem characterizes an orthogonal system in $\mathcal{H}_K$ using the eigenfunctions $\{\phi_j\}$.

Theorem 2.3. *Let ν a Borel measure on X and $K : X \times X \to \mathrm{I\!R}$ a Mercer kernel. Let λ_j be the j-th eigenvalue of L_K and ϕ_j the corresponding eigenfunction. Then $\{\sqrt{\lambda_j}\phi_j : \lambda_j > 0\}$ constitutes an orthogonal system in $\mathcal{H}$.*

Thus given ϕ_j, ϕ_i any two eigenfunctions of the integral operator L_K, then $\|\sqrt{\lambda_j}\phi_j\|_{\mathcal{H}_K} = 1$ and $\langle \sqrt{\lambda_j}\phi_j, \sqrt{\lambda_i}\phi_i \rangle_{\mathcal{H}_K} = 0$ for $i \neq j$. In fact theorem 2.3 is easy to prove applying the definition of ϕ_j and the reproducing property (Theorem 2.2). Let ϕ_j, ϕ_i be two eigenfunctions of L_K. Then

$$\begin{aligned} \langle \phi_i, \phi_j \rangle_{\mathcal{H}_K} &= \left\langle \frac{1}{\lambda_j} \int_X K(x,t)\phi_j(t) d_\nu(t), \phi_i \right\rangle_{\mathcal{H}_K} \\ &= \frac{1}{\lambda_j} \int_X \phi_j(t) \langle K(x,t), \phi_i(x) \rangle_{\mathcal{H}_K} d_\nu(t) \\ &= \frac{1}{\lambda_j} \int_X \phi_j(t)\phi_i(t) d_\nu(t) = \frac{1}{\lambda_j}\delta_{ij}, \end{aligned} \tag{2.16}$$

where $\delta_{ij} = 1$ if $i = j$ and 0 otherwise; therefore

$$\left\langle \sqrt{\lambda_i}\phi_i, \sqrt{\lambda_j}\phi_j \right\rangle_{\mathcal{H}_K} = \frac{\sqrt{\lambda_i \lambda_j}}{\lambda_j}\delta_{ij} = \delta_{ij}. \tag{2.17}$$

When the measure ν is non degenerate [2], the eigenvalues of the integral operator in L_K are all different form zero and the orthogonal system above is a basis of $\mathcal{H}_K$. In Section 2.4.2 we will use this to define a new characterization of $\mathcal{H}_K$. Previously, we introduce the Mercer theorem.

2.4.1 Mercer's theorem

Every Mercer kernel K can be written as a convergent series in $X \times X$. Let $f \in L^2_\nu(X)$ and consider $\{\phi_1, \phi_2, ...\}$ a basis of functions in $L^2_\nu(X)$, then f can be written as $f = \sum_{k=1}^\infty a_k \phi_k$ being the partial sums convergent in $L^2_\nu(X)$. If the convergence holds in $C(X)$ we say the the sum *converges uniformly*. In addition the series $\sum a_k$ *converges absolutely* if $\sum |a_k|$ converges.

Theorem 2.4 (Mercer's theorem (Mercer, 1909)). *Let X a compact domain or manifold, ν a nondegenerate Borel measure in X and $K : X \times X \longrightarrow \mathbb{R}$ a Mercer kernel. Let $\{\lambda_j\}_{j\geq 1}$ be the eigenvalues of L_K and $\{\phi_j\}_{j\geq 1}$ the corresponding eigenfunctions. Then, for all $x, y \in X$*

$$K(x, y) = \sum_{j=1}^{\infty} \lambda_j \phi_j(x) \phi_j(y)$$

where the series converges absolutely (for each $x, y \in X \times X$) and uniformly (in $x, y \in X \times X$).

By the previous theorem every we know that Mercer kernel K can be expressed as $K(x,t) = \sum_{j=1}^\infty \lambda_j \phi_j(x) \phi_j(t)$. In other words, the map $\Phi : X \to l^2$ given by $x \mapsto \left(\sqrt{\lambda_j}\phi_j(x)\right)_{j\in\mathbb{N}}$ (where l^2 is the linear space of all square summable sequences) satisfies

$$K(x,t) = \langle \Phi(x), \Phi(t) \rangle. \tag{2.18}$$

Thus K acts as a dot product in the embedding (the image of the nonlinear mapping Φ).

Example 2.5. The following kernels (if defined on a compact domain $X \in \mathbb{R}^n$) are Mercer Kernels.

Linear kernel: $K(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T \mathbf{y}$.

Polynomial kernel: $K(\mathbf{x}, \mathbf{y}) = (a + \mathbf{x}^T\mathbf{y})^b$ for $a, \in \mathbb{R}^+$ and $b \in \mathbb{N}$.

Gaussian kernel: given by $K(\mathbf{x}, \mathbf{y}) = e^{-\frac{1}{2\sigma^2}\|\mathbf{x}-\mathbf{y}\|^2}$ for $\sigma \in \mathbb{R}^+$.

• • •

[2] A measure on X is non degenerate when for each non empty subset $U \subseteq X$ then $\nu(U) > 0$.

Notice for any two points $\mathbf{x}_i, \mathbf{x}_j \in X$, Theorem 2.3 allows to estimate the inner product of $\Phi(\mathbf{x}_i)$ and $\Phi(\mathbf{x}_j)$ without knowing Φ explicitly. This property of kernel functions allows to kernelize any algorithm that could by written in terms of inner products of data. This can be done in two steps: first, mapping the data into the feature space defined by Φ implicitly through kernel function (replacing the original data inner products $\mathbf{x}_i^T \mathbf{x}_j$ with $K(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle$). Second, using the original algorithm on the corresponding feature space. Kernel Principal Components Analysis (Schölkopf et al., 1999), Kernel Ridge Regression (Ua and Pozdnoukhov, 2002) or the Support Vector Machines (detailed in Section 2.6) are some well known examples.

Mercer's kernels also provide an interpretation of $f \in \mathcal{H}_K$ as an hyperplane in the feature space. Let be $f(x) \in \mathcal{H}_K$ for $x \in X$. Then,

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i K(\mathbf{x}_i, \mathbf{x}) = \sum_{i=1}^{n} \alpha_i \Phi(\mathbf{x}_i)\Phi(\mathbf{x}) = \mathbf{w}^T \Phi(\mathbf{x}) \tag{2.19}$$

where $\mathbf{w} = \sum_i^n \alpha_i \Phi(\mathbf{x}_i)$ and Φ is the map $\Phi : X \to l^2$ given by $x \mapsto \left(\sqrt{\lambda_j}\phi_j(x)\right)_{j \in \mathbb{N}}$. That is, $f(\mathbf{x}) = 0$ describes a hyperplane in the feature space determined by Φ.

Next we conclude this section with a new characterization of RKHSs based on the eigenfunctions and eigenvalues of self-adjoint, positive, compact integral operators.

2.4.2 RKHSs revisited

As we already mentioned in Section 2.4 if ν, the Borel measure in X, is not degenerate then the set $\{\sqrt{\lambda_j}\phi_j : \lambda_j > 0\}$ constitutes a basis of $\mathcal{H}_K$ where ϕ_j is the j-th eigenfunction of L_K and λ_j its corresponding eigenvalue.

Since $\mathcal{H}_K$ is independent of the measure ν, when $dim(\mathcal{H}_K) = \infty$ then L_K has infinitely many positive eigenvalues λ_j and we can characterize $\mathcal{H}_K$ as:

$$\mathcal{H}_K = \left\{ f \in L^2_\nu(X) \ : \ f = \sum_{j=1}^{\infty} a_j \sqrt{\lambda_j}\phi_j \ \ with \ \ a_j \in l^2 \right\}. \tag{2.20}$$

Let $f = \sum_{j=1}^{\infty} a_j \sqrt{\lambda_j}\phi_j$ and $g = \sum_{j=1}^{\infty} b_j \sqrt{\lambda_j}\phi_j$ be two functions in $\mathcal{H}_K$; then the inner product of f and g in $\mathcal{H}_K$ is:

$$\begin{aligned}\langle f, g\rangle_{\mathcal{H}_K} &= \left\langle \sum_{j=1}^{\infty} a_j \sqrt{\lambda_j}\phi_j, \sum_{i=1}^{\infty} b_i \sqrt{\lambda_i}\phi_i \right\rangle_{\mathcal{H}_K} \\ &= \sum_{j=1}^{\infty}\sum_{i=1}^{\infty} b_j a_i \left\langle \sqrt{\lambda_i}\phi_i, \sqrt{\lambda_j}\phi_j \right\rangle_{\mathcal{H}_K} = \sum_{j=1}^{\infty} b_j a_i.\end{aligned}$$

If the dimension of $\mathcal{H}_K$ is finite (only $m < \infty$ eigenvalues of L_K are different form zero) the previous results also applies. In this case we define

$$\mathcal{H}_K = \left\{ f \in L^2_\nu(X) \ : \ f = \sum_{j=1}^{m} a_j \sqrt{\lambda_j}\phi_j \ \ with \ \ (a_1, a_2, \ldots, a_m)^T \in \mathrm{I\!R}^m \right\} \tag{2.21}$$

with inner product $\langle f, g\rangle_{\mathcal{H}_K} = \sum_{j=1}^{m} a_j b_j$ for any pair of functions $f = \sum_{j=1}^{m} a_j \sqrt{\lambda_j}\phi_j$ and $g = \sum_{i=1}^{m} b_i \sqrt{\lambda_i}\phi_i$.

Remark that although $\mathcal{H}_K$ can be defined through the eigenfunctions and eigenvalues of L_K is independent of the measure ν (notice that the space $\mathcal{H}_K$ was defined only using X and K).

In the literature is also common to define $\mathcal{H}_K$ as the space of all functions $f \in L^2_\nu(X)$ such that $f = \sum_{j=1}^{\infty} a_j \phi_j$ where $a_j \in \mathrm{I\!R}$ and $(a_j/\sqrt{\lambda_j}) \in l^2$ equipped with the inner product $\langle f, g\rangle = \sum_{j=1}^{\infty} \lambda_j^{-1} a_j b_j$ for $f = \sum a_j \phi_j$ and $g = \sum b_j \phi_j$. It is straightforward to check that this definition of $\mathcal{H}_K$ is equivalent to eq. (2.20).

For simplicity in notation we will denote $\langle \cdot, \cdot \rangle_{\mathcal{H}_K}$ by $\langle \cdot, \cdot \rangle_K$ and $\| \cdot \|_{\mathcal{H}_K}$ by $\| \cdot \|_K$ in the sequel.

2.5 Regularization in RKHSs

Next we turn back to the main objective of the learning process, the estimation of $f_{s_n,\mathcal{H}}$ the minimizer of eq. (2.10). To this aim we describe two different strategies: Ivanov and Tikhononv Regularization.

2.5.1 Ivanov Regularization

Let X a compact space or manifold and ν a Borel measure in X and Y. Let $s_n = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\} \in Z^n$ a sample of n examples independently drawn from ν in

X and $K : X \times X \longrightarrow \mathbb{R}$ be a Mercer kernel. Let L_K be the integral operator associated to K and I_K the compact inclusion defined in Theorem 2.2.

To ensure the existence and the unicity of the minimizer of eq. (2.10) we need define $\mathcal{H}$ compact. To this aim we follow Propositions 2.1 and 2.2 and we define $\mathcal{H} = I_K(B_R)$, for $B_R := \{f \in \mathcal{H}_K : \|f\|_K \leq R\}$, the compact set where $f_{s_n,\mathcal{H}}$ must be sought. Then we afford the learning problem by finding an *empirical target function* $f_{s_n,\mathcal{H}_K}$ that optimizes

$$\min \ \frac{1}{n}\sum_{i=1}^{n}(f(\mathbf{x}_i) - y_i)^2 \tag{2.22}$$
$$s.t \ f \in B(\mathcal{H}_K, R),$$

where $B(\mathcal{H}_K, R)$ is the ball of radius R defined by the compact inclusion I_K on $\mathcal{H}_K$. Optimization problems in eqs. (2.22) are known as Ivanov Regularization (Ivanov, 1976) problems.

Denote by $\mathcal{H}_{K,s_n}$ the finite subspace of $\mathcal{H}_K$ spanned by $\{K(\mathbf{x}_1, \mathbf{x}), \ldots, K(\mathbf{x}_n, \mathbf{x})\}$ (where the $\mathbf{x}_i$ are now the points of the sample) and P the orthogonal projection $P : \mathcal{H}_K \longrightarrow \mathcal{H}_{K,s_n}$. In practice it makes sense to choose the empirical function from $\mathcal{H}_{K,s_n}$ instead of $\mathcal{H}_K$. If f is a minimizer of R_{s_n} in $B(\mathcal{H}_K, R)$, then $P(f)$ is an minimizer of R_{s_n} in $P(B(\mathcal{H}_K, R))$ (the image of B under P) and the problem (2.22) can restated as a convex programming problem whose solution $f^* = \sum_{i=1}^{n} c_i^* K(\mathbf{x}_i, \mathbf{x})$ can be algorithmically obtained. See (Cucker and Zhou, 2007) for further details.

Ivanov Regularization represents a natural way to apply the Empirical Risk Minimization (ERM) principles proposed by Vapnik (Vapnik, 1998). The idea of ERM is to find $f_{s_n,\mathcal{H}}$ ensuring a balance between its precision (measured in this case by $\frac{1}{n}\sum_{i=1}^{n}(f(x_i) - y_i)^2$) and its complexity (measured by $B(\mathcal{H}_K, R)$). To this aim the ERM considers a sequence of nested hypothesis spaces $\mathcal{H}_1 \subset, \ldots, \mathcal{H}_M$ where $\mathcal{H}_i = \{f \in H : \Omega(f) \leq A_i\}$ for $\Omega(f)$ a convex positive functional and $A_1, \ldots, A_M$ are a sequence of increasing scalars. In our particular case $\Omega(f) = \|f\|_K$ and the $A_i = R_i$ (the radii of the compact balls). For each R_i the Ivanov problem must be solved selecting $f^*_{s_n,\mathcal{H}}$ that minimizes $R_{s_n,\mathcal{H}_i}(f)$.

The Ivanov approach is consistent but computationally very expensive. Other alternatives have been proposed in the literature (Philips, 1962; Tikhonov and Arsenin, 1977). In next chapter we will focus in this last case since it is computationally treatable and it automatically guarantees compactness of the hypothesis space.

2.5.2 Tikhonov Regularization in RKHSs

In this section we move away from the "compact hypothesis space approach" followed until now by slightly changing the point of view. Next we focus on Tikhovov regularization in RKHSs as alternative to the problem in eq. (2.22).

The central idea of Tikhovov regularization is to address well posedness in eq. (2.22) adding a penalization term to $R_\nu(f)$ (or to R_{s_n} in the sampled version). In this section, we will consider that $\mathcal{H} = \mathcal{H}_K$: The hypothesis space is now the whole RKHS.

Let X be a compact space or manifold in an Euclidean Space and $Y = \mathbb{R}$. Let ν be a Borel probability measure defined on $Z = X \times Y$. We define the *γ-Regularized Error* as

$$R_\gamma = \underset{f \in \mathcal{H}_K}{min} \int_Z (f(\mathbf{x}) - y)^2 d_\nu(x, y) + \gamma \|f\|_K^2 \,, \tag{2.23}$$

where $\gamma > 0$ and $\|f\|_K$ represents the norm of f in $\mathcal{H}_K$. Since the measure ν is unknown, in practice we work with $s_n \in$ for $s_n = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\} \in Z^n$ a random sample of n examples independently drawn from ν. Then the previous γ-Regularized Error is approximated by the *Empirical γ-Regularized Error* given by

$$R_{\gamma, s_n} = \underset{f \in \mathcal{H}_K}{min} \frac{1}{n} \sum_{i=1}^{n} (f(\mathbf{x}_i) - y_i)^2 + \gamma \|f\|_K^2 \,. \tag{2.24}$$

where the scalar γ controls the balance between the minimization of R_{s_n} (eq. (2.6)) and the approximation capacity of the hypothesis space (measured by $\|f\|_K^2$).

The existence of functions $f_{\gamma, \mathcal{H}_K}$ and $f_{\gamma, s_n, \mathcal{H}_K}$ minimizing respectively eqs. (2.23) and (2.24) is not guarantee ($\mathcal{H}$ is no longer compact). However in (Cucker and Smale, 2001) is proven that both optimizers exist and are unique. Regarding the problem in eq. (2.23) its minimizer is given by:

$$f_{\gamma, \mathcal{H}_K} = (L_K + \gamma Id)^{-1} L_K f_\nu, \tag{2.25}$$

where L_K is the integral operator associated to the kernel K and f_ν defined in eq. (2.3). Moreover it can be shown (Mukherjee et al., 2002), (Bousquet and Elisseeff, 2002), that the space where the solution is sought takes the form $\{f \in H_K : \|f\|_K^2 \leq \sup_{y \in Y} y^2/\gamma\}$ guaranteeing this way compactness of the hypothesis space.

Notice that when γ grows, the radius of the ball decreases and the space is smaller. Therefore the scalar γ plays a similar role to the radius R of the ball of functions B_R

defined in the Ivanov approach. See Remark 8.26 of (Cucker and Zhou, 2007), for a detailed relationship between problems in eqs. (2.22) and (2.24).

To conclude this section we enunciate the Representer theorem which characterizes the solutions of (2.24). This theorem was introduced in (Kimeldorf and Wahba, 1970) within the context of smoothing splines and it has been widely used to characterize the solution of risks minimization functional in RKHSs. For details, proofs and generalizations, refer to (Schölkopf et al., 2000) (Cox and O'Sullivan, 1990).

Theorem 2.5 (Representer Theorem). *Let $s_n \in Z^n$ be a random sample, K a kernel function and $\gamma > 0$. Then the empirical target function that minimizes eq. (2.24) exists, is unique and admits a representation of the form*

$$f_{\gamma,s_n,\mathcal{H}_K} = \sum_{i=1}^{n} \alpha_i K(\boldsymbol{x}_i, \boldsymbol{x}), \quad \forall \boldsymbol{x} \in X, \tag{2.26}$$

being the coefficients $\alpha_i \in \mathbb{R}$ the solutions to the linear system:

$$(\gamma n I_n + K\big|_{\boldsymbol{x}})\boldsymbol{\alpha} = \boldsymbol{y}, \tag{2.27}$$

where $K\big|_{\boldsymbol{x}}$ is the $n \times n$ kernel matrix with components $(K\big|_{\boldsymbol{x}})_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$, $\boldsymbol{y} = (y_1, \ldots, y_n)^T$, $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$ and $\boldsymbol{I}_n$ is the identity matrix of dimension n.

With the Representer theorem we conclude the learning process giving a close expression for $f_{\gamma,s_n,\mathcal{H}_K}$, the minimizer of eq. (2.24). However the generality of the regularization approach allows to afford a wide range of statistical problems by changing the loss function in eq. (2.24) (Poggio and Gorosi, 1998). Next we will study a particular case specially interesting in classification problems: the Support Vector Machines.

2.6 Classification problems and Support Vector Machines

While in previous sections we have studied the learning problem for the regression perspective now we will focus in binary classification problems. In addition we will see how Support Vector Machines can be studied to address the problem as a variation of eq. (2.24).

Let X be a subset of $\mathbb{R}^p$, $Y = \{-1, 1\}$ (label space) ν and ν Borel measure on $X = X \times Y$. Consider $s_n = \{(\mathbf{x}_i, y_i)\}_{i=1}^n \in Z^n$ a random sample drawn for ν. To solve the classification problem means to learn from s_n a function (called classifier) $f : X \to Y$ that relate each data $\mathbf{x}_i \in X$ with its label $y_i \in Y$. The misclassification error of a classifier $f : X \times X \rightarrow Y$ is defined as the probability of a wrong prediction, that is, the measure of the event $\{f(\mathbf{x}) \neq y\}$ for any $(\mathbf{x}, y) \in Z$ given by

$$R_{c,\nu}(f) = Prob\{f(\mathbf{x}) \neq y\} = \int_X Prob\{f(\mathbf{x}) \neq y \,|\, \mathbf{x}\} d_\nu(\mathbf{x}) \tag{2.28}$$

In Section 2.2 we showed that the function that minimizes the Generalization Error $R_\nu(f)$ (see eq. (2.1)) is the regression function f_ν defined in eq. (2.3). The function that minimizes eq. (2.28) is known as the Bayes rule and it is given by:

$$f_{c,\nu}(\mathbf{x}) = \begin{cases} +1 & \text{if } Prob(y = 1 \,|\, \mathbf{x}) \geq Prob(y = -1 \,|\, \mathbf{x})\,, \\ \\ -1 & \text{if } Prob(y = 1 \,|\, \mathbf{x}) < Prob(y = -1 \,|\, \mathbf{x})\,. \end{cases} \tag{2.29}$$

The function $f_{c,\nu}(\mathbf{x})$ is non computable since ν is unknown. Then to approximate $f_{c,\nu}(\mathbf{x})$ from s_n we will first derivate a computable discrimination function $f : X \rightarrow \mathbb{R}$ and we will define the final classifier as

$$sign(f)(\mathbf{x}) = \begin{cases} +1 & \text{if } f(\mathbf{x}) \geq 0\,, \\ \\ -1 & \text{if } f(\mathbf{x}) < 0\,. \end{cases} \tag{2.30}$$

Next we deduce a particular choice for f, the Support Vector Machines, by using the regularization theory approach.

2.6.1 Support Vector Machines as regularization method

Support Vector Machines (SVMs) appeared as optimal margin classifiers in the context of Vapnik's statistical learning theory (Vapnik, 1998). They have been applied to a large amount of real-world data problems obtaining very competitive results and becoming one of the most relevant techniques in classification (Vapnik, 1998; Cristianini and Shawe-Taylor, 2000; Moguerza and Muñoz, 2006; Hastie et al., 2009), regression (Smola and Schölkopf, 2003) and density estimation (Muñoz and Moguerza, 2006) problems.

A fast search in Google with the terms "*Support Vector Machines*" shows more than 24 millions of results and the number of publisher papers related to SVMs are increasing since they were introduced in (Boser et al., 1992).

The original idea of SVMs appears by first time for binary classification problems. In (Boser et al., 1992) a linear decision function to maximize the separation between the classes in a classification problem is estimated after mapping the data onto a high dimensional space via the use of a kernel function. This approach combines successfully the idea of transforming data onto a high dimensional space using a kernel (kernel-trick) in the spirit of potential functions (Aizerman et al., 1964) with the mathematical programming techniques for the calculation of and hyperplane in a non parametric context (Vapnik and Chervonenkis, 1964).

The interpretation of the SVMs as regularization method is due to Wahba. As is described in (Wahba, 2006), one of the comments to the paper (Moguerza and Muñoz, 2006), the connection between original approach and regularization was discovered at an American Mathematical Society meeting at Mt. Holyoke in 1996: While the speaker was describing the SVM an anonymous person remarked that the SVM with the kernel trick was the solution to an optimization problem in a reproducing kernel Hilbert space.

Let X be a compact space or manifold, $Y = \{-1, 1\}$, ν a Borel measure defined over $Z = X \times Y$ and $s_n = \{(\mathbf{x}_i, y_i)\}_{i=1}^n \in Z^n$ a random sample independently drawn form ν. Then SVMs seeks a function f that minimizes the Empirical Error

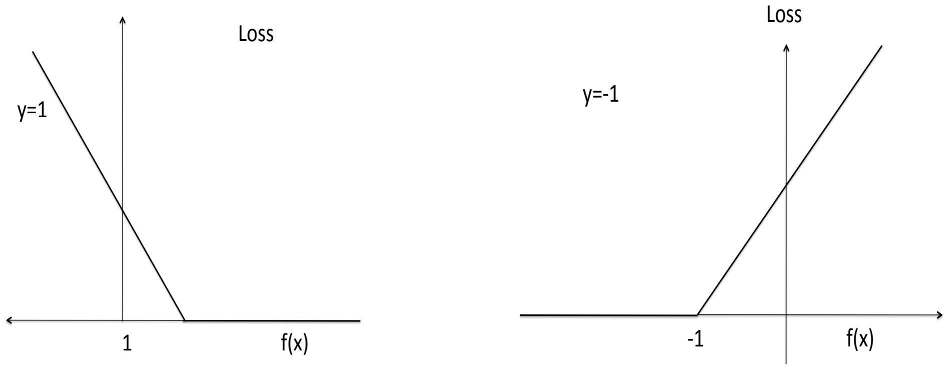
$$R_{s_n,svm}(f) = \frac{1}{n}\sum_{i=1}^{n}(1 - y_i f(\mathbf{x}_i))_+. \tag{2.31}$$

The squared loss function function is here replaced by the so-called *hinge loss* function given by $x_+ = \max(x, 0)$ for any $x \in \mathbb{R}$. The hinge loss does not penalize large values of $f(\mathbf{x}_i)$ with the same sign as y_i (understanding by large $|f(\mathbf{x}_i)| \geq 1$). Therefore only points such that $(1 - y_i f(\mathbf{x}_i))_+ > 0$ will be taken into account in the characterization of the decision function (see Figure 2.2). Remark that, since the hinge loss is convex, Theorem 2.1 can be extended to this case and therefore convergence of the Empirical Error in eq. (2.31) to $R_{\nu,svm}(f) = \int_Z (1 - y_i f(\mathbf{x}_i))_+ d_\nu(x, y)$ is guaranteed.

To reach well-posedness, SVMs make use of regularization theory. In particular, we will use here the Tikhonov regularization approach. Then the SVM can be understood as a variation of eq. (2.24) where the squared loss function is replaced by the hinge loss function. Hence the SVM solution obtains the discrimination function via the minimization of the following risk functional:

$$\min_{f \in H_K} \frac{1}{n}\sum_{i=1}^{n}(1 - y_i f(\mathbf{x}_i))_+ + \gamma \|f\|_K^2 , \tag{2.32}$$

where $\gamma > 0$, $\mathcal{H}_K$ is the RKHS associated to the kernel K and $\|f\|_K$ denotes the norm of f in the RKHS.



(a) Hinge loss function for points such that $y_i = 1$.

(b) Hinge loss function for points such that $y_i = -1$.

Figure 2.2: Hinge loss function.

By convexity of the hinge loss function, the Representer theorem (Theorem 2.5) ensures that the solution to problem (2.32) has the form $f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i K(\mathbf{x}_i, \mathbf{x}) + b$, where the constant b can be included without loss of generality (Poggio et al., 2001).

It is immediate to show that $\|f\|_K^2 = \|\mathbf{w}\|^2$, where $\mathbf{w} = \sum_i^n \alpha_i \Phi(\mathbf{x}_i)$ and Φ is the map $\Phi : X \to l^2$ given by $x \mapsto \left(\sqrt{\lambda_j}\phi_j(x)\right)_{j \in \mathbb{N}}$ (defined in Section 2.4) where ϕ_j and λ_j are respectively the eigenfunctions and eigenvalues of L_K, the integral operator associated to K. Then problem (2.32) can be restated as

$$\min_{\mathbf{w},b} \frac{1}{n} \sum_{i=1}^{n} \left(1 - y_i(\mathbf{w}^T \Phi(\mathbf{x}_i) + b)\right)_+ + \mu \|\mathbf{w}\|^2 . \tag{2.33}$$

Eqs. (2.32) and (2.33) summarize some the key issues of SVMs: Through the use of kernels, the a priori problem of estimating a nonlinear decision function that assigns each data to its label is transformed into the a posteriori problem of estimating the weights $\mathbf{w}$ of a hyperplane in the feature space induced by K (mapping Φ).

The hinge loss is a piecewise linear function and therefore is not differentiable. Then eqs (2.32) and (2.33) are non-differentiable either which implies a difficulty for efficient optimization techniques. See (Bazaraa et al., 1993) for details. However, problem (2.33) can be turned smooth by reformulating it as,

$$\begin{aligned} \min_{\mathbf{w},b} \quad & \tfrac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i \\ \text{s.t.} \quad & y_i\left(\mathbf{w}^T\Phi(\mathbf{x}_i)+b\right) \geq 1-\xi_i\,, \quad && i=1,\dots,n\,, \\ & \xi_i \geq 0\,, && i=1,\dots,n\,, \end{aligned} \tag{2.34}$$

where ξ_i are slack variables introduced to avoid the non-differentiability of the hinge loss function and $C = 1/(2\mu n)$. See (Lin, 2002) for details. Efficient methods have been proposed in the literature to solve (2.34) (Joachims, 2002; Osuna et al., 1997; Platt, 1999).

The desired decision function that will be used in eq. (2.30) will be an hyperplane given by:

$$f^*(\mathbf{x}) = (\mathbf{w}^*)^T\Phi(\mathbf{x}) + b^* = \sum_{i=1}^{n}\beta_i^* y_i K(\mathbf{x}_i,\mathbf{x}) + b^*. \tag{2.35}$$

where the vector $\boldsymbol{\beta}^* = (\beta_1^*,\dots,\beta_n^*)^T$ is the solution the dual problem of (2.34). See (Moguerza and Muñoz, 2006) for details. Both λ_i^* and b^* only depend on the sample. Moreover, those points satisfying that $\lambda_i^* > 0$ are called the support vectors. Therefore $f^*(\mathbf{x})$ is completely determined by the subsample made up by the support vectors. Often happens that the support vectors are a small fraction of the data sample and the solution is usually sparse. This property is due to the use of the hinge loss function.

The hyperplane estimated by the SVM is neither arbitrary nor unstable despite the Cover's theorem (Cover, 1965) which guarantees that any data set becomes arbitrarily separable as the data dimension grows. This happen by two main facts: the SVM solution it is the result of the regularization problem in eq. (2.32) and it is proven that the empirical error for SVMs converges to the expected error as $n \to \infty$. In addition, notice that eq. (2.35) only depends on kernel evaluations of the form $K(\mathbf{x},\mathbf{y})$. Therefore, we do not need to know explicitly Φ to solve the SVM problem.

To conclude this section we illustrate in a simulated example how the choice of the kernel function affects the discrimination function estimated by the SVM.

Example 2.6. In this example we illustrate the influence of the kernel choice in the discrimination function estimated by a SVM in a classification problem. To this aim we consider the task of separating two spirals of 50 data each. The two classes of the generated data are represented by white and black points in any of the four plots of Figure 2.3.

Table 2.1: Results obtained for the two spirals problem. Classification errors and proportion of support vectors are shown.

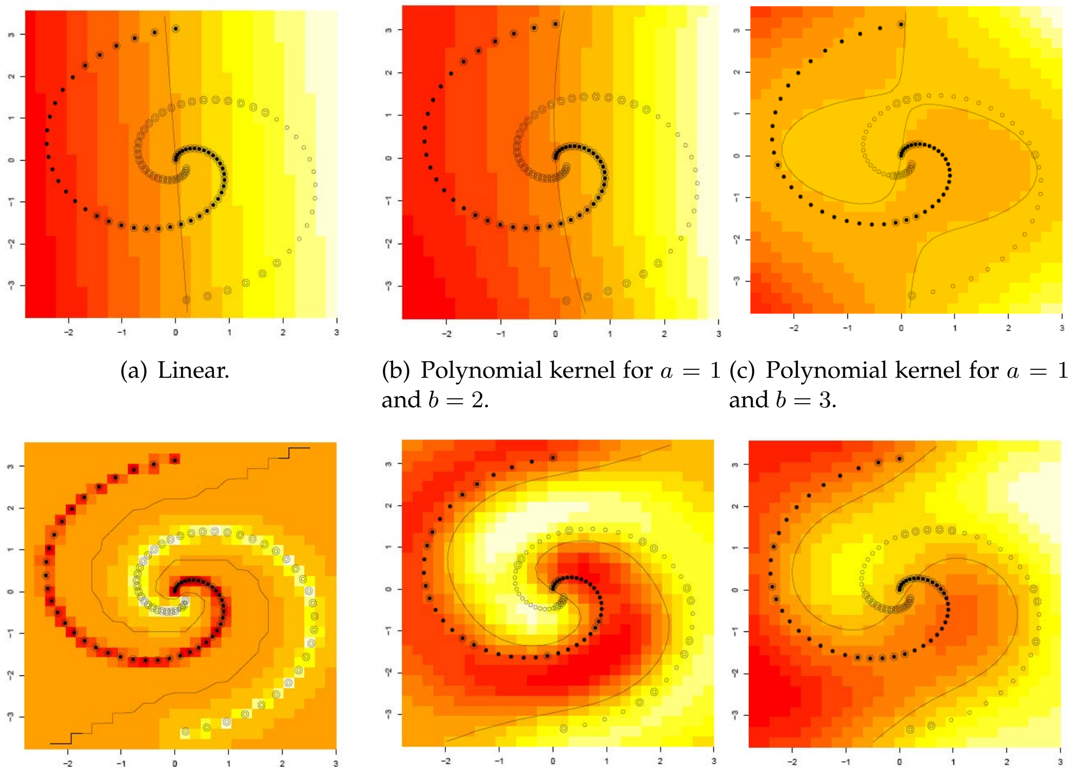
Kernel	Classification Error (%)	Support Vectors (%)
Linear	36	76
Polynomial $a = 1, b = 2$	36	76
Polynomial $a = 1, b = 3$	18	26
Gaussian $\rho = 0.1$	0	96
Gaussian $\rho = 1$	4	34
Gaussian $\rho = 2$	14	48

We train 6 different SVMs with different kernels: a linear kernel, two polynomial kernels of degrees 2, and 3 and three gaussian kernels of parameters 0.1, 1 and 2. The penalization parameter C is fixed to 10 in all cases. Given that the data are non linearly separable the linear kernel will exhibit a poor performance. However the use of other kernels that transform the data to a higher dimensional space (like the polynomial or the Gaussian) will work better that the linear one in this example.

In Table 2.1 we show the percentage of well classified points and support vectors for the six kernels. In addition we include in Figure 2.3 a graphical solution that includes (in black) the discrimination function $f(\mathbf{x}) = 0$ in each case. The data that are support vectors are remarked and the color of the plot represents the value of the estimated function $f(\mathbf{x})$ in each point of the plot.

As expected, the six discrimination function are different. Using the linear kernel and the polynomial kernel of degree 2 we obtain in both cases a 36 % of misclassified data. By far this is the worst result in this experiment. With the polynomial of degree 3 the percentage of misclassified data decreases to 18%. However the best results are obtained for the Gaussian kernels. The only decision functions that separates perfectly the two spirals corresponds to that one estimated using a Gaussian kernel with parameter $\rho = 0.1$. However this discrimination presents a serious drawback. First, the high proportion of support vectors (96 % in this case) generally indicates a poor generalization capability of the model (Cristianini and Shawe-Taylor, 2000). This is confirmed by the colors of the Figure 2.3 d) that represents the value of $f^*(\mathbf{x})$. Excepting for a tiny neighborhood of the observations, the plot is monochromatic. This means that for different points in the sample the discrimination function is useless since its discrimination capability is poor. On the other hand, the solution obtained with a Gaussian kernel with $\rho = 1$, does not presents this problem and only four data points are wrongly classified.

• • •



(a) Linear.

(b) Polynomial kernel for $a = 1$ and $b = 2$.

(c) Polynomial kernel for $a = 1$ and $b = 3$.

(d) Gaussian kernel for $\rho = 0.1$.

(e) Gaussian kernel for $\rho = 1$.

(f) Gaussian kernel for $\rho = 2$.

Figure 2.3: Example of classification problem with two spirals. The support vectors are remarked and the color of the plot represents the value of the estimated function $f(\mathbf{x})$. In black, the decision function $f(\mathbf{x}) = 0$ is also shown.

As it is shown in eq. (2.35) and verified experimentally in Example 2.6 the final model implemented in SVMs is strongly influenced by the choice of the kernel. To decide which kernel is the most suitable for a particular problem is an important and open issue. Several strategies have been proposed to choose K in the SVMs context (Keerthi and Lin, 2003). Nevertheless, to use a single kernel may be not enough to solve accurately the problem under consideration. This happens in analysis where results may vary a lot depending on the data similarity chosen. Thus, the information provided by a single similarity measure (kernel) may be not enough for classification purposes, and the combination of kernels appears as an interesting alternative to the model selection problem. Given the relationship between proximity measures (distances, dissimilarities or similarities) and kernels, we will first review these concepts next as a previous step to study in Section 2.8 some combinations schemes proposed in the literature.

2.7 Proximity measures

In this section we will use the term proximity to refer similarity, dissimilarity and distance functions.

For any set X, a *distance function* is an application $d : X \times X \to \mathbb{R}$ such that for all $x, y \in X$:

(i) $d(x,x) = 0$.

(ii) $d(x,y) \geq 0$ (non-negativity).

(iii) $d(x,y) = d(y,x)$ (symmetry).

(iv) $d(x,y) = 0$ if $x = y$ (definiteness).

(v) $d(x,y) + d(y,z) \leq d(x,z)$ (triangle inequality).

2.7.1 Dissimilarities and distance functions

A function $\delta : X \times X \to \mathbb{R}$ which satisfies the two first conditions is called *dissimilarity function*. The symmetry condition is not always satisfied in real applications it will be studied in Chapter 4.

Let X be an space equipped with a distance d. Then we the say that (X, d) is a metric space. In addition a matrix with components $(\mathbf{D})_{ij} = d(\mathbf{x}_i, \mathbf{x}_j)$ for $i, j = 1, \ldots, n$ (dissimilarities between a set of points $\mathbf{x}_1, \ldots, \mathbf{x}_n$) is called dissimilarity matrix or distance matrix, independently of the specific properties of d.

2.7.2 Similarity functions

For any set X, a *similarity function* is an application $s : X \times X \to \mathbb{R}$ such that given any $x, y \in X$:

(i) $s(x,x) > 0$.

(ii) $s(x,y) = s(y,x)$ (symmetry).

(iii) $s(x,y) \geq 0$ (non-negativity).

(iv) $\sum_{i,j=1}^{n} c_i c_j s(x_i, x_j) \geq 0$ for all $n \in \mathbb{N}$ $c_i \in \mathbb{R}$ and $x_i, x_j \in X$ (positive definiteness).

Similarity functions which are positive definite can be used to define *kernel functions*.

2.7.3 Relationship between similarities, dissimilarities and kernel functions

Most classification and cluster algorithms are able to work with similarities or dissimilarities. However sometimes a transformation to convert similarities into dissimilarities or vice versa it is necessary.

From similarities to dissimilarities

- Let s be a normalized similarity. That is $0 \leq s(x, y) \leq 1$ and $s(x, x) = 1$ for all $x, y \in X$. Then some typical ways to obtain dissimilarity functions from s are:

  (i) $d(x, y) = 1 - s(x, y)$

  (ii) $d(x, y) = \sqrt{1 - s(x, y)}$

  (iii) $d(x, y) = \sqrt{s(x, x) + s(x, x) - 2s(x, y)}$

  See (Gower, 2000) for additional methods.

- If the similarity function comes from a scalar product in a Euclidean space then we can calculate the asociated metric via $d(x, y)^2 = \langle x, x\rangle + \langle y, y\rangle - 2\langle x, y\rangle$. Notice that this is particular case of kernel functions.

From dissimilarities to similarities

- Use the previous equations (i), (ii) and (iii) writing the similarity in terms of the dissimilarity. Remark that (iii) is known in this case as Multidimensional Scaling (Cox and Cox, 2001).

- Let d be a Euclidean distance. Then a positive definite similarity function can be calculated by $s(x, y) = \frac{1}{2}\left(d(x, 0)^2 + d(y, 0)^2 - d(x, y)^2\right)$ where 0 is the origin (or some other point in X playing its role).

- Let d be a dissimilarity function. Then any non-negative decreasing function of d is a similarity function. For instance, $s(x, y) = \exp(-d(x, y)^2/\sigma)$ for $\sigma \in \mathbb{R}^+$ is a similarity function for any $x, y \in X$.

2.8 The combination of kernels within the support vector framework

In this section we review the existing literature in kernel combinations for binary classification problems. Let X be a compact space or manifold $Y = \{-1, 1\}$ and ν a Borel measure defined on $Z = X \times Y$. Let $K_1, \ldots, K_m$ be a set of Mercer's kernels where $K_i : X \times X \to \mathbb{R}$ for $i = 1, \ldots, m$ and $s_n = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\} \in Z^n$ a random sample independently obtained from ν. The goal in kernel combination procedures is to find a kernel function $K^* : X \times X \to \mathbb{R}$ derived from the original collection of kernels that capture the right idea of similarity between the data points. In our context the final kernel K^* will be used to train a Support Vector Machine.

In kernel combination we work with similarities instead of kernels. As we already detailed in the previous section, this is not a problem since Euclidean distances can be deduced from positive definite kernels. Hence similarities can be obtained as well form kernels.

The underlying hypothesis in similarity combination is that, in the feature space, all the point in an small enough neighborhood belong to the same class. Therefore, in classification problems makes sense to define kernel combinations such that points that belong to the same class are close in the feature space and far away otherwise. In (Martín de Diego et al., 2009) this idea in implemented via the so-called Max-Min method. Given the collections of kernels $K_1, \ldots, K_m$ and the sample s_n, K^* is defined by:

$$K^*(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} \max\left(K_1(\mathbf{x}_i, \mathbf{x}_j), \ldots, K_m(\mathbf{x}_i, \mathbf{x}_j)\right) & \text{if } y_i = y_j \,, \\ \\ \max\left(K_1(\mathbf{x}_i, \mathbf{x}_j), \ldots, K_m(\mathbf{x}_i, \mathbf{x}_j)\right) & \text{if } y_i \neq y_j \,. \end{cases} \tag{2.36}$$

It can be proven that, for $m = 2$, eq. (2.36) can be reformulated as

$$K^*(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{2}(K_1(\mathbf{x}_i, \mathbf{x}_j) + K_2(\mathbf{x}_i, \mathbf{x}_j)) + y_i y_j \frac{1}{2} |K_1(\mathbf{x}_i, \mathbf{x}_j) - K_2(\mathbf{x}_i, \mathbf{x}_j)|. \tag{2.37}$$

In (Martín de Diego et al., 2009), a generalization of the previous expression gives raise a wide variety of kernel combinations. Let g be a (convex) function able to meassure the difference of infomation between each two kernels. Then

$$K^*(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{m} \sum_{t=1}^{m} K_i(\mathbf{x}_i, \mathbf{x}_j) + \tau y_i y_j \sum_{t<l} g(K_i(\mathbf{x}_i, \mathbf{x}_j) - K_j(\mathbf{x}_i, \mathbf{x}_j)) \,, \tag{2.38}$$

where $\tau > 0$ generalizes eq. (2.37). Different values of g and τ in eq. (2.38) produce different types of combination schemes. We summarize some cases in Table 2.2.

Another alternative kernel combination scheme proposed in (Martín de Diego et al., 2009) is given by

$$K^*(\mathbf{x}_i, \mathbf{x}_j) = \sum_{t=1}^{m} w_t(\mathbf{x}_i, \mathbf{x}_j) K_t(\mathbf{x}_i, \mathbf{x}_j), \tag{2.39}$$

Table 2.2: Types of kernels combinations for different values of τ and different functions g eq. (2.38).

	Method	τ	$g(x)$	Number of kernels
AKM	**Average Kernel**	0	-	m
MAKM	**Modified Average Kernel**	>0	$g(x)=1$	m
AV	**Absolute Value**	>0	$g(x)=\|x\|$	m
PO	**Pick Out**	$1/2$	$g(x)=\|x\|$	2
SM	**Square Method**	>0	$g(x)=x^2$	m

where the $w_t(\mathbf{x}_i, \mathbf{x}_j)$ are nonlinear functions and $\mathbf{x}_i, \mathbf{x}_j$ are data points in the sample. In this kernel combination procedure it is common to assume that $K_t(\mathbf{x}_i, \mathbf{x}_j) \in [0,1]$ $\forall$ i, j (otherwise they can be scaled). Notice that if $w_t(\mathbf{x}_i, \mathbf{x}_j) = \mu_t$ for $t = 1, \dots, m$ (the functions w_t are constants) the method reduces to calculate the linear combination.

$$K^*(\mathbf{x}_i, \mathbf{x}_j) = \sum_{t=1}^{m} \mu_t K_t(\mathbf{x}_i, \mathbf{x}_j). \tag{2.40}$$

When $\mu_t = 1/m$ in eq. (2.40) and $\tau = 0$ in eq. (2.38) both schemes are equivalent.

Combination schemes based in eqs. (2.38) and (2.39) may lead to indefinite combination matrices. Therefore, to make this methodology useful to train a SVM, the final matrix $K^*\big|_{\mathbf{x}}$ whose elements are given by $(K^*\big|_{\mathbf{x}})_{ij} = \mathbf{K}(\mathbf{x}_i, \mathbf{x}_i)$ has to be projected onto the cone of positive definite matrices. Several techniques have been proposed for this purpose in the literature (see (Muñoz and Martín de Diego, 2006)) but it seems that there is not a universally best method.

An alternative combination based in the semidefinite programming (SDP) is proposed in (Lanckriet et al., 2004). The idea is to optimize the weights of the linear combination in eq. (2.40) maximizing the margin between the classes of the problem and subject to thee conditions: the matrix $K^*\big|_{\mathbf{x}}$ must belong to the cone of positive definite matrices, $\mu_t \geq 0$ and $\sum_{i=1}^{n} \mu_t = 1$. The advantage of this method is that the final matrix of the combination can be directly used to train a SVM (since it is semi positive definite). However, as we will show in Example 2.7, this combination does not obtain as good performance as those in eq. (2.37).

Other alternative in kernel combination (Muñoz et al., 2006) is to define kernels that acts locally depending on the area of the space where the points are located. The idea is to use the scheme in eq. (2.39) and to define the functions $w_t(\mathbf{x}_i, \mathbf{x}_j) = h_t(\mathbf{x}_i)h_t(\mathbf{x}_j)$ where h_t is some type of indicator function. For instance for a problem where the data points are located in two specific areas a suitable kernel is given by

$$K^*(\mathbf{x}_i, \mathbf{x}_j) = w_1(\mathbf{x}_i, \mathbf{x}_j) K_1(\mathbf{x}_i, \mathbf{x}_j) + w_2(\mathbf{x}_i, \mathbf{x}_j) K_2(\mathbf{x}_i, \mathbf{x}_j)\,, \tag{2.41}$$

where for $t = 1, 2$,

$$h_t(\mathbf{x}) = \begin{cases} 1 & \text{if } \|\mathbf{x} - \mathbf{c}_t\|^{1/2} \leq r_t\,, \\ e^{-\gamma\left(\|\mathbf{x}-\mathbf{c}_t\|^2 - r^2\right)} & \text{if } \|\mathbf{x} - \mathbf{c}_t\|^{1/2} > r_t\,, \end{cases} \tag{2.42}$$

being $\| \cdot \|$ the Euclidean distance, $\mathbf{x}$ a sample point, $\mathbf{c}_t \in X$ is the center of each area (circular in this case) and $r_t > 0$ is the radius. Parameter $\gamma > 0$ is fixed in order to obtain a fast transition from 0 to 1.

In this case, due to the particular structure of the defined kernel the solution is given (Muñoz et al., 2006) by:

$$f(x) = \sum_{x_i \in A_1} \alpha_i K_1(\mathbf{x}, \mathbf{x}_i) + \sum_{\mathbf{x}_j \in A_2} \alpha_j K_2(\mathbf{x}, \mathbf{x}_j) + b \tag{2.43}$$

Notice that now K^* behaves like K_1 in the domain of indicator function h_1 and like K_2 in the domain of indicator function h_2. When both K_1 and K_2 are linear kernels the combination K^* in eq. (2.41) is known as Railway Kernel.

To conclude this section we include a real example where the kernel combinations described above are compared.

Example 2.7 (Single kernels and combinations in practice). In this example we deal with a database from the UCI Machine Learning Repository [3]: the Breast Cancer data set (Mangasarian and Wolberg, 1990). The data set consists of 683 observations with 9 features each.

Consider three single kernels: linear, polynolyal of degree 2, and a Gaussian kernel of parameter 1. We compare the classification performance of a SVM with this three kernels and with some of the combination schemes described above. In particular we use the AKM, MaxMin, AV, SDP and the Railway Kernel methods. When necesary we transform the kernel matrices to positive definite via the Positive Eigenvalue Transformation described in (Muñoz and Martín de Diego, 2006). In addition we train two more SVM classifiers built using Gaussian kernels. For the first classifier (SVM$_1$) the parameter is choosen as the inverse of the dimension of the data. For the second (SVM$_2$), ρ and and C (the penalization term) are choosen following (Keerthi and Lin, 2003).

[3] http://archive.ics.uci.edu/ml/

Table 2.3: Percentage of missclassified data, and percentage of support vectors for the cancer data set. Standard deviations in brackets.

Method	Train Error	Test Error	% Support Vectors
Polinomial	0.1 (0.1)	7.8 (2.5)	8.3 (0.8)
Gaussian	0.0 (0.0)	10.8 (1.7)	65.6 (1.0)
Linear	2.6 (0.5)	3.7 (1.8)	7.1 (0.8)
AV	2.4 (0.3)	3.1 (1.3)	2.9 (0.4)
AKM	1.3 (0.2)	3.3 (1.4)	31.1 (0.8)
Max-Min	0.7 (0.1)	2.9 (1.4)	25.3 (0.6)
SDP	0.0 (0.0)	6.2 (1.6)	65.5 (1.9)
RK	2.5 (0.3)	2.9 (0.4)	18.6 (3.6)
SVM$_1$	0.1 (0.1)	4.2 (1.4)	49.2 (1.0)
SVM$_2$	0.0 (0.0)	2.9 (1.6)	49.2 (1.0)

Table 2.7 shows the performance of the three single kernels, the combinations and the two optimized SVMs. The results are averaged over 10 runs and the 80% of the data are used to train the SVMs and the remaining 20% to calculate the test errors. The combination methods AV, AKM, Max-Min and RK improve any single kernel used to build them. In particular, the Max-Min and RK methods obtain the best performance of the experiment together with the SVM_1 that uses and optimized Gaussian kernel. The SDP is, however, the worse combination procedure. It improves the polynomial and Gaussian kernels but does not outperform the classification errors of a linear kernel.

• • •

Chapter 3

Representing Functional Data in Reproducing Kernel Hilbert Spaces

Abstract

Functional data are difficult to manage for many traditional statistical techniques given their very high (or intrinsically infinite) dimensionality. The reason is that functional data are essentially functions and most algorithms are designed to work with (low) finite-dimensional vectors. Within this context we propose techniques to obtain finite-dimensional representations of functional data. The key idea is to consider each functional curve as a point in a general function space and then project these points onto a Reproducing Kernel Hilbert Space with the aid of Regularization theory. In this chapter we describe the projection method, analyze its theoretical properties and develop an strategy to select an appropriate RKHSs to represent the functional data.

Keywords: Functional Data, Reproducing Kernel Hilbert Spaces, Regularization Theory, Subspace Information Criterion.

3.1 Introduction

The field of Functional Data Analysis (FDA) (Ramsay and Silverman, 2006) (Ferraty and Vieu, 2006) deals naturally with data of very high (or intrinsically infinite) dimensionality. Typical examples are functions describing physical processes, genetic data, control quality charts or spectra of data in Chemometrics.

In practice each functional datum is given by a data set $f_n = \{(x_i, y_i) \in X \times Y\}_{i=1}^n$, where X is the space of input variables and, in most cases, $Y = \mathbb{R}$. The first task in any FDA methodology is to transform the data set f_n into a function $f : X \to Y$ and then to apply some generalized multivariate procedure able to cope with functions. Of course n, the number of data points which can be recorded, is finite while an accurate description of the underlying function would require an infinite number of observations. Therefore the choice of a particular f will be done, in general, by selecting it from an infinite collection of alternative models. This is the typical context in which ill-posed problems arise (Tikhonov and Arsenin, 1977).

Most FDA approaches choose an orthogonal basis of functions $B = \{\phi_1, \ldots \phi_d\}$ ($d \in \mathbb{N}$), where each ϕ_j belongs to a general function space (usually $L^2(X)$) and then represent each functional datum by means of a linear combination in $Span(B)$ (Ramsay and Silverman, 2006). Usual choices for functions in B are Fourier, Wavelets or B-splines functions.

Our approach in this thesis will be to evaluate the goodness of fit of a particular function to a given functional datum by means of some "loss function" $L(y, f(x))$. The seeked function will be the minimizer of the empirical error $\frac{1}{n}\sum_{i=1}^n L(y_i, f(x_i))$ in a hypothesis space $\mathcal{H}$. It is well known that to achieve well-posedness of this problem and uniform convergence of the empirical error to the generalization error defined by $\int_{X \times Y} L(y, f(x)) d_\nu(x, y)$ (where ν is some probability measure on $X \times Y$), imposing compactness in $\mathcal{H}$ is a sufficient condition (Cucker and Smale, 2001; Moguerza and Muñoz, 2006). A way to achieve this is to use regularization theory and the natural function spaces to use are the Reproducing Kernel Hilbert Spaces (RKHSs). Following this approach we propose a finite-dimensional representation for functional data based on a particular projection of the original functions onto a Reproducing Kernel Hilbert Space (RKHS).

As we detailed in Chapter 2,RKHSs (Cucker and Smale, 2001; Wahba, 2003) are characterized by a generalized covariance function called kernel and the approximating function will be a linear combination of its eigenfunctions. Under general rather conditions we can build kernels from orthonormal basis of functions (Rakotomamonjy and Canu, 2005). In addition, we can directly choose the kernel (see Section 3.4 for details); in Section 3.2 we propose a method to approximate the eigenfunctions of a given kernel as a previous step to obtain the proposed functional data representation. To focus on the kernel makes accessible a wider class of basis of functions to represent the functional data. In this sense our approach constitutes a generalization of the usual FDA setting.

The choice of the kernel in regularization methods is a relevant problem that has been extensively studied in the literature. We refer to (Keerthi and Lin, 2003; Lanckriet et al., 2004; Moguerza and Muñoz, 2006) for some references in the classification context and to (Cherkassky and Ma, 2004) in regression problems. In this chapter we will make use of the Subspace Information Criterion (SIC) (Sugiyama and Ogawa, 2001; Sugiyama and Muller, 2002) to select the kernel that generates the RKHS. The SIC is designed to approximate the Generalization Error in general regularization methods and it has been proven to be very competitive as model selection criteria compared to other model selection criteria choices (Sugiyama and Ogawa, 2002). In this chapter we will show how to adapt it to our particular problem and we will propose an alternative to improve it in practice.

This chapter is organized as follows. In next section we show how to project functional data onto RKHSs. We propose in Section 3.4 a variation of the SIC designed to select optimal space where project a set curves. In Section 3.5 we study the truncated error of the proposed projection and we conclude is Section 3.6 with some final remarks.

3.2 Representing Functional Data in Reproducing Kernel Hilbert Spaces

As we studied in Chapter 2, a Hilbert function space H is a RKHS where all the (linear) evaluation functionals ($\mathcal{F}_x : H \to \mathbb{R}$ such that $\mathcal{F}_x(f) = f(x)$, where $x \in X$) are bounded (equivalently continuous). By the Riesz representation theorem, for each $x \in X$ there exists $h_x \in H$ such that for every $f \in H$ it holds that $f(x) = \langle h_x, f \rangle$, where $\langle , \rangle$ denotes the inner product in H. The RKHS H is characterized by a continuous symmetric positive definite function $K : X \times X \to \mathbb{R}$ named Mercer Kernel or reproducing kernel for H (Aroszajn, 1950). The elements of H, $\mathcal{H}_K$ in the sequel, can be expressed as finite linear combinations of the form $h = \sum_s \lambda_s K(x_s, \cdot)$ where $\lambda_s \in \mathbb{R}$ and $x_s \in X$.

Consider the linear integral operator L_K (associated to the kernel function K) defined by $L_K(f) = \int_X K(\cdot, s) f(s) ds$. When X is compact and K continuous, then L_K has a countable sequence of eigenvalues $\{\lambda_j\}$ and (orthonormal) eigenfunctions $\{\phi_j\}$ and K can be expressed by $K(x, y) = \sum_j \lambda_j \phi_j(x) \phi_j(y)$ where the convergence is absolute and uniform (Mercer's theorem (Mercer, 1909)).

3.2.1 Projecting functional data onto RKHSs

Let X be a compact space or manifold in an Euclidean Space and $Y = \mathbb{R}$. Let ν be a Borel probability measure defined on $X \times Y$. In the sequel we will assume that ν is non degenerate. Denote by f_n a sample curve drawn form ν identified with a data set $\{(x_i, y_i) \in X \times Y\}_{i=1}^n$. Define $f_\nu : X \to Y$,

$$f_\nu = \int_X y d_\nu(y \,|\, x), \tag{3.1}$$

where $d_\nu(y \,|\, x)$ is the the conditional probability measure on Y. Thus f_n is a sample version of size n of f_ν. In practice we are usually given a set of curves $\{f_{n,1}, \ldots, f_{n,m}\}$ where each sample curve$f_{n,l}$ is drawn, in the most general case, from a different measure ν_l and it is identified with a data set $\{(x_i, y_{il}) \in X \times Y\}_{i=1}^n$. For simplicity in notation we will assume that the vector $\mathbf{x} = (x_1, \ldots, x_n)^T$ is common for all the curves, as it is the habitual case in the literature (Ramsay and Silverman, 2006).

Next we develop a procedure to approximate f_ν using the associated f_n.

Definition 3.1. *Let X be a compact space or manifold in and Euclidean Space, $Y = \mathbb{R}$ and ν a Borel probability measure defined on $X \times Y$. Let $f_n = \{(x_i, y_i) \in X \times Y\}_{i=1}^n$ be a sample curve drawn form ν and consider f_ν defined in eq. (3.1). Let $K : X \times X \to \mathbb{R}$ be a Mercer kernel and $\mathcal{H}_K$ its associated RKHS. Then we define the* ***Regularized γ-Projection*** *of f_ν onto $\mathcal{H}_K$ associated to the sample curve f_n as*

$$f^*_{K,\gamma,n} = \Pi_{K,\gamma,n}(f_\nu) = \arg \min_{f \in \mathcal{H}_K} \frac{1}{n} \sum_{i=1}^n (f(x_i) - y_i)^2 + \gamma \|f\|_K^2 \,, \tag{3.2}$$

where $\gamma > 0$ and $\|f\|_K$ represents the norm of the function f in $\mathcal{H}_K$.

Below, we show that $f^*_{K,\gamma,n} \in span\{K(x, x_i)\}$, then for every $x \in X$ we have that $f(x) = \sum_{j=1}^n \alpha_j K(x_j, x)$, for appropriate $x_j \in X$ and $\alpha_j \in \mathbb{R}$. Thus, calling $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)^T$, $\mathbf{x} = (x_1, \ldots, x_n)^T$, and $K\big|_{\mathbf{x}}$ the matrix whose components are $(K\big|_{\mathbf{x}})_{ij} = K(x_i, x_j)$, we have $\|f^*_{K,\gamma,n}\|_K^2 = \sum_{i=1}^n \sum_{i=1}^n \alpha_i \alpha_j K(x_i, x_j) = \boldsymbol{\alpha}^T K\big|_{\mathbf{x}} \boldsymbol{\alpha}$. Eq. (3.2) quantifies the balance between the fitness of the function to the data (measured by $\frac{1}{n} \sum_{i=1}^n (f(x_i) - y_i)^2$) and the complexity of the solution (measured by $\|f\|_K^2$). Notice that that in eq. (3.2) we denote by $f^*_{K,\gamma,n}$ and $\Pi_{K,\gamma,n}(f_\nu)$ the estimated curve. While we will use the first notation in the sequel, we include the second to remark that the obtained curve is the result of projecting f_ν onto the $\mathcal{H}_K$ using f_n.

Definition 3.1 can be generalized in several directions. The first term can be replaced by a different loss function. For instance we could consider $L(x, y) = |x - y|$, or any linear convex combination of $L(x, y) = |x - y|^p$ loss functions. Other possible choice for the loss function in (3.2) is the so-called ϵ-insensitive loss function, given by $L(y_i, f(x_i)) = (|f(x_i) - y_i| - \varepsilon)_+$, $\varepsilon \geq 0$ (used by the Support Vector Machine for regression (Smola and Schölkopf, 2003)). The conditions for a loss function $L : \mathbb{R} \times Y \to \mathbb{R}^+$ to guarantee uniform stability in the regularization approach are: 1) L is a Lipschitz function, 2) There exists a constant C such that $L(0, y) \leq C \ \forall y \in Y$. (see (Mukherjee et al., 2002) and (Bousquet and Elisseeff, 2002) for further details and implications).

Regarding the second term in (3.2), we can replace $\|f\|_K^2$ with a general convex positive functional $\Omega(f)$. There are two frequent choices. In the first case, we consider $\|Lf\|^2$, where L is a linear differential operator (Ramsay and Silverman, 2006; Chen and Haykin, 2002). In particular the Green's function of the operator L^*L (L^* the adjoint operator to L) satisfies the condition of being a valid kernel and thus, this case may be seen as a particular case in the frame of the RKHS formalism. In the second case we consider $\|Pf\|^2$, where P is a projection operator onto a finite dimensional subspace (Wahba, 1990). The underlying idea is to choose two orthogonal sets of basis functions $\{\phi_k\}$ and $\{\psi_l\}$ in such a way that the $\{\phi_k\}$ (small in number) can provide a first approximation to the function, and the $\{\psi_l\}$ (usually much larger in number) are able to provide a larger accuracy in approximation. P annihilates some of the $\{\psi_k\}$ when using $\|Pf\|^2$. For further details, see (Ramsay and Silverman, 2006), chapter 5. Notice that we need in every case to work with a bounded linear operator to guarantee that we can apply the Riesz representation theorem and be able to define a kernel in each case (see (Wahba, 2003) for additional possibilities).

Concerning the solution to eq. (3.2), by the Representer Theorem (Theorem 2.5), the minimizer $f^*_{K,\gamma,n}$ of the functional optimization problem in eq. (3.2) exists, is unique and admits a representation of the form

$$f^*_{K,\gamma,n}(x) = \sum_{i=1}^{n} \alpha_i K(x_i, x), \quad \forall \mathbf{x} \in X\,, \tag{3.3}$$

where now the x_i points are the sample data (components of the vector $\mathbf{x}$) and the coefficients $\alpha_i \in \mathbb{R}$ are the solutions of the linear system:

$$(\gamma n \mathbf{I}_n + K\big|_{\mathbf{x}})\boldsymbol{\alpha} = \mathbf{y}, \tag{3.4}$$

where $\mathbf{I}_n$ the identity matrix of dimension $n \times n$, $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)^T$ and $\mathbf{y} = (y_1, \ldots, y_n)^T$. By solving eq. (3.4) we get a closed expression for $f^*_{K,\gamma,n}$, the minimizer of problem (3.2). When $\gamma = 0$ we can interpret eq. (3.2) as the orthogonal projection of f_ν onto $\mathcal{H}_K$ via f_n as follows.

Proposition 3.1. *Let X be a compact space or manifold in an Euclidean Space, $Y = \mathbb{R}$ and ν a Borel probability measure defined on $X \times Y$. Let $f_n = \{(x_i, y_i) \in X \times Y\}_{i=1}^n$ be a sample curve drawn form ν and consider f_ν defined in eq. (3.1). Let $K : X \times X \longrightarrow \mathbb{R}$ be a continuous symmetric positive definite kernel with associated integral operator L_K with eigenfunctions $\{\phi_1, \phi_2, \dots\}$ and eigenvalues $\{\lambda_1, \lambda_2, \dots\}$. Then, when $\gamma = 0$, the projected curve $f^*_{K,0,n}$ obtained by solving problem (3.2) can be written by*

$$f^*_{K,0,n} = \Pi_{K,0,n}(f_\nu) = \sum_{j=1} \lambda_j(\boldsymbol{\alpha}^T \boldsymbol{\phi}_{j,\mathbf{x}})\phi_j(x), \tag{3.5}$$

where $\boldsymbol{\alpha}$ is the solution to eq. (3.4) and $\boldsymbol{\phi}_{j,\mathbf{x}} = (\phi_j(x_1), \dots, \phi_j(x_n))^T$. In addition

$$f^*_{K,0,n} = \sum_j \lambda_j(\boldsymbol{\alpha}^T \boldsymbol{\phi}_{j,\mathbf{x}})\phi_j \underset{n\to\infty}{\longrightarrow} \sum_j \lambda_j \langle f_\nu, \phi_j\rangle \phi_j, \tag{3.6}$$

where the convergence is uniform in X.

By the Spectral Theorem (Conway, 1990) $L_K(f_\nu) = \sum_j \lambda_j \langle f_\nu, \phi_j\rangle\phi_j$. Thus $f^*_{K,0,n}$ converges uniformly to $L_K(f_\nu)$ the orthogonal projection of f_ν onto $\mathcal{H}_K$. When $\gamma > 0$, $\Pi_{K,\gamma,n}$ can also be interpreted as a projection of f_ν onto $\mathcal{H}_K$ as it is shown in next proposition.

Proposition 3.2. *Under the same assumptions as in Proposition 3.1, when $\gamma > 0$, the projected curve $f^*_{K,\gamma,n}$, given by the minimization of eq. (3.2) can also be interpreted as a projection of f_ν onto $\mathcal{H}_K$ and*

$$f^*_{K,\gamma,n} = \sum_j \lambda_j(\boldsymbol{\alpha}^T \boldsymbol{\phi}_{j,\mathbf{x}})\phi_j \underset{n\to\infty}{\longrightarrow} \sum_j \lambda_j \langle f_\nu, \phi_j\rangle' \phi_j. \tag{3.7}$$

where the convergence is uniform in X, $\boldsymbol{\alpha}$ is the solution to eq. (3.4), $\{\lambda_j\}$ are the eigenvalues of L_K, $f_{\mathbf{x}} = (f(x_1), \dots, f(x_n))^T$, $\boldsymbol{\phi}_{j,\mathbf{x}} = (\phi_j(x_1), \dots, \phi_j(x_n))^T$ and $\langle f, \phi_j\rangle' = \beta_j\langle f, \phi_j\rangle$ for appropriate $\beta_j \in \mathbb{R}$.

Eq. (3.7) generalizes eq. (3.1) as the Ridge Regression generalizes the Least Squares regression (see (Swindel, 1981) for further details concerning the geometry of ridge regression).

In eq. (3.3) the projected curve $f^*_{K,\gamma,n}$ is expressed, via the vector $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_n)^T$, as a linear combination in $Span\{K(x, x_i)\}$. In addition in eq. (3.7) the same curve can be seen as a linear combination of the eigenfunctions of L_K. Next theorem introduces a practical manner to estimate this representation, that is the weights $\lambda_j(\boldsymbol{\alpha}^T \boldsymbol{\phi}_{j,\mathbf{x}})$ in eq. (3.7).

Theorem 3.1. *Let X be a compact space or manifold in an Euclidean Space, $Y = \mathbb{R}$ and ν a Borel probability measure defined on $X \times Y$. Let $f_n = \{(\boldsymbol{x}_i, y_i) \in X \times Y\}_{i=1}^n$ be a sample curve drawn form ν and consider f_ν defined in eq. (3.1). Let $K : X \times X \longrightarrow \mathbb{R}$ be a continuous symmetric positive definite kernel with associated integral operator L_K with eigenfunctions $\{\phi_1, \phi_2, \dots\}$ and eigenvalues $\{\lambda_1, \lambda_2, \dots\}$. Then, the projected curve $f^*_{K,\gamma,n}$, given by the minimization of (3.2), can be expressed as*

$$f^*_{K,\gamma,n}(\boldsymbol{x}) = \sum_j \lambda^*_j \phi_j(\boldsymbol{x}), \tag{3.8}$$

where λ^*_j are the weights of the projection of $f^*_{K,\gamma,n}(\boldsymbol{x})$ onto the function space generated by the eigenfunctions of L_K. In practice, when a finite sample is available, the first $d = rank(K\big|_{\boldsymbol{x}})$ weigths λ^*_j can be estimated by

$$\hat{\lambda}^*_j = \frac{l_j}{\sqrt{n}} (\boldsymbol{\alpha}^T \boldsymbol{v}_j) \,, \tag{3.9}$$

for l_j the j-th eigenvalue of the matrix $K\big|_{\boldsymbol{x}}$, $\boldsymbol{v}_j = (v_{j1}, \dots, v_{jn})^T$, the j-th eigenvector and $\boldsymbol{\alpha}$ the solution to eq. (3.4).

Hence two possible finite representations are available for the projection of f_ν given f_n. The first one, in eq. (3.3) by the vector $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_n)^T$, will be named as "Kernel Expansion". The second, given in eq. (3.8) by the vector $\hat{\boldsymbol{\lambda}}^* = (\hat{\lambda}^*_1, \dots, \hat{\lambda}^*_d)^T$ will be denominated as "RKHS representation". Next two remarks compare both representations in terms of their stability in the input variables.

Definition 3.2. *Let X be a compact space or manifold in an Euclidean Space, $Y = \mathbb{R}$ and ν a Borel probability measure defined on $X \times Y$. Let $f_n = \{(\boldsymbol{x}_i, y_i) \in X \times Y\}_{i=1}^n$ be a sample curve drawn form ν. We say that $f^\epsilon_n = \{(x_i, y^\epsilon_i)\}_{i=1}^n$ is a ϵ-perturbed curve of f_n if*

$$\frac{|y_i - y^\epsilon_i|}{|y_i|} \leq \epsilon \;\; for \;\; all \;\; i = 1, \dots, n. \tag{3.10}$$

Definition 3.3. *Under the same assumptions as in Definition 3.2, consider a set of continuos functions $B = \{\varphi, \dots, \varphi_q\}$ on X where $q \leq n$. Let f_n be a sample curve, f_ν defined in eq. (3.1) and f^ϵ_n an ϵ-perturbed curve of f_n. Let $\Pi_{B,n} : L^2_\nu(X) \longrightarrow Span(B)$ be a general curves projection method onto $Span(B)$ using a sample curve of size n and let*

$$\Pi_{B,n}(f_\nu) = \sum_j \beta_j \varphi_j \;\; and \;\; \Pi^\epsilon_{B,n}(f_\nu) = \sum_j \beta^\epsilon_j \varphi_j, \tag{3.11}$$

be two projections of f_ν using f_n and f^ϵ_n respectively. Then we say that the representation of f_n

given by $\beta = (\beta_1, \ldots, \beta_q)^T$ is ϵ-stable in the input variables if

$$\frac{|\beta_j - \beta_j^\epsilon|}{|\beta_j|} \leq \epsilon \;\; for \;\; all \;\; j = 1, \ldots, q. \tag{3.12}$$

Theorem 3.2. *Under the conditions described in Theorem 3.1, the representation of $f^*_{K,\gamma,n}$ given by $\hat{\boldsymbol{\lambda}}^* = (\hat{\lambda}_1^*, \ldots, \hat{\lambda}_d^*)^T$, where $\hat{\lambda}_j$ is estimated in eq. (3.9) and $d = rank(K\big|_{\mathbf{x}})$ is ϵ-stable in the input variables.*

Theorem 3.3. *Under the conditions described in Theorem 3.1 the representation of $f^*_{K,\gamma,n}$ in terms of the vector $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)^T$, where $\boldsymbol{\alpha}$ is the solution to eq. (3.4) is not ϵ-stable in the input variables.*

Next we include and illustrative example to show the implications of the two previous theorems in a real example.

Example 3.1. We consider two similar functional data curves to illustrate the behavior of the Kernel expansion (given in (3.3)) and the RKHS representation system (given in (3.8)). The two curves are temperatures curves corresponding to daily series averaged over the period from 1960 to 1994 in Canada ((Ramsay and Silverman, 2006), Chapter 1), and correspond to the cities “St. Johns" and “Halifax". We consider the Gaussian kernel $K(\mathbf{x}, \mathbf{y}) = e^{-\rho\|x-y\|^2}$ with $\rho = 10^{-4}$ and $\gamma = 1$) and obtain the kernel expansion and the RKHS representation for both curves. In the experimental chapter we will detail how to fix the pairs of parameters (σ, γ).

Figure 3.1, left (upper and lower), shows the original curves and their projections onto the function space $\mathcal{H}_K$ generated by the eigenfunctions of K. The two central plots in Figure 3.1 show the kernel expansion representation for both curves and it is apparent they are quite different, despite the fact the two curves are similar. Figure 3.1, right, shows the RKHS representations for both curves and now they look similar, in agreement with Theorem 3.2. In addition, we can see that the RKHS representations is representing the curves in a no more than a 10-dimensional space (essentilly 4), which agrees with the result obtained by the dimensionality test proposed in (Hall and Vial, 2006). We can therefore conclude that the RKHS representation is robust against the presence of noise in the data in agreement with Theorem 3.2.

• • •

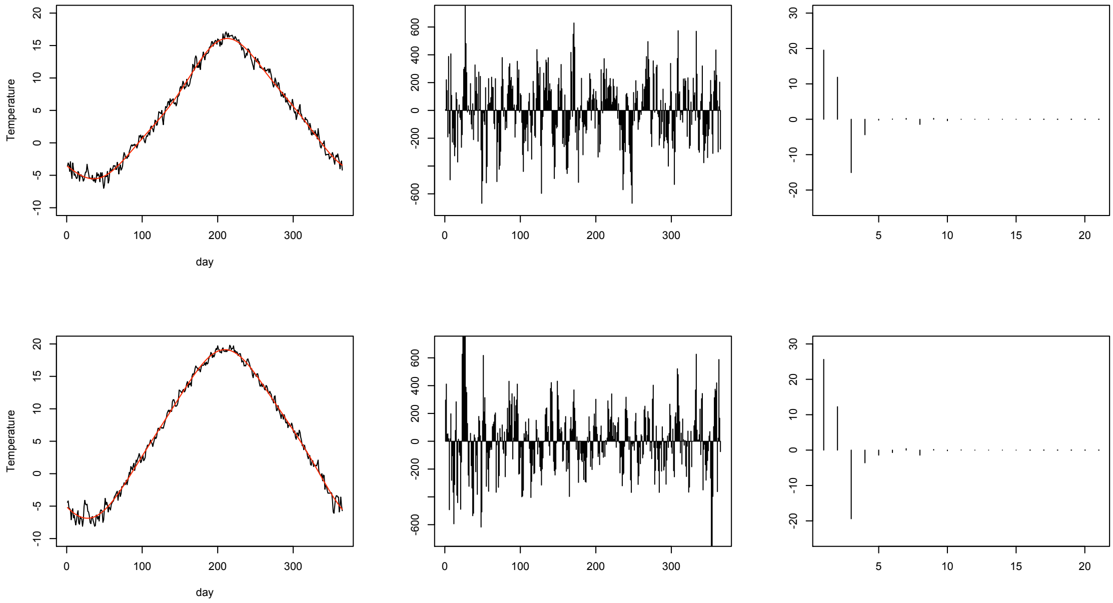


Figure 3.1: Two Canadian curves and their Kernel expansion and RKHS representations.

3.3 Distance measures induced by the projections of functional data onto RKHSs

In Functional Data Analysis we are generally given a set of curves $\{f_{n,1}, \ldots, f_{n,m}\}$ where each sample curve$f_{n,l}$ is identified with a data set $\{(x_i, y_{il}) \in X \times Y\}_{i=1}^{n}$. In practical cluster and classification problems n is generally very large. This makes the functional data to be not tractable for most algorithms that are commonly designed to work either with (small) finite-dimensional vectors or with distances matrices. In this context, to determine an appropriate distance matrix between the curves (with dimensions $m \times m$ where $m \ll n$) makes the problem solvable in practice.

Several methods have been proposed in the literature to define distances between curves. For instance the Dynamic Time Warping (Sakoe and Chiba, 1978) calculates the dissimilarity between two series by warping them before calcuating its Euclidean distance. Other approach followed in (Ferraty and Vieu, 2006) is to define some semi-metric as measure of similarity for the curves. In any of the previous approaches similarities/disimilarities can be transformed to distances. See (Gower, 1986) for details.

In this section we study the metric for curves induced by the projection defined in eq. (3.1). The proposed metric will be determined by K and γ and we will be the input of classification and cluster procedures. Notice that many kernels can determine the same metric (Burges, 1998) which in practice is not a problem for our purposes.

Definition 3.4. *Let X be a compact space or manifold in and Euclidean Space, $Y = \mathbb{R}$ and ν, μ two Borel probability measures defined on $X \times Y$. Let $f_n = \{(x_i, y_i) \in X \times Y\}_{i=1}^n$ and $g_n = \{(x_i, y_i') \in X \times Y\}_{i=1}^n$ two sample curves drawn form ν and μ respectively and let f_ν, g_μ defined following eq. (3.1). Let $K : X \times X \to \mathbb{R}$ be a Mercer kernel and $\mathcal{H}_K$ its associated RKHS. Then we define the* ***Empirical Regularized*** *γ* ***Inner Product*** *between f_ν, g_μ as*

$$\langle f_\nu, g_\mu \rangle_{K,\gamma,n} = \langle \Pi_{K,\gamma,n}(f_\nu), \Pi_{K,\gamma,n}(g_\mu) \rangle_K \tag{3.13}$$

where given h_1 and h_2, $\langle h_1, h_2 \rangle_K = \sum_j \lambda_j^{-1} a_j b_j$ for $h_1 \in \sum_j a_j \phi_j \in \mathcal{H}_K$ and $h_2 = \sum_j b_j \phi_j \in \mathcal{H}_K$ being $\{\lambda_j\}$ the eigenvalues of L_K.

Notice that, given a kernel K, we define the inner product of f_ν and g_μ as the inner product of their projections onto $\mathcal{H}_K$. In practice, using eq. (3.8) and the definition of $\langle \cdot, \cdot \rangle_K$ it is straightforward to check that an estimator of $\langle f_\nu, g_\mu \rangle_{K,\gamma,n}$ is given by

$$\sum_{j=1}^n l_j^{-1} (\hat{\lambda}_{fj}^* \hat{\lambda}_{gj}^*), \tag{3.14}$$

where l_j is the j-th eigenvalue of $K\big|_{\mathbf{x}}$ and $\hat{\lambda}_{fj}^*, \hat{\lambda}_{gj}^*$, the components of the "RKHS" representation of f_ν and g_μ, are given by eq. (3.9).

Definition 3.5. *Given the elements of Definition 3.4 we define the* ***Empirical Regularized*** *γ* ***Distance*** *for two curves f_ν, g_μ as*

$$d_{K,\gamma,n}(f_\nu, g_\mu) = \langle f_\nu, g_\mu \rangle_{K,\lambda,n} + \langle f_\nu, g_\mu \rangle_{K,\lambda,n} - 2\langle f_\nu, g_\mu \rangle_{K,\lambda,n} \tag{3.15}$$

This distance can be estimated by replacing eq. (3.14) in eq. (3.15). Hence given a set of curves, the distance defined in eq. (3.15) can be estimated for each pair of curves obtaining a distance matrix $\mathbf{D}$ that can be used as the input of cluster or classification algorithms. We conclude this section with an illustrative example. In Chapter 6 we will show more experiments and real applications.

Example 3.2. In Statistics it is usual to reduce the dimension of high dimensional data before affording cluster or classification tasks. In FDA this is achieved by using the Functional Principal Components (FPCA) (Ramsay and Silverman, 2006; Hall and Vial, 2006). As in the multivariate case, this technique makes use of the data covariance function to determine the subspace where the data are projected. This subspace is spanned

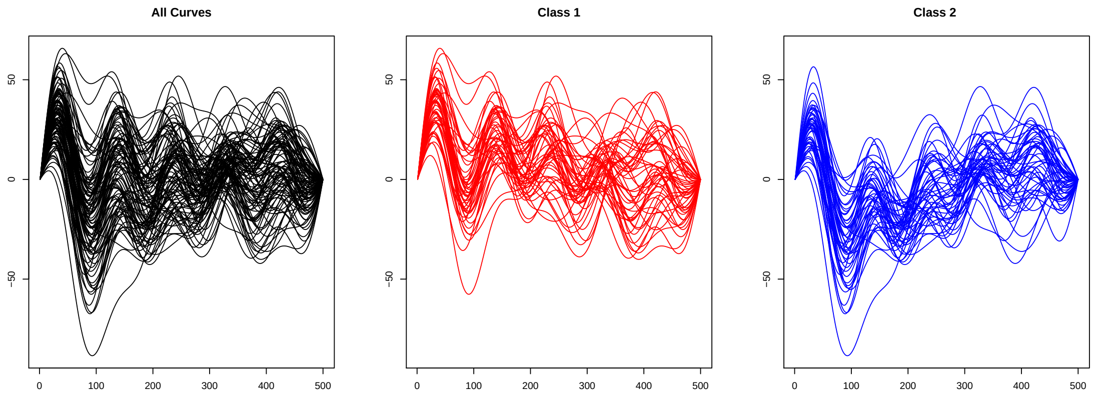


Figure 3.2: Left: all curves together. Center: Class 1 curves. Right Class 2 curves.

by the data covariance eigenfunctions and it is always a RKHS (see (Rakotomamonjy and Canu, 2005)). Within this setting, FPCA can be considered a particular case of our methodology.

The choice of the data covariance S as kernel K in eq. (3.2) is justified in certain theoretical cases (see (James and Sugar, 2003)). In practice, more general kernels can be considered. The next example illustrates this situation in a clustering problem.

Consider two families of 10 dimensional curves sampled at 500 points:

- Class 1: $c(x) = \sum_{j=1}^{10} a_j \phi_j(x) = sin(j\pi x)$, where $a_i \sim N_{10}(\mu_1, \Sigma)$
- Class 2: $c(x) = \sum_{j=1}^{10} b_j \phi_j(x) = sin(j\pi x)$, where $b_j \sim N_{10}(\mu_2, \Sigma)$

with $x \in [0,1]$ and for $\mu_1 = (8,8,1,2,3,4,5,6,7,8)$, $\mu_2 = (-8,-8,1,2,3,4,5,6,7,8)$, and $\Sigma = diag(1,150,150,10,10,10,10,10,10,10)$. For our experiment, we generated 50 curves of each family (see Figure 3.2).

We compare the RKHS representation system ($\boldsymbol{\lambda}^*$) using the data covariance and a generalized covariance: a Gaussian kernel. To this aim we first separate (automatically) the curves using row the data. We performe 10 runs of a k-means algorithm (with 2 centroids) and a hierarchical cluster by using the Ward method. The misclassification errors are 25.2% and 24% respectively.

By using FPCA, the first two principal components explain over 80% of the variability. This two components are plotted in Figure 3 (left). Applying the two previous cluster procedures on this new projection we obtain misclassification errors of 15% (for the

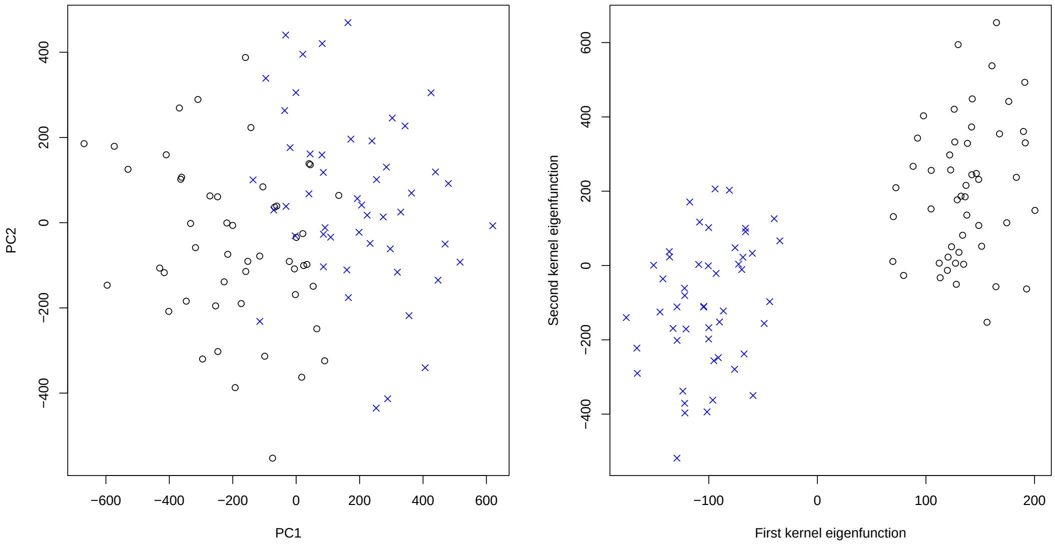


Figure 3.3: Two first FPCA projection (left) and RKHS projections (right).

k-means algorithm) and 18% (for the hierarchical cluster procedure). The dimension reduction improves the results but a large number of curves is still assigned to the wrong class. On the other hand, if the two first projections are achieved by using regularization with the kernel $K(x, y) = e^{-\rho\|x-y\|^2}$ with $\rho = 10$ and regularization parameter $\gamma = 1$ (see Figure 3.3, 0% of errors are obtained with both cluster algorithms, what justify the use of a generalized covariance function. The reason of this improvement is that the kernel K is capturing non linear dependences between the data while the covariance function only deals with linear ones.

• • •

3.4 Model selection in functional data regularization

A central problem in statistics is the selection of appropriate models for the data. In our context, to select a model for a sample curve f_n means to find appropriate K and γ in eq. (3.2).

A typical manner to afford the model selection problem is to minimize some measure of the predictive error, for instance, the averaged difference between the estimated and the true values of some test points contained in the data: the traditional cross validation (CV), its generalized version (GCV) (Craven and Wahba, 1979) or the Cp measure

(Mallows, 1973) constitute some examples. However the optimality of this approach is not guaranteed since the real generalization capacity of the models is not estimated. In contrast, model selection criteria that deals with the generalization error have also been proposed: from the point of view of the information theory the Akaike Information Criterion (AIC) (Akaike, 1974) and its corrected modification (cAIC) (Sugiura, 1978) are the most representative cases. From the Bayesian perspective the Bayesian Information Criterion (BIC) (Schwarz, 1978) is a well known example. Other approaches different form the two previous points of view are the structural risk minimization (SRM) (Vapnik, 1995) or the Vapnik measure (VM) (Cherkassky et al., 1999).

In (Sugiyama and Ogawa, 2001) the Subspace information Criterion (SIC) is proposed as a new alternative of model selection. It is very competitive (Sugiyama and Muller, 2002) compared to previous measures and it represents a natural framework for model selection in regularization methods. In this section we will particularize it for the functional data model selection problem and we will propose an alternative to improve it in certain scenarios.

3.4.1 Model selection problem

Let X a compact space or manifold, $Y = \mathbb{R}$ and ν a probability measure over $X \times Y$. Let f_n be a sample curve drawn form ν identified with a data set $\{(x_i, y_i) \in X \times Y\}_{i=1}^n$ and define the target function $f_\nu : X \to Y$ as $f_\nu = \int_X y d_\nu(y \,|\, x)$ for $\nu(y\,|x)$ the conditional measure on Y. In the sequel we will assume that f_ν belongs to $L^2_\nu(X)$ and that f_ν is a bounded function.

Define $\epsilon = y - f_\nu(x)$. Then

$$\mathsf{E}_\nu(\epsilon) = \int_Y (f_\nu(x) - y) d_\nu(y \,|\, x),$$

where E_ν denote the expectation over the measure ν. It is straightforward to check that $\mathsf{E}_\nu(\epsilon) = 0$ and hence the variance of ϵ is given by

$$\mathsf{Var}_\nu(\epsilon) = \int_Y (f_\nu(x) - y)^2 d_\nu(y \,|\, x),$$

where $\mathsf{Var}_\nu(\epsilon)$ denotes the variance over ν. Using the definition of f_ν and because $\mathsf{E}_\nu(\epsilon) = 0$, given the sample points $\{(x_i, y_i) \in X \times Y\}_{i=1}^n$ we have that

$$y_i = f_\nu(x_i) + \epsilon_i, \tag{3.16}$$

where the ϵ_i are unknown additive independent noise components from a distribution with zero mean and variance $\mathsf{Var}_\nu(\epsilon)$. Notice that both, f_ν and $\mathsf{Var}_\nu(\epsilon)$ are totally determined by ν.

Given f_n consider a set of pairs $\{(K, \gamma)\}$, where each $K : X \times X \longrightarrow \mathbb{R}$ is a Mercer kernel function and $\gamma > 0$. This set can be either finite or infinite. In this last case K is commonly defined as a parameter dependent kernel, for instance, a Gaussian kernel.

Let $f^*_{K,\gamma,n}$ be the projected curve obtained via eq. (3.2) using the sample f_n and some γ and K. The model selection problem is stated as finding, for a fixed sample curve f_n, the pair (K^*, γ^*) that minimizes the generalization error defined as

$$\mathsf{E}_\epsilon \left(\int_X (f^*_{K,\gamma,n} - f_\nu)^2 dx \right) = \mathsf{E} \| f^*_{K,\gamma,n} - f_\nu \|^2, \tag{3.17}$$

where E_ϵ denotes the expectation over the noise ϵ. For simplicity in notation in the sequel we will write E instead of E_ϵ. Notice that $f^*_{K,\gamma,n}$ belongs to $\mathcal{H}_{K,n} = Span(\{K(x, x_i)\})$ while, in general, it is common to assume that the function f_ν belongs to $L^2_\nu(X)$.

3.4.2 Subspace Information Criterion (SIC) for functional data regularization

The Subspace Information Criteria (SIC) (Sugiyama and Ogawa, 2001) was proposed as a procedure to give an unbiased estimator of the generalization error in eq. (3.17) in general regularization methods. In this section we follow the general model selection approach described above and we will adapt the SIC to our particular problem in eq. (3.2), resulting a particular case of our general formalism.

Let K be a Mercer kernel function L_K its associated integral operator and $\mathcal{H}_K$ its corresponding RKHS. We first decompose the target function f_ν as follows. Let $f_{\nu,\mathcal{H}_K}$ be the orthogonal projection of f_ν onto $\mathcal{H}_K$ ($f_{\nu,\mathcal{H}_K} = L_K(f_\nu)$, see Proposition 3.1 for details) and define $f^\perp_{\nu,\mathcal{H}_K}$ as

$$f^\perp_{\nu,\mathcal{H}_K} = f_\nu - f_{\nu,\mathcal{H}_K}, \tag{3.18}$$

the orthogonal complement of $f_{\nu,\mathcal{H}_K}$. In this context, the SIC proposed in (Sugiyama and Muller, 2002) can be understood, as we describe below, as a model selection criterion for functional data when $f^\perp_{\nu,\mathcal{H}_K} = 0$ or equivalently when f_ν is assumed to belong to $\mathcal{H}_K$. Although this hypothesis can be relaxed in some cases (see next section for details), we will assume this property for f_ν here and therefore we will refer f_ν by $f_{\nu,\mathcal{H}_K}$.

To define the SIC in our context we first decompose eq. (3.17) in a sum. Let $f^*_{K,\gamma} =$

$\mathsf{E}(f^*_{K,\gamma,n})$ (see Proposition 3.2). Then the generalization error of $f^*_{K,\gamma,n}$ is given by:

$$\begin{aligned} G(f^*_{K,\gamma,n}) &= \mathsf{E}\|f^*_{K,\gamma,n} - f_{\nu,\mathcal{H}_K}\|^2 \\ &= \mathsf{E}\|f^*_{K,\gamma,n} - f^*_{K,\gamma} + f^*_{K,\gamma} - f_{\nu,\mathcal{H}_K}\|^2 \\ &= \mathsf{E}\|f^*_{K,\gamma,n} - f^*_{K,\gamma}\|^2 + \mathsf{E}\|f^*_{K,\gamma} - f_{\nu,\mathcal{H}_K}\|^2 \\ &+ 2\mathsf{E}\langle f^*_{K,\gamma,n} - f^*_{K,\gamma}, f^*_{K,\gamma} - f_{\nu,\mathcal{H}_K}\rangle, \end{aligned}$$

where the last term equals zero since $(f^*_{K,\gamma,n} - f^*_{K,\gamma})$ and $(f^*_{K,\gamma} - f_{\nu,\mathcal{H}_K})$ are orthogonal functions. Therefore $G(f^*_{K,\gamma,n})$ can be decomposed as

$$G(f^*_{K,\gamma,n}) = \mathsf{Var}(f^*_{K,\gamma,n}) + Bias^2(f^*_{K,\gamma,n}, f_{\nu,\mathcal{H}_K}), \tag{3.19}$$

where

$$\mathsf{Var}(f^*_{K,\gamma,n}) = \mathsf{E}\|f^*_{K,\gamma,n} - f^*_{K,\gamma}\|^2,$$

and

$$Bias^2(f^*_{K,\gamma,n}, f_{\nu,\mathcal{H}_K}) = \mathsf{E}\|f^*_{K,\gamma} - f_{\nu,\mathcal{H}_K}\|^2.$$

Eq. (3.17) assesses the quality of $f^*_{K,\gamma,n}$ in terms of its bias and variance. In practice the functions $f_{\nu,\mathcal{H}_K}$ and $f_{K,\gamma}$ are obviously unknown and therefore eq. (3.17) cannot be directly estimated. The key idea of the SIC is to replace $f_{\nu,\mathcal{H}_K}$ by an unbiased estimator f_u ($\mathsf{E}(f_u) = f_{\nu,\mathcal{H}_K}$) to roughly approximate $\mathsf{E}\|f^*_{K,\gamma,n} - f_{\nu,\mathcal{H}_K}\|^2$ by $\mathsf{E}\|f^*_{K,\gamma,n} - f_u\|^2$. Next we introduce a formal definition of the SIC adapted to our problem in eq. (3.1).

Definition 3.6. *The Subspace Infomation Criterion of the projected curve $f^*_{K,\gamma,n}$ is defined as*

$$SIC(f^*_{K,\gamma,n}) = \mathsf{E}\|f^*_{K,\gamma,n} - f_u\|^2, \tag{3.20}$$

where $f_u = f^*_{K,0,n}$.

The projection $f_u = f^*_{K,0,n}$ is an unbiased estimator of $f_{\nu,\mathcal{H}_K}$ (see Proposition 3.1 where, by hypothesis, $f_\nu = f_{\nu,\mathcal{H}_K} = L_K(f_\nu)$). Remark that while $f^*_{K,0,n}$ is estimated using the sample f_n and therefore it is a random variable, f_u in eq. (3.36) is considered to be a fixed function. In addition, both $f^*_{K,\gamma,n}$ and f_u belong to $\mathcal{H}_{K,n}$ and therefore the properties of the RKHSs can be used to estimate eq. (3.60) by eq. (3.20).

Denote by $\mathbf{K}$ the matrix whose components are defined by $(\mathbf{K})_{ij} = K(x_i, x_j)$. Let $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)^T$ and $\boldsymbol{\alpha}^0 = (\alpha^0_1, \ldots, \alpha^0_n)^T$ the kernel expansion representations (see eq. (3.3)) of $f^*_{K,\gamma,n}$ and $f^*_{K,0,n}$ respectively. In practice, $\boldsymbol{\alpha} = \mathbf{H}_\gamma \mathbf{y}$ where $\mathbf{H}_\gamma = (\gamma n \mathbf{I}_n + \mathbf{K})^{-1}$ and $\boldsymbol{\alpha}^0 = \mathbf{H}_0 \mathbf{y}$ (that is $\mathbf{H}_0 = \mathbf{K}^{-1}$). When $\mathbf{K}$ is not invertible then $\mathbf{H}_0 = \mathbf{K}^+$ is the

Moore-Penrose pseudoinverse of **K**. Then, it holds that

$$f^*_{K,\gamma,n} = \sum_{i=1}^{n} \alpha_i K(x, x_i) \ \ and \ \ f_u = \sum_{i=1}^{n} \alpha_i^0 K(x_i, x).$$

Operating from eq. (3.60) we can rewrite the $SIC(f^*_{K,\gamma,n})$ as

$$SIC(f^*_{K,\gamma,n}) = \mathsf{E}_\epsilon \|\boldsymbol{\alpha} - \mathsf{E}(\boldsymbol{\alpha})\|^2_{\mathbf{K}} + \|\mathsf{E}_\epsilon \boldsymbol{\alpha} - \boldsymbol{\alpha}^0\|^2_{\mathbf{K}}, \tag{3.21}$$

where $\|\mathbf{a}\|_{\mathbf{K}} = \mathbf{a}^T\mathbf{K}\mathbf{a}$. See (Sugiyama and Ogawa, 2001; Sugiyama and Muller, 2002) for further details. Notice that the first term estimates the variance of $f^*_{K,\gamma,n}$ while the second estimates its squared bias. In particular the variance term can be calculated as follows:

$$\widehat{Var}(f_{K,\gamma,n}) = \sigma^2 tr(\mathbf{K}\mathbf{H}_\gamma\mathbf{H}_\gamma^T), \tag{3.22}$$

where following (Wahba, 1990) an estimator of σ^2 is given by

$$\hat{\sigma}^2 = \frac{\|\mathbf{K}\boldsymbol{\alpha} - \mathbf{y}\|^2}{n - tr(\mathbf{K}\mathbf{H}_\gamma)}. \tag{3.23}$$

Regarding the bias term, it can be estimated by

$$\widehat{Bias}^2(f_{K,\gamma,n}) = \|\boldsymbol{\alpha} - \boldsymbol{\alpha}^0\|^2_{\mathbf{K}} - \hat{\sigma}^2 tr\left(\mathbf{K}(\mathbf{H}_\gamma - \mathbf{H}_0)(\mathbf{H}_\gamma - \mathbf{H}_0)^T\right). \tag{3.24}$$

See (Sugiyama and Ogawa, 2001) for details. Finally using eqs. (3.22) and (3.24) the SIC can be finally estimated by

$$\begin{aligned} SIC(f_{K,\gamma,n}) &= \|\boldsymbol{\alpha} - \boldsymbol{\alpha}^0\|^2_{\mathbf{K}} - \hat{\sigma}^2 tr\left(\mathbf{K}(\mathbf{H}_\gamma - \mathbf{H}_0)(\mathbf{H}_\gamma - \mathbf{H}_0)^T\right) \\ &+ \hat{\sigma}^2 tr(\mathbf{K}\mathbf{H}_\gamma\mathbf{H}_\gamma^T) \end{aligned}$$

where an estimator of σ^2 is given in eq. (3.23).

Remark 3.1. It can be proven that $SIC(f_{K,\gamma,n})$ defined in eq. (3.20) is an unbiased estimator of $G(f_{K,\gamma,n})$, that is

$$\mathsf{E}(SIC(f_{K,\gamma,n})) = G(f_{K,\gamma,n}). \tag{3.25}$$

In particular both, $\widehat{\mathsf{Var}}(f_{K,\gamma,n})$ and $\widehat{\mathsf{Bias}}^2(f_{K,\gamma,n})$ are unbiased estimators of $\mathsf{Var}(f_{K,\gamma,n})$ and $\mathsf{Bias}^2(f_{K,\gamma,n})$ respectively. That is,

$$\mathsf{E}\left(\widehat{\mathsf{Var}}(f_{K,\gamma,n})\right) = \mathsf{Var}(f_{K,\gamma,n}), \ \ \mathsf{E}\left(\widehat{\mathsf{Bias}}^2(f_{K,\gamma,n})\right) = \mathsf{Bias}^2(f_{K,\gamma,n}). \tag{3.26}$$

Remark 3.2. In some examples it may happen that the the value of the SIC can be negative if the estimated squared bias in eq. (3.24) is negative. In this cases, since the generalization error is non negative by definition, the following corrected SIC (cSIC) is proposed:

$$\begin{aligned} cSIC(f_{K,\gamma,n}) &= \left[\|\boldsymbol{\alpha} - \boldsymbol{\alpha}^0\|^2_{\mathbf{K}} - \hat{\sigma}^2 tr\left(\mathbf{K}(\mathbf{H}_\gamma - \mathbf{H}_0)(\mathbf{H}_\gamma - \mathbf{H}_0)^T\right)\right]_+ \\ &+ \hat{\sigma}^2 tr(\mathbf{K}\mathbf{H}_\gamma\mathbf{H}_\gamma^T). \end{aligned}$$

where $[t]_+ = max(t, 0)$.

3.4.3 Modified Subspace Information Criterion (MSIC) for Model Selection in Regularization with Nested Spaces

Next we slightly change the point of view of the previous section by relaxing the hypothesis that the target function f_ν must belongs to $\mathcal{H}_K$. In particular we will assume that f_ν belongs to a more general Reproducing Kernel Hilbert space $\mathcal{H}_{K'}$ such that $\mathcal{H}_K \subset \mathcal{H}_{K'}$. This hypothesis it is specially suitable when the optimal model $\{(K^*, \gamma^*)\}$ that minimizes eq. (3.17) has to be selected from a set of nested spaces. For instance, consider for any $x \in X$ the sequence of nested spaces given by $\mathcal{H}_1 = Span\{x\}$, $\mathcal{H}_2 = Span\{x, x^2\}$, $\mathcal{H}_3 = Span\{x, x^2, x^3\}$, $\mathcal{H}_4 = Span\{x, x^2, x^3, x^4\}$, etc., and suppose that f_ν is a polynomial of degree 3. While it is true that f_ν belongs to $\mathcal{H}_3$, $\mathcal{H}_4$, etc., and therefore the SIC estimation of the generalization error will work well in these spaces, such estimation will drastically fail for $\mathcal{H}_1$, $\mathcal{H}_2$ since the hypothesis that f_ν belongs to these spaces clearly does not hold. Similar situations appear in many statistical problems, for instance, when the basis of the eligibles RKHSs are sequences of Fourier, Splines or Wavelets functions or in RKHSs defined by polynomial kernels.

Let X be a compact space or manifold and $Y = \mathbb{R}$. Let ν a non degenerate Borel measure on $X \times Y$ and $f_n = \{(x_i, y_i)\}_{i=1}^n$ a sample curve drawn form ν. Let $K' : X \times X \longrightarrow \mathbb{R}$ be a continuous symmetric positive definite function. Then by Mercer's theorem

$$K'(x, y) = \sum_{j=1}^{\infty} \lambda_j \phi_j(x)\phi_j(y), \tag{3.27}$$

for $\{\phi_j\}$ and $\{\lambda_j\}$ the sets of eigenfunctions and eigenvectors of $L_{K'}$, the integral operator associated to K'. For any $d \in \mathbb{N}$ such as $d < \infty$ consider the "truncated" Mercer

Kernel defined by

$$K(x,y) = \sum_{j=1}^{d} \lambda_j \phi_j(x)\phi_j(y), \tag{3.28}$$

with associated integral operator L_K. Denote by $\mathcal{H}_K$ and $\mathcal{H}_{K'}$ the RKHSs associated to K and K' respectively. Then is is straightforward that

$$\mathcal{H}_K \subset \mathcal{H}_{K'}. \tag{3.29}$$

Let f_ν be the target function defined in eq. (3.1). In the the sequel we will assume that $f_\nu \in \mathcal{H}_{K'}$. Let $f_{\nu,\mathcal{H}_K}$ be the orthogonal projection of f_ν onto $\mathcal{H}_K$ being $f_{\nu,\mathcal{H}}^{\perp} = f_\nu - f_{\nu,\mathcal{H}_K}$ its orthogonal complement and let $f_{K,\gamma,n}^{*}$ be the projected curve obtained via eq. (3.2) using the sample f_n, some $\gamma > 0$ and K. Then our focus of interest in this chapter is the generalization error of $f_{K,\gamma,n}^{*}$ given by

$$\tilde{G}(f_{K,\gamma,n}^{*}) = \mathsf{E}\|f_{K,\gamma,n}^{*} - f_\nu\|^2. \tag{3.30}$$

Notice that $\tilde{G}$ differs form G in the fact that now the target function f_ν does not belong to $\mathcal{H}_K$. The consequence of this is, as we will study below, that the Subspace Infomation Criteron defined in eq. (3.20) is not an unbiased estimator of the generalization error $\tilde{G}$ and therefore it can fail if it is used for model selection. The main objective of this section is to propose an alternative to the SIC to solve this drawback.

Theorem 3.4. *Let $SIC(f_{K,\gamma,n}^{*})$ be the Subspace Information Criterion for $f_{K,\gamma,n}^{*}$ defined in eq. (3.20). Then under the same assumptions above, $SIC(f_{K,\gamma,n}^{*})$ is a biased estimator of the generalization error $\tilde{G}(f_{K,\gamma,n})$. In particular*

$$\mathsf{E}(SIC(f_{K,\gamma,n})) = \tilde{G}(f_{K,\gamma,n}) - \|f_{\nu,\mathcal{H}}^{\perp}\|^2. \tag{3.31}$$

The bias in eq. (3.31) is originated by the choice of f_u in eq. (3.20) since it is not an unbiased estimator of f_ν anymore. Notice that

$$\mathsf{E}(f_u) = \mathsf{E}(f_{K,0,n}^{*}) = f_\nu - f_{\nu,\mathcal{H}_K}^{\perp} = f_{\nu,\mathcal{H}} \tag{3.32}$$

being the term $f_{\nu,\mathcal{H}_1}^{\perp}$ the one that causes the bias in eq. (3.31). Notice that the closer $f_{\nu,\mathcal{H}_K}^{\perp}$ is to zero, the most unbiased is the SIC. In particular when $f_{\nu,\mathcal{H}_K}^{\perp} = 0$ we are in the case described in the previous section. In addition, when $\|f_{\nu,\mathcal{H}}^{\perp}\|^2 > \tilde{G}(f_{K,\gamma,n})$ then the SIC can be negative, which is not reasonable since we are estimating a generalization error that is always positive. In this cases the SIC correction in eq. (3.27) represents a naive solution to the problem that, however, does not guarantee a good estimation of

$\tilde{G}(f_{K,\gamma,n})$.

To address this problem we propose an alternative choice for f_u helpful to define an unbiased estimator of $\tilde{G}(f_{K,\gamma,n})$. The idea is to define f_u as the projection of f_ν (via the sample curve f_n) onto the RKHS ($\mathcal{H}$ or $\mathcal{H}'$ in this case) where the trace of the associated integral operator is maximum. As we detail below, this choice ensures that f_ν is projected onto the "biggest" available space minimizing this way the effect of the term $\|f^{\perp}_{\nu,\mathcal{H}}\|^2$ in eq. (3.31).

Definition 3.7. *Let K a Mercer kernel defined on a compact space X. The trace of the associated integral operator L_K is defined as*

$$trace(L_K) = \int_X K(x,x)dx. \tag{3.33}$$

The trace of an integral operator can be calculated as the sum of its eigenvalues: by Mercer' theorem if K is a Mercer kernel then $K(x,x) = \sum_{j=1} \lambda_j \phi_j(x)^2$ where $\{\lambda_j\}$ and $\{\phi_j\}$ are sets of eigenvalues and eigenfunctions of L_K. Integrating both sides of the equation

$$\int_X K(x,x)dx = \sum_{j=1} \lambda_j \int_X \phi_j(x)^2 dx, \tag{3.34}$$

and since $\{\phi_1, \phi_2, \dots\}$ is an orthonormal basis, that is $\int_X \phi_j(x)^2 dx = 1$ for all $j \geq 1$, then

$$\sum_j \lambda_j = \int_X K(x,x)dx = trace(L_K). \tag{3.35}$$

In practice, the eigenvalues $\{\lambda_j\}$ of the integral operator L_K are generally unknown but they can be easily estimated. Given a set $\mathbf{x} = \{x_1, \dots, x_n\}$ for $x_i \in X$, consider the matrix whose components are given by $(K\big|_{\mathbf{x}})_{ij} = K(x_i, x_j)$ and let l_j be the j-th eigenvalue of $K\big|_{\mathbf{x}}/n$. For any $j = 1, \dots, n$, then $l_j \longrightarrow \lambda_j$ when $n \longrightarrow \infty$ (see Proposition 4.1 in Chapter 4.1 for details). Then a direct way to estimate $trace(L_K)$ is given by $trace(K\big|_{\mathbf{x}}/n) = \sum_{j=1}^n l_j$.

Notice that given the nested structure of $\mathcal{H}_K$ and $\mathcal{H}_{K'}$ the space with the associated integral operator of maximum trace is always the biggest space. In this case it is straightforward that this space is $\mathcal{H}_{K'}$ since $\sum_{j=1}^d \lambda_j < \sum_{j=1}^\infty \lambda_j$ and therefore $trace(L_K) < trace(L_{K'})$.

Definition 3.8. *Let X be a compact space or manifold and $Y = \mathbb{R}$. Let ν a Borel measure on $X \times Y$ and $f_n = \{(x_i, y_i)\}_{i=1}^n$ a sample drawn form ν and $\mathbf{x} = (x_1, \dots, x_n)^T$. Let be $\mathcal{H}_K$*

and $\mathcal{H}_{K'}$ the two Reproducing kernel Hilbert spaces of associated kernels K and K' defined in eqs. (3.27) and (3.28). Consider the kernel matrices $K\big|_{\mathbf{x}}$ and $K'\big|_{\mathbf{x}}$ whose components are given by $(K\big|_{\mathbf{x}})_{ij} = K(x_i, x_j)$ and $(K'\big|_{\mathbf{x}})_{ij} = K'(x_i, x_j)$. Then we define the ***Modified Subspace Information Criterion (MSIC)*** *of the projection $f^*_{K,\gamma,n}$ given in eq. (3.1) as*

$$MSIC(f^*_{K,\gamma,n}) = \mathsf{E}\|f^*_{K,\gamma,n} - f_u\|^2, \tag{3.36}$$

where $f_u = f_{\tilde{K},0,n}$ for $\tilde{K} = \arg\max\{trace(K\big|_{\mathbf{x}}), trace(K'\big|_{\mathbf{x}})\}$.

Theorem 3.5. *The $MSIC(f_{K,\gamma,n})$ is an unbiased estimator of the generalization error $\tilde{G}(f_{K,\gamma,n})$, that is*

$$\mathsf{E}(MSIC(f_{K,\gamma,n})) = \tilde{G}(f_{K,\gamma,n}). \tag{3.37}$$

In practice, given the sample curve f_n, the optimal model (K^*, γ^*) has to be generally selected from a set of finite pairs $\{(K_i, \gamma_i)\}_{i=1}^m$, where each $K_i : X \times X \longrightarrow \mathbb{R}$ is Mercer kernel function and $\gamma_i > 0$. In this case we consider $f_u = f_{\tilde{K},0,n}$ for

$$\tilde{K} = \arg\max_i\{trace(K_i\big|_{\mathbf{x}})\}. \tag{3.38}$$

The MSIC can be therefore used to select the model between any set of pairs $\{(K_i, \gamma_i)\}_{i=1}^m$. See Table 3.1 for details. However, regarding Theorems 3.4 and 3.5 it is specially useful when the kernels $K_1, \ldots, K_m$ induce a set of nested spaces $\mathcal{H}_1 \subset \mathcal{H}_2 \subset \ldots \mathcal{H}_m$, case in which the SIC can fail. We illustrate this in the following example.

Example 3.3. The idea of this experiment in to show how the SIC, in contrast to the MSIC, may fail as model selection criterion when a set of nested spaces are the candidates where project functional data. To this aim we define a set of spaces

$$\mathcal{H}_{K_t} = Span\left\{\frac{\sqrt{2}}{j\pi} sin(j\pi x)\right\}_{j=1}^t,$$

for $t = 1 \ldots, 10$ where

$$K_t(x_i, x_j) = \sum_{j=1}^t \frac{2}{j^2\pi^2} sin(j\pi x_i) sin(j\pi x_j).$$

In this experiment we generate 500 equally spaced data points $(x_i, f(x_i))$ in the interval $[0, 1]$ using:

INPUT	
$f_n = \{(x_i, y_1\}_{i=1}^n$	Sample curve where $\mathbf{y} = (x_1, \ldots, x_n)^T$ and $\mathbf{x} = (y_1, \ldots, y_n)^T$.
$\{(K_i, \gamma_i)\}_{i=1}^m$	Candidate models.
OUTPUT	
(K^*, γ^*)	Optimal model.
MSIC	Estimated generalization error of the optimal model.
STEP 1	**Unbiased estimator of f_u** Calculate $\tilde{K} = \arg\max_i\{trace(K_i\big|_{\mathbf{x}})\}$. Estimate $f_{\tilde{K},0,n}$ via eq. (3.2) and obtain $\tilde{\mathbf{H}}_0 = (\tilde{K}\big|_{\mathbf{x}})^+$ and $\tilde{\boldsymbol{\alpha}}^0 = \tilde{\mathbf{H}}_0\mathbf{y}$.
STEP 2	**For $i = 1, \ldots, m$, estimate the MSIC of $f_{K_i,\gamma_i,n}$** Use eq. (3.25) where $\mathbf{H}_0 = \tilde{\mathbf{H}}_0$ and $\boldsymbol{\alpha}^0 = \tilde{\boldsymbol{\alpha}}^0$.
STEP 3	**Select the optimal model** $(K^*, \gamma^*) = \arg\, min_i\, MSIC(f_{K_i,\gamma_i,n})$.

Table 3.1: Algorithm to select the optimal model from a set of candidates using the Modified Subspace Information Criterion.

$$
\begin{aligned}
f(x) \quad &= \quad \frac{\sqrt{2}}{\pi} sin(\pi x) + 1.5\frac{\sqrt{2}}{2\pi} sin(2\pi x) + 2\frac{\sqrt{2}}{3\pi} sin(3\pi x) \\
&+ \quad 2.5\frac{\sqrt{2}}{4\pi} sin(4\pi x) + 3\frac{\sqrt{2}}{5\pi} sin(5\pi x),
\end{aligned} \tag{3.39}
$$

and we assume that

$$y_i = f(x_i) + \epsilon_i, \tag{3.40}$$

for $\epsilon_i \sim N(0, 0.075)$. Obviously the optimal space where project the data is $\mathcal{H}_{K_5}$.

The problem is stated as seeking the projection $f^*_{K_t,\gamma,n}$ with the minimum generalization error among the pairs $(K_1, \gamma), \ldots, (K_{10}, \gamma)$ for $\gamma = 10^{-4}$. To this aim we estimate both, SIC and MSIC for the 10 pairs (K_i, γ_i) and we compare the obtained results. A decomposition of the estimated squared bias and variance of the two criteria is shown in Table 3.2. In addition we include in Figure 3.3 a plot of the SIC and MSIC results (in logarithms) in the 10 spaces and a plot of the estimated projections in $\mathcal{H}_{K_1}, \ldots, \mathcal{H}_{K_5}$.

The first and most important conclusion of this experiment is that the MSIC selects the right model ($\mathcal{H}_{K_5}$), in contrast to the SIC that selects $\mathcal{H}_{K_1}$. The reason of this behavior

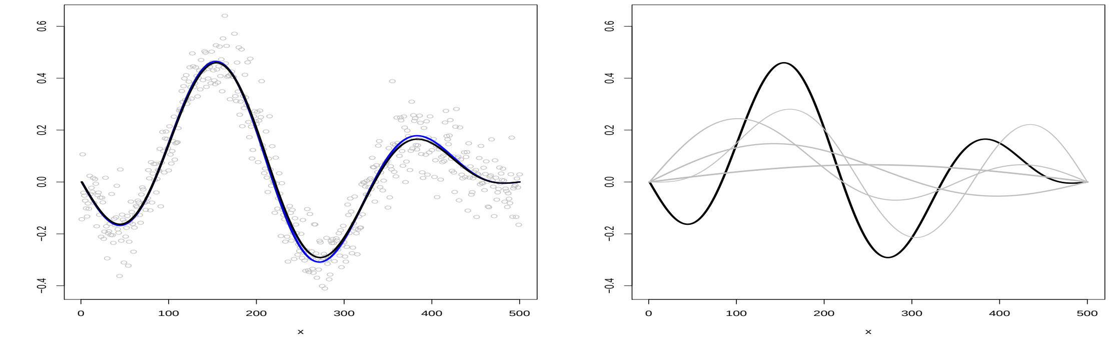


Figure 3.4: Left: Simulated data, real model (blue) and estimated projection (black). Right: sequence of projected curves onto $H_{K_1}, \dots H_{K_5}$. Thinner curves correspond to lower dimensional spaces. $f^*_{K_5,\gamma,n}$ is pointed out in black.

Table 3.2: SIC and MSIC values for the 10 curves projections. Their bias-variance decomposition is also included.

	dim	1	2	3	4	5	6	7	8	9	10
SIC	bias	0.0000	0.0000	0.0001	0.0005	0.0030	0.0030	0.0030	0.0030	0.0030	0.0029
	var	0.0019	0.0051	0.0105	0.0140	0.0062	0.0101	0.0152	0.0218	0.0298	0.0395
	SIC	0.0019	0.0051	0.0106	0.0145	0.0093	0.0131	0.0183	0.0248	0.0328	0.0424
MSIC	bias	7.1277	6.9355	6.3137	4.3097	0.0129	0.0127	0.0124	0.0101	0.0061	0.0029
	var	0.0019	0.0051	0.0105	0.0140	0.0062	0.0101	0.0152	0.0218	0.0298	0.0395
	MSIC	7.1296	6.9405	6.3242	4.3237	**0.0191**	0.0228	0.0277	0.0319	0.0360	0.0424

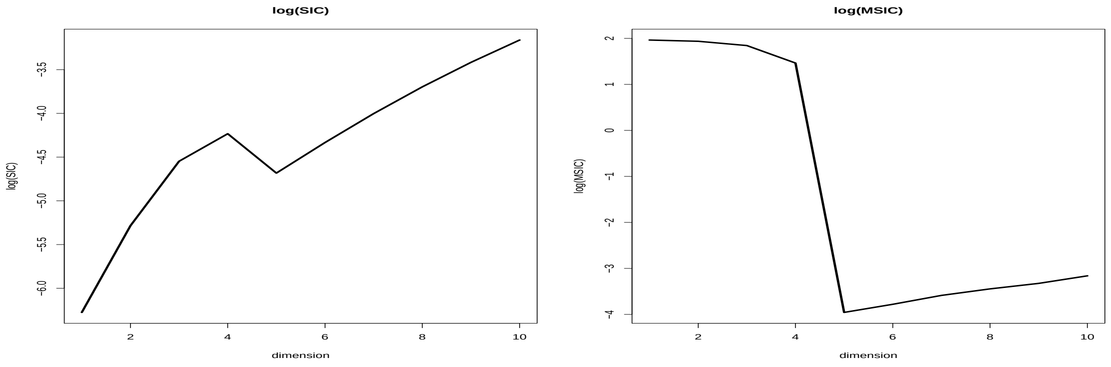


Figure 3.5: Logarithmic transformation of SIC and MSIC values for the 10 projections.

can be clearly understood in terms of the estimated bias and variance of each criterion. First of all, notice that the estimation of the variance, that is not affected by the definition of f_u, is equal in both cases. However, strong differences are found regarding the estimated squared bias. In the MSIC the bias decreases with the dimension specially for values of t form 1 to 5. This behavior is reasonable by the definition of f. On the other hand, the bias estimated by the SIC is close to zero for values of t form 1 to 4. This effect appears due to the choice of f_u which, in the SIC, is not an unbiased estimator of f in $\mathcal{H}_1, \ldots, \mathcal{H}_5$. The consequence of this is that the real generalization error is underestimated (see Theorem 3.4) and hence the SIC fails selecting $\mathcal{H}_1$ as optimal space.

• • •

3.5 Truncation Error analysis

Given a Mercer kernel K defined on a compact space or manifold X and for any $\mathbf{x}, \mathbf{y} \in X$, the kernel expansion $K(\mathbf{x}, \mathbf{y}) = \sum_{j=1} \lambda_j \phi_j(\mathbf{x}) \phi_j(\mathbf{y})$ for contains $d = rank(L_K) \leq \infty$ non null terms. In practice, we always work with a finite sample of size n and the approximation errors that appear when we calculate the projection $f^*_{K,\gamma,n}$ via eq. (3.2) (using f_n) must be taken into account.

Let $K\big|_{\mathbf{x}}$ the kernel matrix whose components are given by $(K\big|_{\mathbf{x}})_{ij} = K(x_i, x_j)$ and consider $n = rank(K_{\mathbf{x}})$ an estimator of $rank(L_K)$. Define

$$K^{[n]} = \sum_{j=1}^{n} \lambda_j \phi_j(x) \phi_j(y) \tag{3.41}$$

the truncated kernel of n elements. If $rank(K\big|_{\mathbf{x}}) = rank(L_K)$ then $K^{[n]} = K$ and there is no loss in using $K^{[n]}$ (all the eigenfunctions of K can be approximated). If $rank(K\big|_{\mathbf{x}}) < rank(L_K)$ (what can only happen when $rank(K\big|_{\mathbf{x}}) = n$), the number of eigenfunctions of K is larger than the number of data points and $K^{[n]}$ takes into account only the first n eigenvalues of K. Next we analyze the truncation and approximation error in this case.

Let

$$f^*_{K,\gamma} = \sum_{j=1}^{\infty} \lambda^*_j \phi_j, \quad and \quad f^{*[n]}_{K,\gamma} = \sum_{j=1}^{n} \lambda^*_j \phi_n, \tag{3.42}$$

be the projections of f_ν (defined in eq. (3.1)) onto $\mathcal{H}_K$ and $\mathcal{H}_{K^{[n]}}$ respectively. Then we are interested in sample bounds for

$$Error(f^{*[n]}_{K,\gamma}) = \|f^*_{K,\gamma} - f^{*[n]}_{K,\gamma}\|^2. \tag{3.43}$$

First, we prove that the norm of $\|f^*_{K,\gamma} - f^{*[n]}_{K,\gamma}\|^2$ cannot be larger than the $\|f^*_{K,\gamma}\|^2$. By Parseval's identity (Conway, 1990) we have that

$$\|\sum_{j=1}^{\infty} \lambda^*_j \phi_j(x)\|^2 = \sum_{j=1}^{\infty} (\lambda^*_j)^2 \|\phi_j(x)\|^2 = \sum_{j=1}^{\infty} (\lambda^*_j)^2, \tag{3.44}$$

and therefore

$$\|f^*_{K,\gamma} - f^{*[n]}_{K,\gamma}\|^2 = \sum_{j=n+1} (\lambda^*_j)^2 \leq \sum_{j=1} (\lambda^*_j)^2 = \|f^*_{K,\gamma}\|^2 = M^2. \tag{3.45}$$

for $M = \|f^*_{K,\gamma}\|$. Notice that $M > 0$. Next, by the following lemma we show that $f_{K,\gamma}$ and also $f^*_{K,\gamma} - f^{*[n]}_{K,\gamma}$ are uniformly bounded functions.

Lemma 3.1. *Let K be a kernel function defined on a compact space X. Then*

$$|f^*_{K,\gamma}| \leq C_K M,$$

$$|f^*_{K,\gamma} - f^{*[n]}_{K,\gamma}| \leq C_K M,$$

for $M > 0$ and $C_K = \sup_{x,y} |K(x,y)|$.

Consider a random sample $x_1, \dots, x_m$ drawn form ν_x the marginal probability measure of ν on X. For simplicity in notation in the sequel we will denote as E and and Var the expectation and variance over this measure. Define the vectors $\mathbf{x} = (x_1, \dots, x_m)^T$, $\mathbf{f}^*_{K,\gamma,\mathbf{x}} = (f^*_{K,\gamma}(x_1), \dots, f^*_{K,\gamma}(x_m))^T$ and $\mathbf{f}^{*[n]}_{K,\gamma,\mathbf{z}} = (f^{*[n]}_{K,\gamma}(x_1), \dots, f^{*[n]}_{K,\gamma}(x_m))^T$. Then by the Strong Law of Large Numbers:

$$\begin{aligned} \frac{1}{m}\|\mathbf{f}^*_{K,\gamma,\mathbf{x}} - \mathbf{f}^{*[n]}_{K,\gamma,\mathbf{x}}\|^2 &= \frac{1}{m}\sum_{i=1}^{n}(f^*_{K,\gamma}(x_i) - f^{*[n]}_{K,\gamma}(x_i))^2 \\ &\underset{m\to\infty}{\longrightarrow} \mathsf{E}\left((f^*_{K,\gamma}(x) - f^{*[n]}_{K,\gamma}(x))^2\right) = \|f^*_{K,\gamma} - f^{*[n]}_{K,\gamma}\|^2, \end{aligned} \tag{3.46}$$

almost surely. In order to obtain finite sample size bounds for $Error(f^{*[n]}_{K,\gamma})$ we first deduce some probabilistic properties of it. Regarding the averaged error expectation, by eq. (3.46) we have that

$$\begin{aligned} \mathsf{E}\left(\frac{1}{m}\|\mathbf{f}^*_{K,\gamma,\mathbf{z}} - \mathbf{f}^{*[n]}_{K,\gamma,\mathbf{z}}\|^2\right) &= \|f^*_{K,\gamma} - f^{*[n]}_{K,\gamma}\|^2 \\ &= \sum_{j=n+1} (\lambda^*_j)^2. \end{aligned} \tag{3.47}$$

On the other hand, by the definition of the variance and applying the Hölder inequality:

$$\begin{aligned}
\mathsf{Var}\left((f^*_{K,\gamma} - f^{*[n]}_{K,\gamma})^2\right) &\leq \mathsf{E}\left((f^*_{K,\gamma} - f^{*[n]}_{K,\gamma})^4\right) - \mathsf{E}\left((f^*_{K,\gamma} - f^{*[n]}_{K,\gamma})^2\right)^2 \\
&\leq \mathsf{E}\left((f^*_{K,\gamma} - f^{*[n]}_{K,\gamma})^4\right) \\
&\leq \|(f^*_{K,\gamma} - f^{*[n]}_{K,\gamma})^2\|_\infty \|(f^*_{K,\gamma} - f^{*[n]}_{K,\gamma})^2\|_1.
\end{aligned}$$

By Lemma 3.1 we know that $(f^*_{K,\gamma} - f^{*[n]}_{K,\gamma})^2 \leq (f^*_{K,\gamma})^2 \leq C_K^2 M^2$ and therefore $\|(f^*_{K,\gamma} - f^{*[n]}_{K,\gamma})^2 \leq \|f^*_{K,\gamma}\|_\infty \leq C_K^2 M^2$. In addition $\|(f^*_{K,\gamma} - f^{*[n]}_{K,\gamma})^2\|_1 = \|f^*_{K,\gamma} - f^{*[n]}_{K,\gamma}\| \leq M^2$ and therefore

$$\mathsf{Var}\left((f^*_{K,\gamma} - f^{*[n]}_{K,\gamma})^2\right) \leq C_K^2 M^4. \tag{3.48}$$

Finally we use the results in eqs. (3.47) and (3.48) to give finite sample size bounds for $Error(f^{*[n]}_{K,\gamma})$ in next theorem.

Theorem 3.6. *Unnder the conditions above for any* $0 < \delta < 1$, *with probability larger that* $1 - \delta$

$$\frac{1}{m}\|\boldsymbol{f}^*_{K,\gamma,z} - \boldsymbol{f}^{*[n]}_{K,\gamma,z}\|^2 < \|f^*_{K,\gamma} - f^{*[n]}_{K,\gamma}\|^2 + C_K M^2 \sqrt{\frac{1}{m\delta}}, \tag{3.49}$$

for any random sample $x_1, \ldots, x_m$ *of size* m *drawn from* ν *and for* $C_K = sup_{x,y \in X}|K(x,y)|$ *and* $M > 0$.

3.6 Conclusions and final remarks

In this chapter we have proposed a methodology to represent functional data via their projections onto Reproducing Kernel spaces with the aid of Regularization theory. In addition we have proposed a model selection criterion to estimate the generalization error of such representations.

Regarding the projection method, two representation systems for functional data naturally appear: the RKHS and the Kernel expansion representation. In Theorems 3.2 and 3.3 we have studied their stability properties concluding that the RKHS Representation is ϵ-stable in the input variables, and therefore adequate to represent functional data, in contrast to the Kernel Expansion. In addition the RKHS Representation (Theorem 3.1) allows to evaluate the dimension of the curves (Example 3.1) and it enables to reinterpret the regularization process like a curve projection mechanism onto $\mathcal{H}_K$ (Propositions 3.1 and 3.2).

Finally, we have proposed the MSIC as alternative to the SIC for models selection. In contrast to the SIC, this new criterion gives an unbiased estimator of the generalization error of the projected curves (Theorems 3.4 and 3.5) and it is proven to work better that the SIC when a set of nested spaces is available (Example 3.3).

The last contribution of this chapter is the generalization of the classical FDA representation techniques. Any orthogonal basis $B = \{\varphi_1, \ldots, \varphi_d\}$ of continuos functions on X, for instance B-splines, fourier basis, P-splines, defines a kernel (and therefore an RKHS) given by

$$K(x, y) = \sum_{j=1}^{d} \varphi_j(x)\varphi_j(y). \tag{3.50}$$

See (Rakotomamonjy and Canu, 2005) for details. However, in this chapter we have shown how to select generalized covariance functions appropriate for functional data and to work directly with their eigenfunctions (basis of the RKHS). This makes accessible a larger class of basis of functions where represent the functional data that otherwise are ingnored, constituting this methodology a generalization of the classical FDA formalism.

3.7 APPENDIX: Proofs

Proof Proposition 3.1. First, operating from eq. (3.3), we have that

$$\begin{aligned} f^*_{K,0,n} &= \sum_{i=1}^{n} \alpha_i K(x_i, x) = \sum_{i=1}^{n} \alpha_i \left(\sum_{j=1} \lambda_j \phi_j(x_i) \phi_j(x) \right) \\ &= \sum_{j=1} \lambda_j \left(\sum_{i=1}^{n} \alpha_i \phi_j(x_i) \right) \phi_j(x) \\ &= \sum_{j=1} \lambda_j (\boldsymbol{\alpha}^T \boldsymbol{\phi}_{j,\mathbf{x}}) \phi_j(x), \end{aligned} \tag{3.51}$$

To check the uniform convergence of $f^*_{K,0,n}$ to $L_K(f_\nu)$ we have to prove that for every $\epsilon > 0$ there exists a $N \in \mathbb{N}$ such that for all $x \in X$ and all $n \leq N$, then $|f^*_{K,0,n}(x) - L_K(f_\nu)(x)| < \epsilon$. To this aim, consider the sequence

$$a_n = \sup |f^*_{K,0,n}(x) - L_K(f_\nu)(x)|,$$

where the supremum is taken over all $x \in X$. Then $f^*_{K,0,n}$ converges to $L_K(f_\nu)$ uniformly if and only if a_n goes to 0 when $n \longrightarrow \infty$.

Let $f_{\nu,\mathcal{H}_K} = L_K(f_\nu)$ be the orthogonal projection of f_ν onto $\mathcal{H}_K$. By the spectral theorem

$$f_{\nu,\mathcal{H}_K} = L_K(f_\nu) = \sum_j \lambda_j \langle f_\nu, \phi_j \rangle \phi_j,$$

When $n \longrightarrow \infty$ the problem in eq. (3.2) tends to

$$f^*_{K,\gamma} = \Pi_{K,\gamma,\infty}(f) = \arg \min_{f \in \mathcal{H}_K} \int_{X \times Y} (y - f(x))^2 d_\nu(x, y) + \gamma \|f\|^2_K \,, \tag{3.52}$$

which unique minimizer (Cucker and Smale, 2001) is given by

$$f^*_{K,\gamma} = (Id + \gamma L_K)^{-1} f_{\nu,\mathcal{H}_K}. \tag{3.53}$$

Since $\gamma = 0$, is direct to see (from eq. (3.53)) that $f^*_{K,0} = f_{\nu,\mathcal{H}_K}$. Then when $n \longrightarrow \infty$, $f^*_{K,0}$ the unique solution to eq. (3.2) tends to $f_{\nu,\mathcal{H}_K}$ the unique solution of eq. (3.52) and therefore $a_n \longrightarrow 0$. Then

$$\Pi_{K,0,n}(f) = \sum_j \lambda_j (\boldsymbol{\alpha}^T \boldsymbol{\phi}_{j,\mathbf{x}}) \phi_j \underset{n \to \infty}{\longrightarrow} L_K(f) = \sum_j \lambda_j \langle f, \phi_j \rangle \phi_j$$

uniformly in X, what concludes the proof. □

Proof Proposition 3.2. By Proposition 3.1 we now that, for $\boldsymbol{\alpha}$ the solution to eq. (3.4), then $f^*_{K,\gamma,n} = \sum_j \lambda_j(\boldsymbol{\alpha}^T\boldsymbol{\phi}_{j,\mathbf{x}})\phi_j$. In addition the unique solution for problem in eq. (3.2) when $n \longrightarrow \infty$ is given by $f^*_{K,\gamma} = (Id+\gamma L_K)^{-1}f_{\nu,\mathcal{H}_K}$ for $f_{\nu,\mathcal{H}_K} = L_K(f_\nu)$ the orthogonal projection of f_ν onto $\mathcal{H}_K$.

Since $f^*_{K,\gamma} \in H_K$ the we can write $f^*_{K,\gamma} = \sum_j \beta'_j\phi_j$ for appropriate $\beta' \in \mathbb{R}$ and for $\phi_1, \phi_2 \ldots$ the eigenfunctions of K. Without loss of generally we can rewrite $f^*_{K,\gamma}$ as

$$f^*_{K,\gamma} = \sum_j \lambda_j\beta_j\langle f_\nu, \phi_j\rangle\phi_j$$

since $\langle\cdot,\cdot\rangle$ is well defined and λ_j, the eigenvalues of L_K, are all real. Denote $\beta_j = \beta'{}_j(\lambda_j\langle f, \phi_j\rangle)^{-1}$ and define $\langle f, \phi_j\rangle'$ such that $\langle f, \phi_j\rangle' = \beta_j\langle f, \phi_j\rangle$. Then we have that $f^*_{K,\gamma} = \sum_j \lambda_j\langle f, \phi_j\rangle'\phi_j$.

To end the proof, we only have to check the uniform convergence in X of $f^*_{K,\gamma,n}$ to $f^*_{K,\gamma}$. Following the same reasoning that in Proof 3.7 we define the sequence

$$b_n = \sup |f^*_{K,\gamma,n}(x) - f^*_{K,\gamma}(x)|,$$

where the supremum is taken over all $x \in X$. Then b_n goes to 0 when $n \longrightarrow \infty$ by the same reason that a_n goes to 0 in Proof 3.7 and therefore

$$f^*_{K,\gamma,n} = \sum_j \lambda_j(\boldsymbol{\alpha}^T\phi_{j,\mathbf{x}})\phi_j \underset{n\to\infty}{\longrightarrow} f^*_{K,\gamma} = \sum_j \lambda_j\langle f, \phi_j\rangle'\phi_j,$$

uniformly in X what concludes the proof. □

Proof Theorem 3.1. Operating from eq. (3.51)

$$f^*_{K,\gamma,n}(\mathbf{x}) = \sum_{j=1} \lambda_j(\boldsymbol{\alpha}^T\phi_{j,\mathbf{x}})\boldsymbol{\phi}_j(\mathbf{x}) = \sum_{j=1} \lambda^*_j\phi_j(\mathbf{x}),$$

for $\lambda^*_j = \lambda_j(\boldsymbol{\alpha}^T\boldsymbol{\phi}_{j,\mathbf{x}})$.

Following (Smale and Zhou, 2007) the eigenvalues and eigenvectors of $\mathbf{K}\big|_{\mathbf{x}}/n$ converge, to the eigenvalues and eigenfunctions of L_K. Following Theorem 4.1 in Chapter 4 each $\phi_j(x_i)$ and λ_j can be estimated by $\sqrt{n}\mathbf{v}_{ji}$ and $\hat{\lambda}_j = l_j/n$ respectively. Therefore replacing in $\lambda^*_j = \lambda_j(\boldsymbol{\alpha}^T\boldsymbol{\phi}_{j,\mathbf{x}})$ each λ_j and $\phi_j(x_i)$ by its estimators

$$\hat{\lambda}_j^* = \hat{\lambda}_j(\boldsymbol{\alpha}^T \hat{\phi}_{j,\mathbf{x}}) = \frac{l_j}{n}(\boldsymbol{\alpha}^T \sqrt{n}\mathbf{v}_j) = \frac{l_j}{\sqrt{n}}\boldsymbol{\alpha}^T \mathbf{v}_j \tag{3.54}$$

what concludes the proof. □

Proof Theorem 3.2. Consider a sample curve f_n and an ϵ-perturbed curve $f_n^\epsilon \equiv \{(x_i, y_i^\epsilon) \in X \times Y\}_{i=1}^n$. Then $f_{K,\gamma,n}^*(\mathbf{x}) \simeq f_{K,\gamma,n}^{*\epsilon}(\mathbf{x})$ and given that the ϕ_j are a basis for $\mathcal{H}_K$, it must happen that $\lambda_j^* \simeq \lambda_j^{*\epsilon}$ and therefore $\hat{\lambda}_j^* \simeq \hat{\lambda}_j^{*\epsilon}$. Hence

$$\frac{|\hat{\lambda}_j^* - \hat{\lambda}_j^{*\epsilon}|}{|\hat{\lambda}_j|} \leq \epsilon, \tag{3.55}$$

for $j = 1, \ldots, d$ and the representation system is ϵ-stable. Notice that the truth of this statement relies in the fact that the eigenvalues and eigenvectors of $\mathbf{K}\big|_{\mathbf{x}}$ converge, respectively, to the eigenvalues and eigenfunctions of L_K and therefore $\hat{\lambda}_j^* \longrightarrow \lambda_j^*$. See Theorem 3.7 for details. □

Proof Theorem 3.3. By Theorem 3.7 we know that $f_{K,\gamma,n}^*(x) = \sum_{j=1} \lambda_j(\boldsymbol{\alpha}^T \boldsymbol{\phi}_{j,\mathbf{x}})\phi_j(x)$. In addition, since $\{\phi_j\}$ is a basis for $\mathcal{H}_K$, then $\boldsymbol{\alpha}^T \boldsymbol{\phi}_{j,\mathbf{x}} \longrightarrow \langle f_\nu, \phi_j \rangle$ (see Theorem 3.7). Therefore, for any set $\boldsymbol{\alpha}' = (\alpha_1', \ldots, \alpha_n')^T$ such that $(\boldsymbol{\alpha}')^T \boldsymbol{\phi}_{j,\mathbf{x}} \longrightarrow \langle f_\nu, \phi_j \rangle$ we will have that $\sum_{i=1}^n \alpha_i^{*\prime} K(x_i, x) = f_{K,\gamma,n}^*(x)$. Now, given the sample curve $f_n \equiv \{(x_i, y_i) \in X \times Y\}_{i=1}^n$, consider an ϵ-perturbed curve $f_n^\epsilon \equiv \{(x_i, y_i^\epsilon) \in X \times Y\}_{i=1}^n$, such that

$$\frac{|y_i - y_i^\epsilon|}{|y_i|} \leq \epsilon, \tag{3.56}$$

Denote by $(\boldsymbol{\alpha}^\epsilon)$ the representation corresponding to f_n^ϵ. Given that $f_{K,\gamma,n}^{*\epsilon}(x) \simeq f_{K,\gamma,n}^*(x)$ (because of the continuity of f_ν), it will happen that $(\boldsymbol{\alpha}^\epsilon)^T \boldsymbol{\phi}_{j,\mathbf{x}} \simeq \boldsymbol{\alpha}^T, \boldsymbol{\phi}_{j,\mathbf{x}^\epsilon}$ and, nevertheless, by the previous reasoning, $\boldsymbol{\alpha}^\epsilon$ and $\boldsymbol{\alpha}$ can be quite different. Therefore is not guaranteed that

$$\frac{|\alpha_i - \alpha_i^\epsilon|}{|\alpha_i|} \leq \epsilon. \tag{3.57}$$

for all $i = 1, \ldots, n$ and therefore the representation is not ϵ-stable. □

Proof Theorem 3.4. The Bias-Variance decomposition of $\tilde{G}(f^*_{K,\gamma,n})$ is

$$\begin{aligned}\tilde{G}(f^*_{K,\gamma,n}) &= \mathsf{E}\|f^*_{K,\gamma,n} - f_\nu\|^2 \\ &= \mathsf{E}\|f^*_{K,\gamma,n} - f^*_{K,\gamma} + f^*_{K,\gamma} - f_\nu\|^2 \\ &= \mathsf{E}\|f^*_{K,\gamma,n} - f^*_{K,\gamma}\|^2 + \mathsf{E}\|f^*_{K,\gamma} - f_{\nu,\mathcal{H}_K}\|^2 \\ &+ 2\mathsf{E}\langle f^*_{K,\gamma,n} - f^*_{K,\gamma}, f^*_{K,\gamma} - f_\nu\rangle,\end{aligned}$$

where the last term equals zero since $(f^*_{K,\gamma,n} - f^*_{K,\gamma})$ and $(f^*_{K,\gamma} - f_\nu)$ are orthogonal functions. Then

$$\tilde{G}(f^*_{K,\gamma,n}) = \mathsf{Var}(f^*_{K,\gamma,n}) + Bias^2(f^*_{K,\gamma,n}, f_\nu), \tag{3.58}$$

where

$$\mathsf{Var}(f^*_{K,\gamma,n}) = \mathsf{E}\|f^*_{K,\gamma,n} - f^*_{K,\gamma}\|^2,$$

and

$$Bias^2(f^*_{K,\gamma,n}, f_\nu) = \mathsf{E}\|f^*_{K,\gamma} - f_\nu\|^2 = \mathsf{E}\|f^*_{K,\gamma} - f_{\nu,\mathcal{H}}\|^2 + \|f^\perp_{\nu,\mathcal{H}}\|^2.$$

since $f^*_{K,\gamma} - f_{\nu,\mathcal{H}}$ and $f^\perp_{\nu,\mathcal{H}}$ are also orthogonal functions.

Following eq. (3.60) then $\tilde{G}(f^*_{K,\gamma,n}) = G(f^*_{K,\gamma,n}) + \|f^\perp_{\nu,\mathcal{H}}\|^2$ and since $\mathsf{E}(G(f^*_{K,\gamma,n})) = G(f^*_{K,\gamma,n})$ then

$$\mathsf{E}(SIC(f_{K,\gamma,n})) = G(f_{K,\gamma,n}) = \tilde{G}(f^*_{K,\gamma,n}) - \|f^\perp_{\nu,\mathcal{H}}\|^2,$$

what concludes the proof. □

Proof. By definition of K and K', then $K' = K + K^\perp$ where $K^\perp(x,y) = \sum_{j=d+1}^\infty \lambda_j\phi_j(x)\phi_j(y)$ for any $x, y \in X$. Then $K'\big|_{\mathbf{x}} = K\big|_{\mathbf{x}} + K^\perp\big|_{\mathbf{x}}$.

By the properties of the trace we have that

$$trace(K'\big|_{\mathbf{x}}) = trace(K\big|_{\mathbf{x}}) + trace(K^\perp\big|_{\mathbf{x}}) \tag{3.59}$$

and therefore $trace(K\big|_{\mathbf{x}}) < trace(K'\big|_{\mathbf{x}})$ and $\tilde{K} = K'$.

The $MSIC(f^*_{K,\gamma,n})$ can be decomposed as

$$MSIC(f^*_{K,\gamma,n}) = \hat{\mathsf{Var}}(f^*_{K,\gamma,n}) + \hat{\mathsf{Bias}}^2(f^*_{K,\gamma,n}, f_u), \tag{3.60}$$

where $\hat{\mathsf{Var}}(f^*_{K,\gamma,n}) = \mathsf{E}\|f^*_{K,\gamma,n} - f^*_{K,\gamma}\|^2$ and $\hat{\mathsf{Bias}}^2(f^*_{K,\gamma,n}, f_\nu) = \mathsf{E}\|f^*_{K,\gamma} - f_u\|^2$ for $f_u = f^*_{K',0,n}$. To prove the theorem we check that the estimated squated bias an variance are unbiased estimators of $\mathsf{Bias}^2(f^*_{K,\gamma,n}, f_\nu)$ and $\mathsf{Var}(f^*_{K,\gamma,n})$.

The estimated variance, it does not depends of f_u and, $\widehat{\mathsf{Var}}(f^*_{K,\gamma,n}) = \mathsf{Var}(f^*_{K,\gamma,n})$ by the properties of the SIC. Regarding the bias term, we first decompose $f^*_{K',0,n}$ as a sum. Then

$$f^*_{K',0,n} = f^*_{K,0,n} + f^{*\perp}_{K,0,n},$$

where $f^{*\perp}_{K,0,n}$ is the orthogonal complement to $f^*_{K,0,n}$ and being $\mathsf{E}(f^{*\perp}_{K,0,n}) = f^{\perp}_{\nu,\mathcal{H}_K}$. Then,

$$\begin{aligned} \mathsf{E}\|f^*_{K,\gamma} - f_u\|^2 &= \mathsf{E}\|f^*_{K,\gamma} - f^*_{K,0,n} - f^{*\perp}_{K,0,n}\|^2 \qquad (3.61) \\ &= \mathsf{E}\|f^*_{K,\gamma} - f^*_{K,0,n}\|^2 + \mathsf{E}\|f^{*\perp}_{K,0,n}\|^2, \end{aligned}$$

since $f^*_{K,\gamma} - f^*_{K,0,n}$ and $f^{*\perp}_{K,0,n}$ are orthogonal functions. Therefore

$$\begin{aligned} \mathsf{E}(\widehat{\mathsf{Bias}}^2(f^*_{K,\gamma,n}, f_\nu)) &= \mathsf{E}(\mathsf{E}\|f^*_{K,\gamma} - f^*_{K,0,n}\|^2) + \mathsf{E}(\mathsf{E}\|f^{*\perp}_{K,0,n}\|^2) \qquad (3.62) \\ &= \mathsf{E}(\mathsf{Bias}^2(f^*_{K,\gamma,n}, f_{\nu,\mathcal{H}_K})) + \|f^{*\perp}_{K,0,n}\|^2 \\ &= \mathsf{Bias}^2(f^*_{K,\gamma,n}, f_\nu), \end{aligned}$$

what concludes the proof. □

Proof Lemma 3.1. First of all, let

$$C_K = sup_{x,y\in X}|K(x,y)|. \qquad (3.63)$$

Since K is continuous and X is compact, C_K always exists and $C_K < \infty$. By fixing one of the arguments of the kernel, we have that

$$|f^*_{K,\gamma}| = |\langle K(x,\cdot), f^*_{K,\gamma}\rangle| \leq \|K(x,\cdot)\|\|f^*_{K,\gamma}\| \leq C_K\|f^*_{K,\gamma,n}\| = C_K M, \qquad (3.64)$$

applying that $\|K(x,\cdot)\| = \left(\int_X K^2(x,y)dy\right)^{1/2} \leq C_K$ and the Cauchy-Schwarz inequality. By the same reasoning and since $\|f^*_{K,\gamma,n} - f^*_{K,\gamma}\| \leq \|f^*_{K,\gamma,n}\|$ we have that,

$$|f^*_{K,\gamma} - f^{*[n]}_{K,\gamma}| \leq C_K\|f^*_{K,\gamma} - f^{*[n]}_{K,\gamma}\| \leq C_K\|f^*_{K,\gamma}\| = C_K M, \qquad (3.65)$$

by using (3.45). □

Proof Theorem 3. By the Tchebychev inequality is straightforward to prove that

$$\begin{aligned} P\left\{\frac{1}{n}\|\mathbf{f}^*_{K,\gamma,\mathbf{z}} - \mathbf{f}^{*[n]}_{K,\gamma,\mathbf{z}}\|^2 - \|f^*_{K,\gamma} - f^{*[n]}_{K,\gamma}\|^2 \geq \varepsilon\right\} &\leq \frac{\mathsf{Var}\left(f^*_{K,\gamma,n} - f^*_{K,\gamma}\right)}{n\varepsilon^2} \\ &\leq \frac{C_K^2 M^4}{n\varepsilon^2} \end{aligned}$$

by applying (3.48) and since

$$\delta = \tfrac{C_K^2 M^4}{n\varepsilon^2} \text{ then, } \varepsilon = \sqrt{\tfrac{C_K^2 M^4}{n\delta}}.$$

then it hods that

$$P\left\{\frac{1}{n}\|\mathbf{f}^*_{K,\gamma,\mathbf{x}} - \mathbf{f}^{*[n]}_{K,\gamma,\mathbf{x}}\|^2 < \|f^*_{K,\gamma} - f^{*[n]}_{K,\gamma}\|^2 + \sqrt{\frac{C_K^2 M^4}{n\delta}}\right\} \geq 1 - \delta$$

what proves the statement. □

Chapter 4

Functional Data Analysis for proximity data with applications

Abstract

In this chapter we propose a Functional Data Analysis (FDA) approach to deal with proximity (similarity or distance) matrices in classification problems by estimating a particular class of integral operators. We analyze the connection between proximity measures and integral operators and we come up with a methodology able to estimate an integral operator whose associated kernel function, evaluated at the sample, will approach the sample proximity matrix of the problem. In particular, we develop the previous approach in three applications: (1) when the available information for the data is an asymmetric similarity matrix, (2) in partially labeled classification problems and (3) in similarities combination procedures.

Keywords: Integral Operators, Kernel Matrix, Classication, Asymmetric Similarities, Kernel Combinations.

4.1 Introduction

Consider a classification problem with two classes. Data can be given as a sample $s_n = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ where $\mathbf{x}_i \in X$ (some subset of $\mathbb{R}^p$) and $y_i \in \{-1, 1\}$ are the labels. In other cases, the available information is a proximity (distance or similarity) matrix **S** between the data. Distances can be easily transformed into similarities (see, for instance, (Gower, 1986)). Most classification algorithms will use either the sample s_n or the similarity matrix S to build a discriminant function.

In this work we use a Functional Data Analysis (FDA) approach to deal with proximity (or distance) matrices in classification problems by estimating certain integral operators associated to such matrices. In FDA the concept of vector (finite dimensional setting) is generalized to the concept of function (infinite dimensional case). Analogously, matrices (linear transformations) will generalize to operators. Consider, for instance, Functional Principal Components Analysis (FPCA) (Ramsay and Silverman, 2006), the generalization of PCA to the infinite dimensional case (where the counterpart of the data points $\mathbf{x}_i \in X$ are now functions $x_i(t)$, $t \in [0, T]$). While (assuming centered data) in PCA we need to calculate the eigenvectors $\mathbf{v}_j$ of the data covariance matrix $\mathbf{C} = n^{-1}\mathbf{X}^T\mathbf{X}$, in FPCA we have to obtain the eigenfunctions ϕ_j of the covariance operator defined by $(L_c f)(\mathbf{s}) = \int c(\mathbf{s}, \mathbf{t}) f(\mathbf{t}) dt$ where $c(s,t) = n^{-1}\sum_{i=1}^{n} x_i(s)x_i(t)$ is the sample covariance function. Therefore the eigensystem $\mathbf{C}\mathbf{v}_j = l_j\mathbf{v}_j$, for $j = 1, \ldots, n$ generalizes to the eigensystem $L_c\phi_j = \lambda_j\phi_j$ (for appropriate $l_j, \lambda_j \in \mathbb{R}$) and the matrix $\mathbf{C}$ generalizes to the integral operator L_c.

As will be made clear in next section, integral operators naturally arise when considering proximity matrices and therefore will play a crucial role in what follows. Therefore in this chapter we are mainly interested in estimating integral operators from proximity matrices. Section 4.2 is devoted to this task. In Section we 4.3 show three main applications of the previous theory: (1) when the available information for the data is an asymmetric similarity matrix, (2) in partially labeled classification problems and (3) in similarities combination procedures. In Section 4.4 we present some conclusion of this work.

4.2 Estimating Integral operators from proximity matrices

Let $C(X)$ be the Banach space of continuous functions in X (X a compact space) with the norm $\|f\|_\infty = sup_{\mathbf{x}\in X}|f(\mathbf{x})|$. Let $L^2_\nu(X)$ the space of square integrable functions in X where ν is a Borel measure. Let $K : X \times X \to \mathbb{R}$ be a continuous functions. Then the (linear) map $L_K : L^2_\nu(X) \to C(X)$ defined by the operator

$$(L_K f)(\mathbf{x}) = \int K(\mathbf{x}, \mathbf{t}) f(\mathbf{t}) d_\nu(\mathbf{t}), \tag{4.1}$$

is well defined and the function K is called the kernel of L_K. Several properties of L_K can be derived from the properties of K. For instance, if K is continuous then L_K is compact and $\|L_K\| \leq \sqrt{\nu(X)} \sup_{x,t \in X} |K(x,t)|$ where is $\nu(X)$ the measure of X (Cucker and Smale, 2001). In the sequel we will exclusively concentrate on continuous, symmetric and positive definite kernels which are known as Mercer's kernels (Mercer, 1909). Then L_K is self-adjoint, positive, compact and the Spectral theorem applies (Hochstadt, 1973; Conway, 1990): There exists a countable sequence of eigenvalues $\lambda_j \in \mathbb{R}$ and corresponding eigenfunctions ϕ_j ($j \geq 1$) of L_K. By Mercer's theorem (Mercer, 1909; Hochstadt, 1973) function K can be expressed as $K(\mathbf{x}, \mathbf{t}) = \sum_{j=1}^{\infty} \lambda_j \phi_j(\mathbf{x}) \phi_j(\mathbf{t})$, where the convergence is absolute (for each $\mathbf{x}, \mathbf{t} \in X \times X$) and uniform (on $X \times X$).

Next we show the relationship between distance functions and integral operators via the use of K to conclude that the natural operators corresponding to similarity (distances) matrices are integral operators.

Proposition 4.1. *Let X be a compact space.*

(i) Consider an integral operator $L_K : L^2_\nu(X) \rightarrow C(X)$ with associated Mercer kernel function $K(\boldsymbol{x}, \boldsymbol{t}) = \sum_{j=1}^{\infty} \lambda_j \phi_j(\boldsymbol{x}) \phi_j(\boldsymbol{t})$. Let's define the map $\Phi : X \rightarrow l^2$ by $\boldsymbol{x} \mapsto \left(\sqrt{\lambda_j}\phi_j(\boldsymbol{x})\right)_{j \in \mathbb{N}}$ (where l^2 is the linear space of all square summable sequences). Let's define a function $d_K : X \times X \rightarrow \mathbb{R}^+$ by

$$d_K(\boldsymbol{x}, \boldsymbol{t}) = d(\Phi(\boldsymbol{x}), \Phi(\boldsymbol{t})) = \|\Phi(\boldsymbol{x}) - \Phi(\boldsymbol{t})\| = \left(\sum_{j \in \mathbb{N}} \lambda_j (\phi_j(\boldsymbol{x}) - \phi_j(\boldsymbol{t}))^2 \right)^{1/2}, \quad (4.2)$$

where d represents the Euclidean distance in l^2.

Then if Φ is injective then the function d_K induced by L_K on X is a metric and Φ is an isometric mapping between (X, d_K) and (l^2, d).

(ii) Consider a random sample $s_n = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\} \subset X$ and let $\boldsymbol{M}$ denote a symmetric and positive definite $n \times n$ proximity matrix ($(\boldsymbol{M})_{ij}$ represents the proximity between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$). Then, there exists an integral operator L_{K^*} such that

$$K^*\big|_{s_n} = \boldsymbol{M}. \quad (4.3)$$

That is $K^*(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{M})_{ij}\ \forall\ \boldsymbol{x}_i, \boldsymbol{x}_j \in s_n$, where K^* is the Mercer kernel associated to L_{K^*}.

Proposition 4.1 represents a manner to relate linear integral operators associated to Mercer kernels with proximity (distance/similarity) matrices. Given a positive definite matrix **M** Proposition 4.1 ii) guarantees the existence of a kernel function

$$K^*(\mathbf{x},\mathbf{y}) = \sum_{j=1} l_j \varphi_j(\mathbf{x})\varphi_j(\mathbf{y}), \tag{4.4}$$

for $l_j > 0$ and $\{\varphi_1, \varphi_2, \dots\}$ an orthogonal set of continuous functions in X such that $K^*\big|_{s_n} = \mathbf{M}$. That is, its evaluations on the sample correspond to the elements of $\mathbf{M}$. Of course, the basis $\{\varphi_1, \varphi_2, \dots\}$ in expansion shown in eq. (4.4) is not necessary unique and the question of how to choose it appropriately arises. By Mercer's theorem the "true" K^* associated to $\mathbf{M}$ (that is $K^*\big|_{s_n} = \mathbf{M}$) will admit an expansion $K^*(\mathbf{x},\mathbf{t}) = \sum_{j=1} \lambda_j \phi_j(\mathbf{x})\phi_j(\mathbf{t})$ where ϕ_j are the eigenfunctions of K^* and therefore they represent a natural choice for the φ_j in eq. (4.28). Of course these eigenfunctions are unknown since the true kernel K^* is not available. Consider the spectral decomposition of $\mathbf{M}$, given by $\mathbf{M} = \sum_{j=1}^{n} l_j \mathbf{v}_j \mathbf{v}_j^T$ where $(l_j, \mathbf{v}_j)$ are the pairs of eigenvalues and eigenvectors of $\mathbf{M}$. Next we show that when the sample size increases the vectors $\mathbf{v}_j$ for $j = 1, \dots, n$ converge to the true eigenfunctions ϕ_j of L_{K^*}.

Theorem 4.1. *Let L_K be the integral operator associated to a kernel function $K : X \times X \to \mathbb{R}$. Let $\nu(X)$ a Borel measure in X and ν_n the empirical measure defined by $\nu_n(X) = \frac{1}{n}\sum_{j=1}^{n} I_X(\boldsymbol{x}_i)$ where I_X is the indicator function in X. Let $(L_K^n f)(\boldsymbol{x}) = \int K(\boldsymbol{x},\boldsymbol{t}) f(\boldsymbol{t}) d\nu_n(\boldsymbol{t})$ be the corresponding empirical integral operator, that is:*

$$(L_K^n f)(\boldsymbol{x}) = \frac{1}{n}\sum_{i=1}^{n} K(\boldsymbol{x},\boldsymbol{x}_i) f(\boldsymbol{x}_i) \tag{4.5}$$

Let $\kappa = \sqrt{\sup_{x \in X} K(\boldsymbol{x},\boldsymbol{x})}$, $s_n = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_n\}$ a random sample independently drawn from $\nu(X)$ and $K(\boldsymbol{x}_i,\boldsymbol{x}_j) = \left(K\big|_{s_n}\right)_{ij}$ the resulting data kernel matrix components. Let $\{\phi_j, \lambda_j\}$ denote the pairs of eigenvalues and eigenfunctions of L_K. Then the following statements hold:

(i) Convergence of L_K^n to L_K: Let H_K the RKHS associated to K and $HS(H_K)$ the Hilbert space of Hilbert-Schmidt operators on H_K. Then with a confidence of $1 - \delta$:

$$\|L_K^n - L_K\|_{HS} \le \frac{4\kappa^2 \log(2/\delta)}{\sqrt{n}}, \tag{4.6}$$

where $\|L_K\|_{HS} = \|K\|_{L^2_\nu(X)}$.

(ii) Convergence of the eigenvalues of L_K^n to the eigenvalues of L_K with a confidence of $1 - \delta$:

$$\sup_{j \ge 1}(\lambda_j - \hat{\lambda}_j) \le \frac{4\kappa^2 \log(2/\delta)}{\sqrt{n}} \tag{4.7}$$

where the $\{\lambda_j\}$ and $\{\hat{\lambda}_j\}$ are the sets of eigenvalues of L_K and L_K^n respectively.

(iii) *Convergence of the eigenfunctions of L_K^n to the eigenfunctions of L_K : For $r = min\{\lambda_{j-1} - \lambda_j, \lambda_j - \lambda_{j+1}\} > 0$, $\delta \in (0,1)$ and $\forall n \in \mathbb{N}$ then $\frac{4\kappa^2 \log(2/\delta)}{\sqrt{n}} \leq \frac{r}{2}$ and with confidence $1-\delta$*

$$\left\| \frac{\phi_j}{\sqrt{\lambda_j}} - \hat{\phi}_j \right\|_K \leq \frac{16\kappa^2 \log(2/\delta)}{r\sqrt{n}}, \tag{4.8}$$

where

$$\hat{\phi}_j(\boldsymbol{x}) = \frac{1}{\sqrt{n\hat{\lambda}_j}} \sum_{j=1}^{n} K(\boldsymbol{x}, \boldsymbol{x}_i)\boldsymbol{u}_{ij} \tag{4.9}$$

is the j-th normalized eigenfunction of L_K^n associated to the j-th eigenvalue $\hat{\lambda}_j$. Moreover, the pairs $\{l_j, \boldsymbol{u}_j\}$ are the eigenvalues and normalized eigenvectors of $\frac{1}{n}K\big|_{s_n}$. That is $\boldsymbol{u}_j = \frac{1}{\sqrt{l_j}}\boldsymbol{v}_j$ for $\boldsymbol{v}_j$ the eigenvectors of $\frac{1}{n}K\big|_{s_n}$.

Theorem 4.1 adapts to our problem several results form (Smale and Zhou, 2007). Points i) and ii) can be deduced from Proposition 1 and Proposition 2 of the paper (Smale and Zhou, 2007). Statement i) represents a particular case of the Theorem 2 in (Smale and Zhou, 2007) (with non normalized kernels) that can also be deduced from the previous propositions. Next we check that eq. (4.9) provides a particular manner of to find a basis (to construct the kernel function) such that Proposition 4.1 ii) holds.

Proposition 4.2. *Let L_K be the integral operator associated to a kernel function $K : X \times X \to \mathbb{R}$. Let $\nu(X)$ a Borel measure in X, and $s_n = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ a random sample independently drawn from $\nu(X)$. Let L_K^n be its associated empirical operator defined as Theorem 4.1 and $(K\big|_{s_n})_{jk} = K(\boldsymbol{x}_i, \boldsymbol{x}_k)$ the elements of the kernel matrix associated to K and s_n. Then*

$$\sum_{j=1}^{n} \hat{\lambda}_j \hat{\phi}_j(\boldsymbol{x}_i)\hat{\phi}_j(\boldsymbol{x}_k) = \left(K\big|_{s_n}\right)_{ik}, \tag{4.10}$$

for any $\boldsymbol{x}_i$, $\boldsymbol{x}_k$ in s_n where $\hat{\lambda}_j$ are the eigenvalues of $\frac{1}{n}K\big|_{s_n}$ and $\hat{\phi}_j$ are the eigenfunctions of L_K^n given in eq. (4.9).

Eq. (4.9) represents a particular case of basis such that Proposition 4.1 ii) holds. However eq. (4.9) is not always estimable and some alternative basis have to be proposed to construct K^*. We illustrate this in next section with three real applications.

4.3 Applications

Next we show three scenarios where the knowledge of integral operators associated to Mercer kernels helps to improve the performance of discrimination procedures: classification with partially labeled data, with asymmetric proximity similarities and problems the data similarity matrix is a combination of some original similarities and the labels.

4.3.1 Classification with partially labeled data

Classification with partially labeled data (Chapelle et al., 2006) deals with discrimination problems where the data set consist of some labeled points and the remaining unlabeled. The idea is to use the structure of the unlabeled data to help to improve the classification procedure. This situation often arises in real situations (Abney, 2008), where the proportion of unlabeled data is usually higher than the proportion of labeled data. In this section we show how to address this problem in the framework of integral operator estimation.

Consider a training sample $s_n = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_t, y_t), \mathbf{x}_{t+1}, \dots, \mathbf{x}_n\}$ made up of a subset s_t of t labeled points and a subset s^u_{n-t} of $n-t$ unlabeled, where $\mathbf{x}_i \in X$ (a compact set of $\mathbb{R}^p$) and $y_i \in \{-1, 1\}$. Let $S : \mathbb{R}^p \times \mathbb{R}^p \rightarrow \mathbb{R}^+$ be a proximity function (or a function directly related to S). Most discrimination procedures use the matrix $S\big|_{s_t}$ (whose elements are given by $(S\big|_{s_t})_{ij} = S(\mathbf{x}_i, \mathbf{x}_j)$ for $i, j = 1 \dots, n$) and the labels as input for the classification algorithm. For instance, in Fisher Discriminant Analysis (FDA) (assuming centered data) the similarity matrix $S\big|_{s_t}$ is some transformation of the matrix $t^{-1}\mathbf{X}\mathbf{X}^T$ where $\mathbf{X} \in \mathbb{R}^{t \times p}$ is the matrix of labeled data. On the other hand when the available information is a similarity matrix (instead of a set of data points), we can always obtain the data coordinates transforming the similarities into distances by $d_{ij} = 1 - s_{ij}$ (see (Gower, 1986) for more alternatives) and then applying some multidimensional scaling (MDS) procedure.

To build a discriminative procedure that integrates the unlabeled data points we will proceed in three steps. In the first step we obtain a similarity function $S^* : \mathbb{R}^p \times \mathbb{R}^p \rightarrow \mathbb{R}^+$ from S that takes into account both labeled and unlabeled data points, thus $S^* = F(S, s_n)$. The idea is to use the neighborhood information provided by the labeled points to increase the similarity of unlabeled points that are close to points of the same class and to decrease it otherwise. For instance we can define S^* as:

$$S^*(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{2}\left(S(\mathbf{x}_i, \mathbf{x}_j) + S(\mathbf{z}_{x_i}, \mathbf{z}_{x_j})\right) = +\tau y_{x_i} y_{x_j} \left|S(\mathbf{x}_i, \mathbf{x}_j) - S(\mathbf{z}_{x_i}, \mathbf{z}_{x_j})\right| \tag{4.11}$$

for $\mathbf{x}_i, \mathbf{x}_j \in s_n$, $\sigma \geq 0$ and where $z_{\mathbf{x}_i}$ and $z_{\mathbf{x}_j}$ are the closest labeled data points to $\mathbf{x}_i$ and $\mathbf{x}_j$ with labels y_{x_i}, y_{x_j}. Therefore for two unlabeled points the similarity S is increased by $\tau \left|S(\mathbf{x}_i, \mathbf{x}_j) - S(\mathbf{z}_{x_i}, \mathbf{z}_{x_j})\right|$ when $\mathbf{z}_{x_i}$ and $\mathbf{z}_{x_j}$ belong to the same class and decreased in the same quantity otherwise. In addition it is straightforward to show that $S^*(\mathbf{x}_i, \mathbf{x}_j) = S(\mathbf{x}_i, \mathbf{x}_j)$ when both points are labeled.

The matrix $S^*\big|_{s_t}$ is not necessarily positive definite and therefore it cannot be directly used as input for the algorithms. Thus, the second step will be to project the sample matrix $S^*\big|_{s_t}$ onto the cone of positive definite matrices (See (Muñoz and Martín de Diego, 2006) for a battery of methods to do this). Denote by $\mathbf{M}$ this projection. Next, in the third step of the process, we estimate an integral operator L_{K*} such that K^* is defined such that $(\mathbf{M})_{ij} = K^*(\mathbf{x}_i, \mathbf{x}_j)$. To this aim we proceed by estimating its eigenvalues and eigenfunctions. We first decompose $\mathbf{M}/n = \sum_{j=1}^{d} l_j \mathbf{v}_j \mathbf{v}_j^T$ where the pairs $(l_j, \mathbf{v}_j)$ are the eigenvalues and eigenvectors of $\mathbf{M}/n$ and $d = rank(\mathbf{M}/n)$. Following Theorem 4.1, we can estimate each λ_j (the eigenvalues of L_{K^*}) by $\hat{\lambda}_j = l_j$. In addition following eq. (4.31) we know that the eigenfunctions of K^* verify that $\phi_j(\mathbf{x}_i) = \sqrt{n}\mathbf{v}_{ij}$. In Proposition 4.1, our choice for such eigenfunctions was an orthogonal basis of polynomials (see proof for details). In this case, we will use the neighborhood information of the data to determine them. Here we propose to estimate $\phi_j(\mathbf{x}_k)$ for each test point $\mathbf{x}_k$ as a weighted sum of $\sqrt{n}\mathbf{v}_{j1}, \ldots, \sqrt{n}\mathbf{v}_{jn}$. We define

$$\hat{\phi}_j(\mathbf{x}_k) = \sqrt{n} \sum_{i=1}^{n} \theta_{ki} \mathbf{v}_{ji} \tag{4.12}$$

where the weight $\theta_{ki} = \exp\{-\gamma \|\mathbf{x}_k - \mathbf{x}_i\|^2\} \left(\sum_{h=1}^{n} \exp\{-\gamma \|\mathbf{x}_k - \mathbf{x}_h\|^2\}\right)^{-1}$ are positive and $\sum_{i=1}^{n} \theta_{ki} = 1$. The final estimated kernel is given, for any two points $\mathbf{x}_l$ and $\mathbf{x}_k$ by $\hat{K}^*(\mathbf{x}_l, \mathbf{x}_k) = \sum_{j=1}^{d} \hat{\lambda}_j \hat{\phi}_j(\mathbf{x}_l) \hat{\phi}_j(\mathbf{x}_k)$. Then the final estimated kernel is :

$$\hat{K}^*(\mathbf{x}_l, \mathbf{x}_k) = \sum_{j=1}^{d} l_j (\boldsymbol{\theta}_l^T \mathbf{v}_j)(\boldsymbol{\theta}_k^T \mathbf{v}_j) \tag{4.13}$$

where $\boldsymbol{\theta}_k = (\theta_{ki}, \ldots, \theta_{kn})^T$ for any $\mathbf{x}_k$. This kernel allows to estimate the components of $\mathbf{M}$ for test data that generally are non available when the solution to the classification problem is estimated.

To conclude this section we summarize the previous methodology in Table 4.1 and we illustrate its performance in a simulated example. In Chapter 6 a battery of real real examples using the previous methodology are also included.

Example 4.1. Consider a two-class classification problem where the classes are realizations of bivariate normal distributions with equal covariance matrices. In particular we generate a sample s_n of $n = 1000$ data (500 for each class) made up from bivariate normal distributions $N(\mu_i, I)$, with $\mu_1 = (0, 0)$ and $\mu_2 = (0, 0)$ respectively. See Figure 4.2

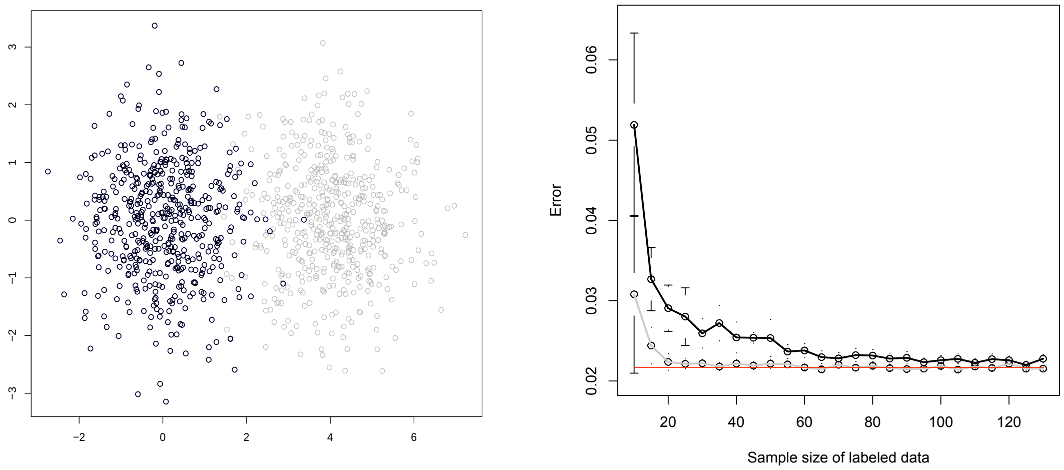
INPUT		
	s_n:	Partially labeled sample $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_t, y_t), \mathbf{x}_{t+1}, \dots, \mathbf{x}_n\}$
	σ, γ:	Parameters of the procedure.
	S:	Similarity function.
OUTPUT		
	$\hat{\Phi}$:	Estimated embedding for s_n.
	$\hat{K}^*$:	Estimated kernel associated to the integral operator L_{K^*}.
STEP 1:		**Modify the original similarity fucntion S**
	1.1)	Define S^* as in eq. (4.11).
	1.2)	Calculate the matrix $S^*\big\|_{s_t}$.
STEP 1:		**Project $S^*\big\|_{s_t}$ onto the cone of pos. def. matrices.**
	2.1)	Obtain the projected matrix $\mathbf{M}$.
STEP 2:		**Estimate the integral operator associated to $\mathbf{M}$**
	3.1)	Estimate each ϕ_j by eq. (4.12) for $j = 1, \dots, d$.
	3.2)	Estimate each $\hat{\lambda}_j$ by l_j for $j = 1, \dots, d$.
	3.3)	Estimate $\hat{K}^*$ the kernel of the integral operator L_{K^*} following eq. (4.13) where the $\hat{\phi}_j$ are given in Step 3.1 and $\hat{\lambda}_j$ in Step 3.2.

Table 4.1: Integral operator estimation for partially labeled classification problems

a). In addition a population of 20000 test points (10000 of each class) is also generated following the same distribution. In this problem the optimal linear discrimination function is given by $x = 2$ and the theoretical Bayes error of the solution equals to 2.227%.

The Fisher Discriminant Analysis (LDA) is optimal Bayes for this problem. Of course it can not obtain an optimal perfomance when the sample size is small. In this example we illustrate how the LDA can be improved in this situations by using the information provided by non labeled data. To this aim we perform the following experient. We determine several scenarios selecting (randomly) from s_n and increasing number t of data form 10 to 250. These data will be considered the labeled data and those not selected, but in s_n will be the unlabeled data. For each scenario, we compare the averaged errors using LDA with the t labeled data and with the kernel matrix estimated via the previously proposed procedure (matrix $K^*\big|_{s_t}$). In both cases the test errors are estimated over the remaining 20000 test points as the average of 30 runs (for each scenario).

To estimate K^*, we need to define the similarity S^* first. Let $S(\mathbf{x}_i, \mathbf{x}_j) = 1 - (\mathbf{x}_i, \mathbf{x}_j)/d_{max}$ be the original similarity between the data where d_{max} is the maximum distance be-



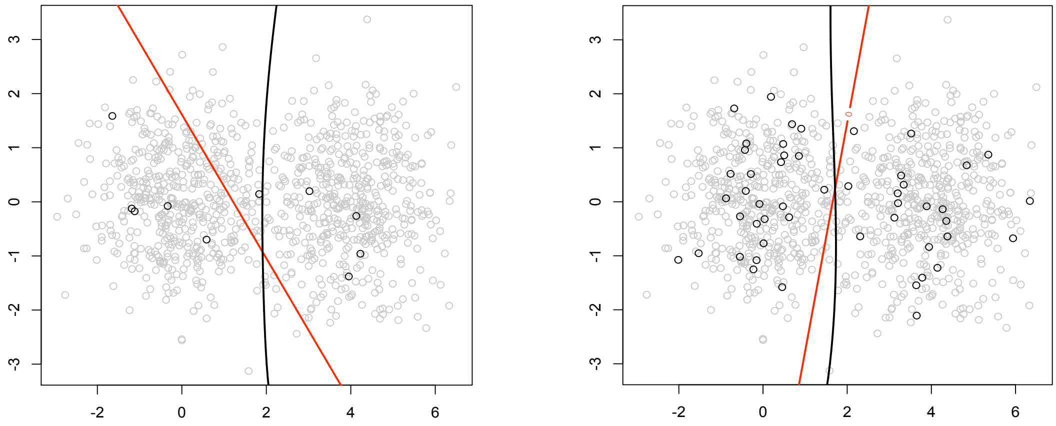
(a) Simulated data from a two classes clasifcation problem with equal covariance matrices.

(b) Convergence of the FDA error. We show the results for $S\big|_{s_t}$ (in black) and for $K^*\big|_{s_t}$ (in grey). Bayes pointed out shown in the horizontal line.

Figure 4.1: Classification problem where the classes are realizations of bivariate normal distributions and errors convergence of the Fisher Discriminant Analysis.

tween the training points. We modify S with the information of the labeled data via eq. (4.11). To this aim we fix the parameters γ and σ by cross validation in a grid of 25 points in the range $[0.5, 3]$. The matrix $S^*\big|_{s_t}$ is then projected onto the cone of positive definite matrices obtaining $\mathbf{M} = \sum_{j=1}^{d} max(l_j, 0)\mathbf{v}_j^T\mathbf{v}_j$ where the pairs $(l_j, \mathbf{v}_j)$ are the eigenvalues and eigenvectors of $S^*\big|_{s_t}$ and $d = rank(S^*\big|_{s_t})$. The final kernel function K^* is constructed, following eq. (4.13), considering the two first eigenfunction (those based on the largest eigenvalues).

Results are shown in Figure 4.1 b). In this plot the averaged errors (using S and K^*) are shown for different number of labeled data form starting in $t = 10$. The 95% confidence intervals are included in each case for all the scenarios. It is clear that the proposed methodology improves the errors obtained with the classical LDA specially when the the amount of labeled data is small. The new proposed methodology achieves errors significative equal the Bayes error for 20 data using a T-test with alternative hypothesis $H_1 : \mu_{error} < 0.0227$ obtaining a p-value of 0.0013. Notice that when $t = 20$ we have a total of $(1000 - 20)/20 = 49$ unlabeled data for each labeled point, which in this case is enough to improve the results. In the experimental section we will give more details regarding the "cost" of the technique in terms of the unlabeled data required to improve



(a) Discrimination function in one run of the experiment for the $LDA + S\big|_{s_t}$ and $LDA + K^*\big|_{s_t}$ when $t = 10$.

(b) Discrimination function in one run of the experiment for the $LDA + S\big|_{s_t}$ and $LDA + K^*\big|_{s_t}$ when $t = 50$.

Figure 4.2: Illustration of the change in the discrimination function estimated by the LDA when the information provided by the unlabeled data is considered.

the results. Regarding the simple LDA (using S), we need at least 95 labeled data to statistically reach the convergence (p-value for the T test equals to 0.0023) as can be seen in plot 4.1 b).

As final remark, we illustrate how the use of the unlabeled data modifies the original discrimination function estimated by the LDA. In Figure 4.2 a) and b) we include the estimated discrimination functions for $LDA+S$ and $LDA+K^*$ in one run of the experiment when the number of labeled points are 10 and 50. It is plain to see that the use of unlabeled data makes the decision function to be close to the optimal solution $(x = 2)$ in both cases. In is remarkable that, using K^*, such decision function is not linear. However, notice that when all the training samples are labeled $S = K^*$ and therefore both discrimination functions are equal.

• • •

4.3.2 Classification with asymmetric proximity matrices

Consider a binary classification problem where the proximities between the points are given by an asymmetric similarity matrix **S**. In this case there is no immediate way to obtain Euclidean coordinates and thus apply standard classification procedures.

In this section we use the approach shown in Section 4.2 to afford the task of embedding data defined by an asymmetric proximity matrix into a Euclidean space as a previous step to the use of any classification algorithm. To this aim we first obtain, with the aid of the data labels, a symmetric matrix $\mathbf{S}^*$ close to $\mathbf{S}$. Next we project $\mathbf{S}^*$ onto the convex cone of positive definite matrices. The projection of $\mathbf{S}^*$, $\Pi_+(\mathbf{S}^*)$, will induce an embedding of the data sample into a Euclidean space, in which the data points will be classified.

Symmetrization of S

The immediate way to obtain a symmetric matrix close to $\mathbf{S}$ is to consider the triangular decomposition of $\mathbf{S}$: let $\mathbf{S}_1$ and $\mathbf{S}_2$ be the two symmetric matrices built from the upper and lower triangular parts of $\mathbf{S}$ respectively. Then $\mathbf{S} = \frac{1}{2}(\mathbf{S}_1 + \mathbf{S}_2) + \frac{1}{2}(\mathbf{S}_1 - \mathbf{S}_2)$. Denote by $\mathbf{Y}$ the diagonal matrix containing the labels. We will obtain $\mathbf{S}^*$ as a function that combines $\mathbf{S}_1$, $\mathbf{S}_2$ and $\mathbf{Y}$:

$$\mathbf{S}^* = F(\mathbf{S}_1, \mathbf{S}_2, \mathbf{Y}), \tag{4.14}$$

where F is some function to implement the combination. We want $\mathbf{S}^*$ to be as similar as possible to $\mathbf{S}$, positive definite and suitable to derive the discrimination function. When $\mathbf{S}_1$ and $\mathbf{S}_2$ are positive-definite we can use semi-definite programming (Lanckriet et al., 2004) to look for a linear combination that optimizes some objective function involving $\mathbf{S}$ and the labels. Unfortunately we are not in this case because $\mathbf{S}_1$ and $\mathbf{S}_2$ are not necessarily positive-definite. Let $\mathbf{S}_y = \mathbf{Y}\mathbf{1}_n\mathbf{1}_n^T\mathbf{Y}$, where $\mathbf{1}_n$ is a column vector of n ones, the optimal discrimination matrix. To obtain $\mathbf{S}^*$ in eq. (4.14) we will express $\mathbf{S}^*$ as a linear combination of $\mathbf{S}_1$, $\mathbf{S}_2$ and $\mathbf{S}_y$ by:

$$\mathbf{S}^* = \frac{1}{2}(\mathbf{S}_1 + \mathbf{S}_2) + \tau\mathbf{S}_y \tag{4.15}$$

where $\tau > 0$. The intuition here is that if we increase similarities for points in the same class and decreased for points with different class labels then we expect the discrimination function to work better. The idea of using labels to transform a similarity matrix has been previously used in (Amari and Wu, 1999).

An alternative to obtain $\mathbf{S}^*$ in eq. (4.15) is to use the polar decomposition of S (Horn and Johnson, 1991; Higham, 1986). Consider $\mathbf{S} = \mathbf{U}\Sigma\mathbf{V}^T$ the singular value decomposition of $\mathbf{S}$ and define $\mathbf{Q} = \mathbf{U}\mathbf{V}^\mathbf{T}$. Then $\mathbf{S} = \mathbf{M}_1\mathbf{Q} = \mathbf{Q}\mathbf{M}_2$, where $\mathbf{M}_1 = \mathbf{U}\Sigma\mathbf{U}^T$ and $\mathbf{M}_2 = \mathbf{V}\Sigma\mathbf{V}^T$. Then substitute $\mathbf{S}_1$ by $\mathbf{M}_1$ and $\mathbf{S}_2$ by $\mathbf{M}_2$ in eq. (4.15).

Projecting $\mathbf{S}^*$ onto the cone of positive definite matrices

Given that $\mathbf{S}^*$ is not necessarily positive definite we will project it onto the convex cone of positive definite matrices (SPD) of size n defined by:

$$\mathsf{K}_+^n = \{\mathbf{K} = \mathbf{K}^T \in \mathbb{R}^{n \times n} : K \geq 0\} \tag{4.16}$$

where $\mathbf{K} \geq 0$ means that $\mathbf{K}$ is semi-positive-definite. Next we propose two different projection methods. The first is the orthogonal projection of $\mathbf{S}^*$ onto K_+^n, and can be calculated (Higham, 2002) by:

$$\Pi_+^1(\mathbf{S}^*) = \sum_{j=1}^{n} max(0, l_j)\mathbf{v}_j\mathbf{v}_j^T \tag{4.17}$$

where $\mathbf{v}_j$ are the eigenvectors of $\mathbf{S}^*$ and l_j its corresponding eigenvalues (some of them could be negative). Matrix $\Pi_+^1(\mathbf{S}^*)$ is usually known as the positive part of $\mathbf{S}^*$.

The second uses the method of Alternating Projections (AP)(Deutsch, 2001; Luenberger, 1969). Consider the set of matrices given by:

$$\mathsf{Q}^n = \{\mathbf{Q} \in \mathbb{R}^{n \times n} : q_{ii} = 1\}. \tag{4.18}$$

Let be $\mathsf{I}_+^n = \mathsf{K}_+^n \bigcap \mathsf{Q}^n$. We will obtain $\Pi_+^2(S^*)$ as the projection of S^* onto I_+^n using the AP method. Notice that the elements of Q^n can be interpreted as similarity matrices and thus $\Pi_+^2(\mathbf{S}^*)$ will be a positive-definite similarity matrix.

The AP method finds the closest matrix to $\mathbf{S}^*$ (in terms of the Frobenius norm) in the space I_+^n. To proceed, we create a sequence of alternating projections onto K_+^n and Q^n until the algorithm converges. The projections onto K_+^n are calculated by eq. (4.17) and the projections onto Q^n by setting to one the elements of the diagonal of the matrices. Since K_+^n and Q^n are close and convex spaces the convergence is ensured. In (Higham, 2002) a similar problem is solved for correlation matrices in the finance industry field.

Estimating an integral operator from $\Pi_+(\mathbf{S}^*)$

In this section we afford the problem of estimating a kernel function K^* such that $K^*\big|_{S_n} = \Pi_+(\mathbf{S}^*)$ where Π_+ is any of the two matrix projections described above. Notice that the existence of K^* is ensured by Proposition 4.1 ii).

By Mercer's theorem the (unknown) kernel function K^* admits an expansion $K^*(\mathbf{x},\mathbf{t}) = \sum_{h=1}\lambda_h\phi_h(\mathbf{x})\phi_h(\mathbf{t})$. We will build the kernel estimator $\hat{K}^*$ by replacing in the kernel expansion of K^*, λ_h and ϕ_h by estimators $\hat{\lambda}_h$ and $\hat{\phi}_h$.

To proceed, denote by $\{l_h, \mathbf{v}_h\}$ the pairs of eigenvalues and eigenvectors of $\frac{1}{n}\Pi_+(\mathbf{S}^*)$ and let $d = rank(\frac{1}{n}\Pi_+(\mathbf{S}^*))$ be the rank of $\Pi_+(\mathbf{S}^*)$. Following Propostion 4.2, an estimator of each λ_h of K^* is given by $\hat{\lambda}_h = l_h$.

Regarding the eigenfunctions ϕ_j of K^*, eq. (4.9) is the optimal manner to estimate them. Nevertheless this expression is usefulness here since the evaluations of K^* in any $\mathbf{x}_k$ out of the training sample are unknown. The labels are not available and therefore $\mathbf{S}^*$ (and obviously $\Pi_+(\mathbf{S}^*)$) just can be estimated for the sample data. To solve this problem we will make use of the matrices $\mathbf{S}_1$ and $\mathbf{S}_2$ (or $\mathbf{M}_1$ and $\mathbf{M}_2$ in the polar decomposition) whose components are available for training and testing points.

Let $\Pi_+(\mathbf{S}_1)$ and $\Pi_+(\mathbf{S}_2)$ be two kernel matrices obtained as the projections of $\mathbf{S}_1$ and $\mathbf{S}_2$ onto the cone of positive definite matrices. Notice that, for the polar decomposition we do not need to project $\mathbf{M}_1$ and $\mathbf{M}_2$ since they already are positive-definite. Let K_1 and K_2 two kernel functions such as $K_1(\mathbf{x}_i,\mathbf{x}_j) = (\Pi_+(\mathbf{S}_1))_{ij}$ and $K_2(\mathbf{x}_i,\mathbf{x}_j) = (\Pi_+(\mathbf{S}_2))_{ij}$ for all $\mathbf{x}_i$, $\mathbf{x}_j \in s_n$. Again, such functions exist by Proposition 4.1 ii) and their evaluations on test points are available (can be estimated for any test point $\mathbf{x}_k$ via eq. (4.9)).

The key idea to estimate each ϕ_h is to use the spectral information of $\frac{1}{n}\Pi_+(\mathbf{S}^*)$, $\frac{1}{n}\Pi_+(\mathbf{S}_1)$ and $\frac{1}{n}\Pi_+(\mathbf{S}_2)$ to define ϕ_h as a linear combination of the eigenfunctions of K_1 and K_2. Hence we will estimate each ϕ_h by

$$\hat{\phi}_h(\mathbf{x}) = \sum_{j=1}^{d_1}\hat{c}_{1j,h}\hat{\phi}_{1j}(\mathbf{x}) + \sum_{j=1}^{d_2}\hat{c}_{2j,h}\hat{\phi}_{2j}(\mathbf{x}) \quad for \quad h = 1,\ldots,d. \tag{4.19}$$

where d_1 and d_2 are the ranks of $\frac{1}{n}\Pi_+(\mathbf{S}_1)$ and $\frac{1}{n}\Pi_+(\mathbf{S}_2)$, $\{\hat{c}_{1j,h}\}$ and $\{\hat{c}_{2j,h}\}$ the weights of the mentioned combination and $\hat{\phi}_{1j}$, $\hat{\phi}_{2j}$ the approximations to the eigenfunctions of K_1 and K_2 given by eq. (4.9).

Several ways to determine $\{\hat{c}_{1j}\}$ and $\{\hat{c}_{1j}\}$ can be considered. Here we define them as follows. Let $\{\mathbf{w}_{11},\ldots,\mathbf{w}_{1d_1}\}$ and $\{\mathbf{w}_{21},\ldots,\mathbf{w}_{2d_2}\}$ be the sets of eigenvectors of $\frac{1}{n}\Pi_+(\mathbf{S}_1)$ and $\frac{1}{n}\Pi_+(\mathbf{S}_2)$ and define the matrix

$$\mathbf{W} = [\mathbf{w}_{11},\ldots,\mathbf{w}_{1d_1},\mathbf{w}_{21},\ldots,\mathbf{w}_{2d_2}]. \tag{4.20}$$

Then we determine each vector $\hat{\mathbf{c}}_h = [\hat{c}_{11,h}, \ldots, \hat{c}_{1d_1,h}, \hat{c}_{21,h}, \ldots, \hat{c}_{2d_2,h}]$ as the minimizer of

$$\hat{\mathbf{c}}_h = \arg\min \|\mathbf{v}_h - \mathbf{W}\mathbf{c}_h\|^2, \tag{4.21}$$

for $h = 1, \ldots d$. Notice that eq. (4.21) is a least squares problem whose solution can be obtained by solving the linear system $\mathbf{W}^T\mathbf{W}\mathbf{c}_h = \mathbf{W}^T\mathbf{v}_h$. Remark that geometrically, eq. (4.21) estimates the orthogonal projection of each $\mathbf{v}_h$ onto the space generated by $Span\langle \mathbf{w}_{11}, \ldots, \mathbf{w}_{1d_1}, \mathbf{w}_{21}, \ldots, \mathbf{w}_{2d_2}\rangle$.

Finally, the estimator of K^* given by

$$\hat{K}^*(\mathbf{x}, \mathbf{t}) = \sum_{h=1}^{d} \hat{\lambda}_h \hat{\phi}_h(\mathbf{x}) \hat{\phi}_h(\mathbf{t}), \tag{4.22}$$

for $\hat{\lambda}_h = l_h$ (h-th eigenvalue of $\frac{1}{n}\Pi_+(\mathbf{S}^*)$) and $\hat{\phi}_h$ given by eq. (4.19). Notice that $\hat{K}^*$ is a kernel function since it is symmetric, continous (the eigenfunctions are linear combinations of linear functions) and positive definite (the eigenvalues $\hat{\lambda}$ are are real and positive since the matrix $\Pi_+(\mathbf{S})$ is positive definite).

To conclude this section we summarize in Table 4.2 the main steps to estimate an integral operators from an available asymmetric similarity matrix **S**. In addition we include next an example to illustrate the utility of the proposed procedure.

Example 4.2. Let X a $n \times p$ matrix representing a text database where $x_{ij} = 1$ if the ith term appears in the document jth and 0 otherwise. Let $|x_i|$ denote the number of documents indexed by term ith and $|x_i \wedge x_j|$ the number of documents indexed by both i and j terms. Consider the following asymmetric similarity measure:

$$s_{ij} = \frac{|x_i \wedge x_j|}{|x_i|} = \frac{\sum_k min(x_{ik}, x_{jk})}{\sum_k x_{ik}}. \tag{4.23}$$

Measure s_{ij} can be interpreted as the degree in which topic represented by term i is a subset of topic represented by term j. This numeric measure of subsethood is originally proposed in (Kosko, 1991) in the contest of fuzzy set theory. Consider, for instance, a collection of documents containing the term “statistics". In this case a more specific term like “bayesian" will occur just in a subset. The relation between “bayesian" and “statistics" is strongly asymmetric, in the sense that the concept represented by the word “bayesian" is a subset of the concept represented by the word “statistics" but not conversely.

We consider in this example the 20 Newsgroups data set which is a collection of approximately 20,000 newsgroup documents, partitioned across 20 different topics (Lang,

INPUT	
$\mathbf{S}$:	Asymmetric similarity matrix.
$\mathbf{Y}$:	Diagonal matrix of sample data labels.
F:	Similarities combination procedure.
tol:	Tolerance.
OUTPUT	
$\hat{\Phi}$:	Estimated Euclidean embedding for $\mathbf{S}$.
$\hat{K}^*$:	Estimated kernel of the integral operator L_{K^*} associated to $\mathbf{S}$.
STEP 1:	**Symmetrization of S. Do A) or B)**
	A) Triangular decomposition of S
1.1)	Obtain $\mathbf{S}_1$ and $\mathbf{S}_2$ via the triangular decomposition of $\mathbf{S}$.
1.2)	Obtain $\mathbf{S}^* = F(\mathbf{S}_1, \mathbf{S}_2, \mathbf{Y})$ via eq. (4.14) for the sample data.
	B) Polar decomposition of S
1.1)	Estimate the SVD decomposition of $\mathbf{S}$, thus $\mathbf{S} = \mathbf{U}\Sigma\mathbf{V}^T$.
1.2)	Let $\mathbf{M}_1 = \mathbf{U}\Sigma\mathbf{U}^T$ and Let $\mathbf{M}_2 = \mathbf{V}\Sigma\mathbf{V}^T$.
1.3)	Obtain $\mathbf{M}^* = F(\mathbf{M}_1, \mathbf{M}_2, \mathbf{Y})$ via eq. (4.14) for the sample data.
STEP 2:	**Projection of $\mathbf{S}^*$ onto K^n_+ or $\mathbf{I}^n_+$**
	If *A) in STEP 1:*
2.1)	Project the similarity matrices $\mathbf{S}$, $\mathbf{S}_1$ and $\mathbf{S}_2$ onto K^n_+ to obtain the matrices $\Pi_+(\mathbf{S})$, $\Pi_+(\mathbf{S}_2)$ and $\Pi_+(\mathbf{S}_2)$. Assume $K^*\big\vert_{s_n} = \Pi_+(\mathbf{S})$, $K_1\big\vert_{s_n} = \Pi_+(\mathbf{S}_2)$ and $K_2\big\vert_{s_n} = \Pi_+(\mathbf{S}_2)$.
	If *B) in STEP 1:*
2.1)	Do not project. Assume that $K^*\big\vert_{s_n} = \mathbf{M}^*$, $K_1\big\vert_{s_n} = \mathbf{M}_1$ and $K_2\big\vert_{s_n} = \mathbf{M}_2$.
STEP 3:	**Estimation of the integral operator associated to K^***
3.1)	Obtain $\{l_h, \mathbf{v}_h\}_{h=1}^d$ the pairs of eigenvalues and eigenvectors of $\frac{1}{n}\Pi_+(\mathbf{S}^*)$.
3.2)	Obtain $\{\mathbf{w}_{1j}\}_{j=1}^{d_1}$ and $\{\mathbf{w}_{2j}\}_{j=1}^{d_2}$ the sets of eigenvectors of $\frac{1}{n}\Pi_+(\mathbf{S}_1)$ and $\frac{1}{n}\Pi_+(\mathbf{S}_2)$.
3.3)*	Build the matrix $\mathbf{W} = [\mathbf{W}_1^{tol}, \mathbf{W}_2^{tol}]$.
3.4)	For $h = 1, \ldots, d$ solve the linear system $\mathbf{W}^T\mathbf{W}\hat{\mathbf{c}}_h = \mathbf{W}^T\mathbf{v}_h$.
3.5)	Estimate each ϕ_h by eq. (4.25) for $h = 1, \ldots, d$ for $\hat{\mathbf{c}}_h$ in 3.4).
3.6)	Estimate $\hat{K}^*$ using eqs. (4.22).

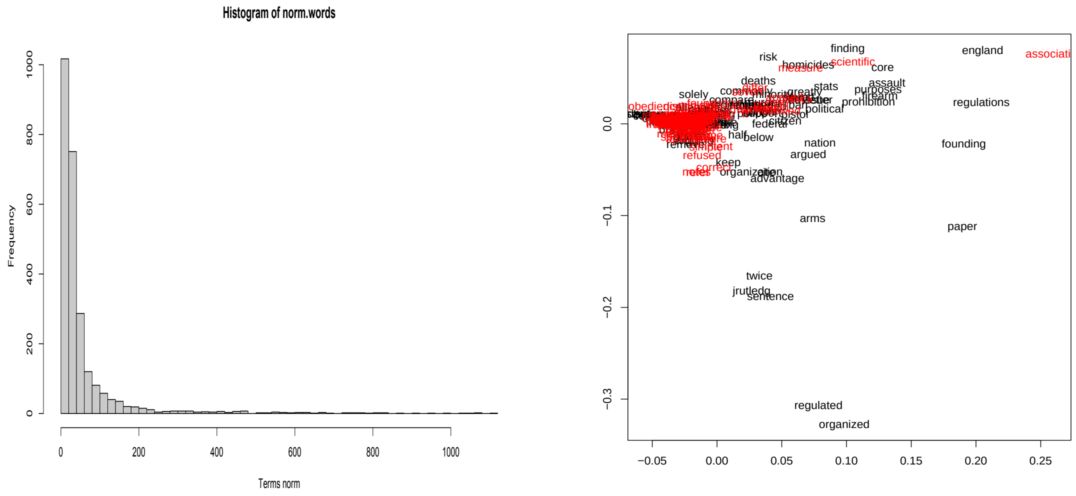
Table 4.2: Integral operator estimation procedure for asymmetric proximities. $\mathbf{W}_1^{tol}$ and $\mathbf{W}_1^{tol}$ have as columns the eigenvectors whose eigenvalues are smaller than the threshold tol.

1995). These 20 newsgroups collection has become a popular data set for experiments in text applications such as text classification and text clustering. For this experiment we take into account the two topics “Religion-Christian" and “Politics-Guns" since they are different matters with considerable overlapping. For instance words like “biblical", “word" and “temple" appear mostly in the religious-christian topic and terms like “articles", “charge" or “critique" in the politics-guns. However, there exists an overlap due to some common words to the two topics like “punished", “society" or ‘freedom". The final data set consists of a set of 1144 documents with a dictionary of 2564 words.

To run the experiments we select a sample of 1000 words and we use for the experiment those with a norm larger that 10. As a consequence we have a classification problem of 253 terms in dimension 1144 where the terms of the database are assigned to each group by voting. The histogram of the norms of the words, shown in Figure 4.3 a), verifies the Zipf law (Martín-Merino and Muñoz, 2005). There are a few terms with very large norms (appear in a lot of documents), and in the opposite side of the distribution, there are a lot of terms with very small norms. This indicates an asymmetric similarity between the terms (Martín-Merino and Muñoz, 2005) what can be also noticed in Figure 4.3 b). To build this graph we applied a Multidimensional Scaling (MDS) based on the Euclidean distance between the terms using its documents-frequency representation. That is, each term i is represented by a vector whose component j is estimated by $df_{ij} =$ (# times the term i appears in the document j) / (# times the term i appears in the data base). In this plot the terms of both classes present a hierarchical structure when they are represented in the plane what agrees with the information provided by Figure 4.3 a).

The objective of this experiment is to show that an appropriate use of the asymmetry helps to define distances appropriate in classification. To this aim we use the information provided by the matrix **S** estimated by eq. (4.23). Since **S** is asymmetric we perform its triangular (obtaining $\mathbf{S}_1$ and $\mathbf{S}_2$) and polar decomposition (having $\mathbf{M}_1$ and $\mathbf{M}_2$). We consider independently the two sources of asymmetry (obtained via the two matrices decompositions) and we combine both of them (separately) via the MAKM method via eq. (4.15) for $\tau = 10^{-3}$ and we project the combination matrices onto the cone of positive definite matrices using eq. (4.17).

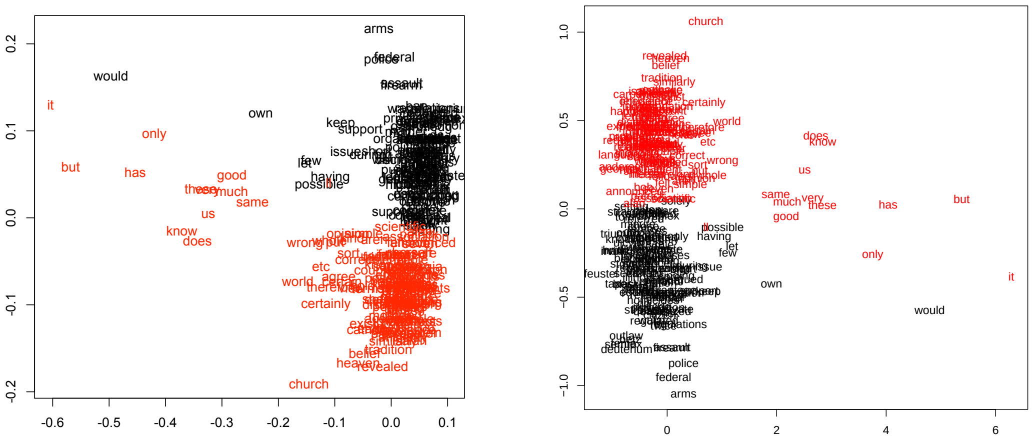
To show the behavior of our proposed approach we include Figures 4.4 a) and 4.4 b). In these two plots we show the MDS of the terms (for both the triangular and polar decomposition) with similarities obtained via eq. (4.15). Both mappings (similar up to sign of the components) totally change the representations of the terms compared to Figure 4.3 b). Now, the two classes of terms are clearly separated and the hierarchical structure is



(a) Histogram of the norms of the words following a Zipf law.

(b) MDS of the terms using de Euclidean distance.

Figure 4.3: Terms data frequency histogram an bidimensional representation via MDS.



(a) MDS of the terms data using the distance induced by combination of $\mathbf{S}_1$ and $\mathbf{S}_2$ (triangular decomposition) and the matrix labels $\mathbf{Y}$ in eq. (4.15) where $\tau = 10^{-3}$.

(b) MDS of the terms data using the distance induced by combination of $\mathbf{M}_1$ and $\mathbf{M}_2$ (polar decomposition) and the matrix labels $\mathbf{Y}$ in eq. (4.15) where $\tau = 10^{-3}$.

Figure 4.4: Comparison of the terms considering asymmetric similarities.

removed and then classification procedures will work better. In addition the two main components are clearly interpretable. The first (horizontal) separates words in terms of the semantic content. Thus in the extremes of this component we find words like “it" or “but" (without semantic content) and “outlaw" or “announce" (whose semantic content is strong). The second component clearly separates words associated to each class. In the extremes of this components we find the words “church" and “arms" which are very specific of each group. In addition, in the area where the classes are overlapped we find words like “announce" or “minute" which are not clearly identified with any of the groups.

In the experimental chapter we will study the classification results of a battery of classification methods using the previous defined similarities. In such experiments we will use the algorithm described in Table 4.2 to extend, to test points, the proposed similarities.

• • •

4.3.3 Classification within the kernel combination framework

In the discrimination context, it is common to have several sources of information that must be combined to design an optimized classifier (Kittler et al., 1998). A particular case of this problem is the combination of the sources of asymmetry described in Section 4.3.2. Here we will generalize the previous approach when a collection of two or more similarities/kernels are available to construct a classifier (Moguerza and Muñoz, 2006; Lanckriet et al., 2004). As in the previous case, the way to proceed is to design a single kernel function that collects all the relevant information of each available kernel and use it to train a classifier.

The best available techniques use the classification labels to combine kernel matrices and produce another kernel matrix (Moguerza and Muñoz, 2006; Muñoz and Martín de Diego, 2006), but not a kernel function. As a consequence, as in Section 4.3.2, there is no way to evaluate the combination kernel at points for which we do not know the label. The concrete goal of this section is to extent the methodology proposed in Section 4.3.2 in order to propose a consistent method to produce a kernel function from a similarity matrix calculated by any given similarity/kernel combination technique. Once the kernel function is available, it will be possible to evaluate the kernel at any data point and to use the information of the combination to train the classifier.

Consider a data set $s_n = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$ where $\mathbf{x}_i \in X$ (some subset of $\mathbb{R}^p$) and $y_i \in \{-1, 1\}$ are the labels of the data. Let $K_1, ..., K_m$ a set of kernel functions defined on X and let

$\mathbf{K}_1\big|_{s_n}, \ldots, \mathbf{K}_m\big|_{s_n}$ the corresponding kernel matrices such that $(\mathbf{K}_l\big|_{s_n})_{jk} = K_l(\mathbf{x}_j, \mathbf{x}_k)$ for $l = 1, \ldots, m$ and $j, k = 1, \ldots, m$. For simplicity of notation, we will use through this chapter $\mathbf{K}_l$ instead of $\mathbf{K}_l\big|_{s_t}$ to denote the kernel matrices of the set.

Next we transform each kernel matrix $\mathbf{S}_l$ onto a similarity matrix $\mathbf{S}_l$ for $l = 1, \ldots, m$ (see Section 2.7 for details) and we combine the set of obtained matrices by

$$\mathbf{S}^* = F(\mathbf{S}_1, ..., \mathbf{S}_m, \mathbf{Y}), \tag{4.24}$$

where $\mathbf{Y}$ is diagonal matrix whose non-null elements are the labels of the problem and F is any technique to combine the matrices $\{\mathbf{K}_l\}$ and $\mathbf{Y}$.

Notice that eq. (4.24) is generalization of eq. (4.14) for more that two similarities. Then the methodology in Section 4.3.2 can be easily generalized to estimate a kernel function K^* such that $K^*\big|_{s_n} = \Pi_+(\mathbf{S}^*)$ where S^* is the similarity matrix obtained in eq. (4.24) where Π_+ is any of the two matrix projection methods described in Section 4.3.2.

Denote $\Pi_+(\mathbf{S}_1), \ldots, \Pi_+(\mathbf{S}_m)$ the projections onto the cone of positive definite matrices of $\mathbf{S}_1, \ldots, \mathbf{S}_m$. We obtain the set of eigenvectors $\frac{1}{n}\{\mathbf{w}_{lj}\}$ of each $\Pi_+(\mathbf{S}_l)$ for $l = 1, \ldots, m$ and the eigenvectors $\frac{1}{n}\{\mathbf{v}_h\}$ of $\Pi_+(\mathbf{S}^*)$. Then, following the same reasoning that in the previous section, we will build the kernel estimator $K^*(\mathbf{x}, \mathbf{t}) = \sum_{h=1} \lambda_h \phi_h(\mathbf{x}) \phi_h(\mathbf{t})$ by replacing in the kernel expansion of K^*, λ_h and ϕ_h by estimators $\hat{\lambda}_h$ and $\hat{\phi}_h$. In this case we will estimate each ϕ_h by

$$\hat{\phi}_h(\mathbf{x}) = \sum_{l=1}^{m} \sum_{j=1}^{d_l} \hat{c}_{jl,h} \hat{\phi}_{jl}(\mathbf{x}) \quad for \quad h = 1, \ldots, d. \tag{4.25}$$

for d_l the rank of $\frac{1}{n}\Pi_+(\mathbf{S}_l)$. Notice that eq. (4.25) is a generalization of eq. (4.19) for more that two kernels. Next we build the matrix

$$\mathbf{W} = [\mathbf{w}_{11}, \ldots, \mathbf{w}_{1d_1}, \ldots, \mathbf{w}_{21}, \ldots, \mathbf{w}_{md_m}], \tag{4.26}$$

and we estimate $\hat{\mathbf{c}}_h = [\hat{c}_{11,h}, \ldots, \hat{c}_{1d_1,h}, \ldots, \hat{c}_{m1,h}, \ldots, \hat{c}_{md_m,h}]$ as the minimizer of eq. (4.21) for $h = 1, \ldots, d$ where $d = rank(\frac{1}{n}\Pi_+(\mathbf{S}^*))$. Finally, the final estimator of K^* is obtained, as in the previous section, by replacing in eq. (4.22) λ_h by the eigenvalues of $\frac{1}{n}\Pi_+(\mathbf{S}^*)$ and ϕ_h by the eigenfunctions estimated in 4.25. See Table 4.3 for a detailed description of the final algorithm.

Example 4.3. The algorithm proposed in Table 4.3 is designed to estimate the evaluations of kernel combinations in points out of the training sample. This is specially

INPUT	
$K_1,\dots,K_m$:	A set of kernel functions.
$\mathbf{Y}$:	Diagonal matrix of sample data labels.
F:	Similarities combination procedure.
tol:	Tolerance.
OUTPUT	
$\hat{K}^*$:	Estimated kernel function .
STEP 1:	**Kernel combination**
1.1)	Calculate $\mathbf{K}_1\big\vert_{s_n},\dots,\mathbf{K}_m\big\vert_{s_n}$ and $\mathbf{Y}$.
1.2)	Transform each kernel $\mathbf{K}_l$ into a similarity $\mathbf{S}_l$
1.3)	Combine the similarity matrices by $\mathbf{S}^* = F(\mathbf{S}_1, ..., \mathbf{S}_m, \mathbf{Y})$
STEP 2:	**Projection of $\mathbf{S}^*$ and the set of matrices $\{\mathbf{S}_l\}$ onto $\mathbf{K}_+^n$ or $\mathbf{I}_+^n$**
2.1)	Use some matrices projection and obtain $\Pi_+(\mathbf{S}^*)$ and $\Pi_+(\mathbf{S}_l)$ for $l = 1,\dots,m$. Assume $K^*\big\vert_{s_n} = \Pi_+(\mathbf{S}^*)$, $K_l\big\vert_{s_n} = \Pi_+(\mathbf{S}_l)$ for $l = 1,\dots,m$
STEP 3:	**Estimation of the integral operator associated to K^***
3.1)	Obtain $\{l_h, \mathbf{v}_h\}_{h=1}^{d}$ the pairs of eigenvalues and eigenvectors of $\frac{1}{n}\Pi_+(\mathbf{S}^*)$.
3.2)	Obtain $\{\mathbf{w}_{1j}\}_{j=1}^{d_l}$ the sets of eigenvectors of $\frac{1}{n}\Pi_+(\mathbf{S}_l)$ for $l = 1,\dots,m$.
3.3)*	Build the matrix $\mathbf{W} = [\mathbf{W}_1^{tol},\dots,\mathbf{W}_m^{tol}]$.
3.4)	For $h = 1,\dots,d$ solve the linear system $\mathbf{W}^T\mathbf{W}\mathbf{c}_h = \mathbf{W}^T\mathbf{v}_h$.
3.5)	Estimate each ϕ_h by eq. (4.25) for $h = 1,\dots,d$.
3.6)	Estimate $\hat{K}^*$ using eq. (4.22).

Table 4.3: Integral operator estimation procedure for proximities combination methods. $\mathbf{W}_1^{tol}$ and $\mathbf{W}_1^{tol}$ have as columns the eigenvectors whose eigenvalues are smaller than the threshold tol.

challenging when the labels, unknown for test data, are involved in the combination. In this experiment we will show the ability of the proposed procedure in this task.

We work with a data base consisting of radar data consisting of a phased array of 16 high-frequency antennas. The targets are the free electrons in the ionosphere (Sigillito et al., 1989) and the two classes are labeled as "Good" – radar returns showing evidence of some type of structure in the ionosphere – and "Bad" (returns without structure). There are 351 data points. We randomly select 200 for training and we consider the rest 151 for testing.

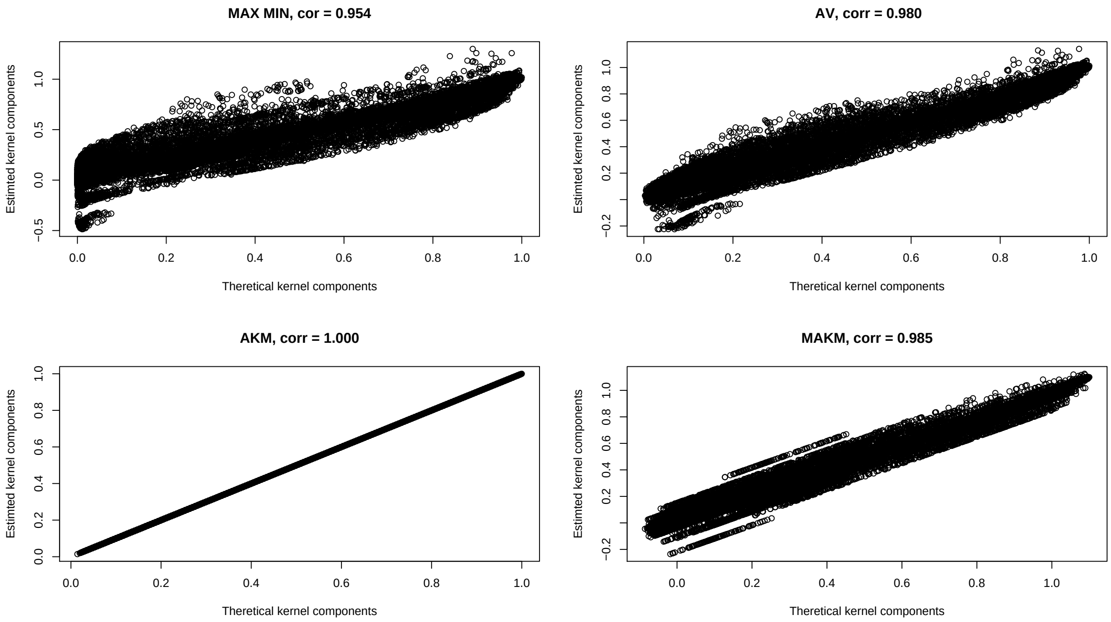


Figure 4.5: Scatterplot of the theoretical versus the estimated kernel matrices components for the four combinations schemes.

To show the ability of the proposed procedure approximating kernel combination components in test points, we consider a battery 10 Gaussian kernels $K(x, y) = exp\{-\rho\|x - y\|^2\}$ with parameters

$$\rho = \{0.189,\ 0.171,\ 0.104,\ 0.081,\ 0.069,\ 0.062,\ 0.057,\ 0.053,\ 0.050,\ 0.047\}.$$

We calculate the corresponding kernel matrices (over a training sample s_n of size 200) $\mathbf{K}_1, \dots, \mathbf{K}_{10}$ and we combine them using four combination schemes: $Max{-}Min$, $AV_{\tau=0.01}$, AKM and $MAKM_{\tau=0.01}$. See Section 2.8 for details.

To check the accuracy of the algorithm in Table 4.3 recovering the evaluations of the estimated kernels in test points, we use it to estimate the kernel functions $\hat{K}^*_{Max-Min}$, $\hat{K}^*_{AV}$, $\hat{K}^*_{AKM}$ and $\hat{K}^*_{MAKM}$. Then we evaluate these kernel functions on the 151 test points. Finally we draw, for the four combinations, an scatterplot of such estimated values versus its theoretical counterparts. Results are shown in Figure 4.5 where we also include the correlation between the theoretical and the estimated values. Notice that for the AKM method, that does not use the labels, the reconstruction is perfect. However for the AV and $MAKM$ methods (that are essentially a variation of the AKM using the labels) the approximation is good (correlations 0.98 and 0.985) but some error is achieved. Similar results are obtained for the Max-Min method.

It is apparent that the procedure proposed in Table 4.3 is estimating a kernel function that approximates well the kernel combinations in test points. Hence, such kernel can be used for classification purposed as we will do in the experimental chapter.

• • •

4.4 Conclusions and final remark

In this chapter we have shown how proximity matrices in classification problems can be handle via a FDA approach by estimating certain class of integral operators. This point of view offers some interesting advantages and can be applied to improve classification algorithms in several scenarios.

First of all, the previous approach has been tested in the context of partially labeled classification problems. In this cases, the information provided by unlabeled data points can be taken into account to improve the performance of classification procedures that only consider labeled data. In Section 4.3.1 we have afforded this problem via the estimation of an integral operator whose associated kernel function considers both, labeled and unlabeled data points. In particular we have shown that our methodology improves the Linear Discriminant Analysis in a simulated example.

Second, we have proposed a methodology to deal with asymmetric similarity matrices in classification problems. We have proposed a parametrized procedure to estimate and integral operator whose associated kernel mapping represents the data onto a Euclidean space. This approach has been tested successfully to represent the terms of a collection of documents and it will be tested in a battery of classification problems in Chapter 6.

Finally, we have used the previous approach to estimate a kernel function from any proximity matrices combination. We have shown that the new kernel function is able to extend, for out-of sample points, the components of the combination even when the data labels are involved in the combination process.

4.5 APPENDIX: Proofs

Proposition 4.1. We prove Proposition 4.1 as follows:

(i) By Mercer's theorem for every $x \in X$, $\sum_j \lambda_j \phi_j^2(x)$ converges to $K(x,x)$ that is $\phi(\mathbf{x}) \in l^2$. Given that $K(\mathbf{x},\mathbf{t}) = \sum_{j=1}^{\infty} \lambda_j \phi_j(\mathbf{x})\phi_j(\mathbf{t})$ for all $\mathbf{x},\mathbf{t} \in X$ the map $\Phi : X \to l^2$ given by $\mathbf{x} \mapsto \left(\sqrt{\lambda_j}\phi_j(\mathbf{x})\right)_{j \in \mathbb{N}}$ satisfies $K(\mathbf{x},\mathbf{t}) = \langle \Phi(\mathbf{x}), \Phi(\mathbf{t}) \rangle$. Thus K acts as a dot product in the embedding (the image of the map Φ) induced by the eigenfunctions of the operator L_K. Given $\mathbf{x},\mathbf{t} \in X$, by eq. (4.2) the Euclidean distance between two points in the image of ϕ is given by:

$$\begin{aligned} d^2(\Phi(\mathbf{x}), \Phi(\mathbf{t})) &= \langle \Phi(\mathbf{x}), \Phi(\mathbf{x}) \rangle + \langle \Phi(\mathbf{t}), \Phi(\mathbf{t}) \rangle - 2\langle \Phi(\mathbf{x}), \Phi(\mathbf{t}) \rangle \qquad (4.27) \\ &= K(\mathbf{x},\mathbf{x}) + K(\mathbf{t},\mathbf{t}) - 2K(\mathbf{x},\mathbf{t}). \end{aligned}$$

By eq. (4.2) it is trivial to check that for all $x, t, y \in X$ the function d_K satisfies: $d_K(\mathbf{x},\mathbf{t}) \geq 0$, $d_K(\mathbf{x},\mathbf{t}) = d_K(\mathbf{t},\mathbf{x})$ and $d_K(\mathbf{x},\mathbf{t}) \leq d_K(\mathbf{x},\mathbf{y}) + d_K(\mathbf{y},\mathbf{t})$. In addition since Φ is injective $d_K(\mathbf{x},\mathbf{t}) = 0$ if only if $\mathbf{x} = \mathbf{t}$ and $d_k : X \times X \to \mathbb{R}^+$. Therefore, every integral operator L_K induces a dissimilarity function d_K (on X) that will be a distance if the mapping Φ is injective.

Now consider the metric spaces (X, d_K) and (l^2, d). Since for all $\mathbf{x},\mathbf{t} \in X$, Φ is an injective mapping from (X, d_K) to (l^2, d) and $d_K(\mathbf{x},\mathbf{t}) = d(\Phi(\mathbf{x}), \Phi(\mathbf{t}))$, we conclude that Φ is an isometric mapping.

(ii) The second part of the proposition is proven as follows. Let $(l_j, \mathbf{v}_j)$ for $j = 1, \ldots, d$ denote the pairs of eigenvalues of eigenvectors of $\mathbf{M}$. Then $\mathbf{M} = \sum_{j=1}^{d} l_j \mathbf{v}_j \mathbf{v}_j^T$ where $d = rank(\mathbf{M})$.

For each $j = 1, \ldots, d$, consider the set of points $\{(\mathbf{x}_1, v_{j,1}), \ldots, (\mathbf{x}_n, v_{j,n})\} \subset X \times \mathbb{R}$. Consider interpolating polynomials $p_j : X \to \mathbb{R}$ for $j = 1, \ldots, d$ such that $p_j(\mathbf{x}_i) = v_{ji}$ (see (Gasca and Sauer, 2000; Lorentz, 2000) for a review of polynomial interpolation in several dimensions). Take $\varphi_j = p_j$ and define K^* by:

$$K^*(\mathbf{x},\mathbf{y}) = \sum_{j=1}^{d} l_j \varphi_j(\mathbf{x}) \varphi_j(\mathbf{y}). \qquad (4.28)$$

K^* is obviously symmetric, continuous, and positive definite (the l_j are the eigenvalues of M, a positive definite matrix). Thus K is a Mercer Kernel since $\{\varphi_1, \ldots, \varphi_d\}$ is a finite set of continuous functions in X (See (Rakotomamonjy and Canu, 2005) for conditions on φ's to be K^* a kernel).

By construction of K^*, $K^*\big|_{s_n} = \mathbf{M}$, that is $K^*(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{M})_{ij}$ for all $\mathbf{x}_i, \mathbf{x}_j \in s_n$ what concludes the proof.

□

Proof. Let $\{\hat{\lambda}_j, \hat{\phi}_j\}$ for $j = 1, \dots, n$ the pairs of eigenvalues and eigenfunctions of L_K^n. Then $(L_K^n \hat{\phi}_j)(x) = \hat{\lambda}_j \hat{\phi}_j(x)$ and thus

$$(L_K^n \hat{\phi}_j)(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^{n} K(\mathbf{x}, \mathbf{x}_i) \hat{\phi}_j(\mathbf{x}_i). \tag{4.29}$$

Then if $l_1, \dots, l_n$ is the spectrum of $\frac{1}{n} K\big|_{s_n}$, a natural estimator of the eigenvalues of L_K^n is given by

$$\hat{\lambda}_j = l_j. \tag{4.30}$$

In addition, the estimated eigenfunctions $\hat{\phi}_j$ given in eq. (4.9) (essentially the Nyström formula (Baker, 1977)) give raise the sample embedding (up to the factor $\sqrt{n}$) when evaluated in the sample: Let $\mathbf{x}_k$ a sample point, then

$$\begin{aligned} \hat{\phi}_j(\mathbf{x}_k) &= \frac{1}{\sqrt{n l_j}} \sum_{i=1}^{n} K(\mathbf{x}_i, \mathbf{x}_k) \mathbf{u}_{ij} \\ &= \frac{1}{l_j \sqrt{n}} \sum_{i=1}^{n} K(\mathbf{x}_i, \mathbf{x}_k) \mathbf{v}_{ij} \\ &= \frac{1}{l_j \sqrt{n}} n l_j \mathbf{v}_{jk} = \sqrt{n} \mathbf{v}_{jk} \end{aligned} \tag{4.31}$$

Consider now the expansion given by $\sum_{j=1}^{n} \hat{\lambda}_j \hat{\phi}_j(\mathbf{x}) \hat{\phi}_j(\mathbf{t})$ (kernel function associated to L_K^n) where $\hat{\lambda}_j$ and $\hat{\phi}_j$ are obtained in eqs. (4.30) and (4.9). Then applying the result in eq. (4.31):

$$\sum_{j=1}^{n} \hat{\lambda}_j \hat{\phi}_j(\mathbf{x}_i) \hat{\phi}_j(\mathbf{x}_k) = n \sum_{j=1}^{n} l_j \mathbf{v}_{ji} \mathbf{v}_{jk} = \left(K\big|_{s_n} \right)_{ik}, \tag{4.32}$$

for any $\mathbf{x}_i$, $\mathbf{x}_k$ in s_n. □

Chapter 5

Analysis of redundancies in proximities matrices combinations

Abstract

Information Fusion techniques are becoming increasingly important in fields such as Image Processing, Web Mining or Information Retrieval where is common to have several sources of information that must be combined. In this chapter we propose an spectral framework for information fusion when the sources of information are given by a set of proximity matrices. Our approach is based on the simultaneous diagonization of the original matrices of the problem and it represents a natural way to manage the redundant information involved in the fusion process. In particular, we define a new metric for proximity matrices and we propose a method that automatically eliminate the redundant information among a set of matrices when they are combined.

Keywords: Kernel Combinations, Redundant Information, Matrix Pencil, Simultaneous Diagonalization, Approximate Joint Diagonalization.

5.1 Introduction

Increasingly interest has focused in the last years in the development of statistical techniques that combine several sources of information. For instance, in Image Fusion (Choi et al., 2005), a typical problem considers different satellite pictures, with different resolutions and different color qualities, and the task is to produce a picture that has maximum resolution and the best color quality. In the field of Information Retrieval, the goal can be to classify a set of web pages (Joachims, 2002), and the information that has to be combined lies in the co-citation matrix and in the terms-by-documents matrix.

In the context of information fusion often happen that the sources of information are given by a set of proximity matrices and therefore their combination is the natural step to obtain a unified representation of the data. In this chapter we will focus on the problem of proximity (similarity/disimilarity) matrices combination in the context of Support Vector Machine (SVM) classification (Martín de Diego et al., 2009; Lanckriet et al., 2004) described in Chapter 2.

Consider a data set $s_n = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ where $\mathbf{x}_i \in X$ (some subset of $\mathbb{R}^p$) and $y_i \in \{-1, 1\}$ are the labels of the data. Let $K_1, ..., K_m$ a set of kernel functions defined on X and let $\mathbf{K}_1\big|_{s_n}, \ldots, \mathbf{K}_m\big|_{s_n}$ the corresponding kernel matrices such that $(\mathbf{K}_t\big|_{s_n})_{jk} = K_t(\mathbf{x}_j, \mathbf{x}_k)$ for $t = 1, \ldots, m$ and $j, k = 1, \ldots, m$. For simplicity of notation, we will use through this chapter $\mathbf{K}_t$ instead of $\mathbf{K}_t\big|_{s_n}$ to denote the kernel matrices of the set.

As we showed in Chapter 2, eq. (2.37) represents a kernel combination scheme that has been proven to achieve good results in classification problems. Rewriting eq. (2.37) in terms of the kernel matrices of $\mathbf{K}_1, \ldots, \mathbf{K}_m$, the final matrix of the combination $\mathbf{K}^*$ is given by

$$\mathbf{K}^* = \frac{1}{m}\sum_{t=1}^{m} \mathbf{K}_t + \tau \mathbf{Y} \sum_{t<l} g(\mathbf{K}_i, \mathbf{K}_j)\mathbf{Y}\,, \tag{5.1}$$

where $\tau > 0$, $\mathbf{Y} = diag(y_1, \ldots, y_n)$ and is g a convex continuous function, for instance $g(\mathbf{K}_i, \mathbf{K}_j) = |\mathbf{K}_i, -\mathbf{K}_j|$. Notice that we change slightly the point of view of Section 2.8 and we focus on the kernel matrices instead of the kernel functions.

The combination scheme in eq. (5.1) has two main ingredients: First, the average of the kernel matrices. Second, a term that includes the labels of the problem and some measure g that aims to capture the differences between each pair of kernels. Therefore, eq. (5.1) can be interpreted as manner to combine the "common" information between the kernels (measured by $m^{-1}\sum_{t=1}^{m} \mathbf{K}_t$) with their "independent" information (measured by $g(\mathbf{K}_i, \mathbf{K}_j)$).

In the second term of eq. (5.1), the function g acts element by element and the similarities between the matrices (viewed as elements of certain matrix space) is not extracted. One of our purposes of this chapter is to define dissimilarity measures δ :

$\mathbb{R}^{n\times n} \times \mathbb{R}^{n\times n} \rightarrow \mathbb{R}^{+}$ able to quantify the independent information between each pair of matrices $\mathbf{K}_i$ and $\mathbf{K}_j$ that can be used in scheme in eq. (5.1) to define new kernel combinations.

Next we focus on the average of the kernels $m^{-1}\sum_{t=1}^{m}\mathbf{K}_t$. Each positive-definite kernel function K_t induces a transformation of the data set into a (possibly) high dimensional Euclidean space $\mathbb{R}^{n_t}$. Following Theorem 4.1, the set of eigenvectors $V_t = \{\mathbf{v}_1, \ldots, \mathbf{v}_{n_t}\}$ of each kernel matrix $\mathbf{K}_t$ allows to approximate, for the sample points, the particular representation of the data set (induced by K_t) using some basis of $\mathbb{R}^{n_t}$ (see eq. (4.31)). If we want to combine the information provided by a set of m kernels, we will have to find some "common" basis $\{\mathbf{v}_1^*, \ldots, \mathbf{v}_{n^*}^*\}$ from the individual basis $V_1, \ldots, V_m$, such that the inmersion of the data set in the resulting $\mathbb{R}^{n^*}$ contains all the relevant information from the individual kernels. Any technique to produce the desired combination basis needs to take into account the problem of information redundance. In this sense to perform the direct sum (or the average) of $\mathbf{K}_1, \ldots, \mathbf{K}_m$ in eq. (5.1) presents a serious drawback. To illustrate it in a simple example, let us consider a data set, and two representations given by two projections on two pairs of principal axes (x, y) and (x, z), where the x variable is present in both representations. If we use the direct sum of the corresponding spaces as solution for the combination problem, we will have the representation (x, y, x, z). Thus, the weight of the x variable will be doubled when using the Euclidean distance and the results of the classification and regression algorithms will be distorted. In a general case, the correlation between the variables induces by the kernels $\mathbf{K}_1, \ldots, \mathbf{K}_m$ will cause similar problems when they are averaged.

To address the two previous issues we will follow an approach based on the simultaneous diagonalization of the matrices $\mathbf{K}_1, \ldots, \mathbf{K}_m$ involved in the problem. The idea is to find a matrix $\mathbf{V}$ whose columns constitute a basis of generalized eigenvectors such that the matrices given by $\mathbf{D}_t = \mathbf{V}^T\mathbf{K}_t\mathbf{V}$ for $i = 1, \ldots, m$ are all diagonal (or quasi diagonal). The diagonal elements of $\mathbf{D}_t$ (generalized eigenvalues) will be the key ingredient to manage the redundant infomation of the problem and to define new metrics for matrices.

This chapter is organized as follows. In Section 5.2 we define a general framework to define metrics for matrix spaces using the generalized eigenvalues of matrix pencils. In particular, in Section 5.2.1 we propose a new dissimilarity measure for matrices that is tested to behave well in real applications. In Section 5.3 we propose a new methodology, based on the Approximated Joint Diagonalization of matrices, to remove the redundant information when several kernel are combined. Finally we conclude in Section 5.4 with some comments and final remarks.

5.2 Proximities measures for matrices

The use of standard metrics inherited from Euclidean geometry may not be appropiate for many statistical problems where proximities matrices has to be compared. For instance, consider a isometry in $\mathbb{R}^n$, given by a matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ and let $\mathbf{K} \in \mathbb{R}^{n\times n}$ a kernel matrix. Then $\mathbf{K}$ transforms under $\mathbf{A}$ as $\mathbf{K}^* = \mathbf{A}\mathbf{K}\mathbf{A}^{-1}$. In this case, the Frobenious distance (FD), induced by the Frobenious norm between the original and the transformed matrix, is given by $d_F(\mathbf{K}, \mathbf{K}^*) = \|\mathbf{K} - \mathbf{K}^*\|_F \neq 0$ (Omladic and Semrl, 1990). However, since we have simply performed a change of basis it must happen that $d(\mathbf{K}, \mathbf{K}^*) = 0$ if d_F were an appropriate manner to measure the distance between the matrices.

The use of eigenvalues tends to avoid these problems because the spectrum of a matrix is invariant under many common transformations in statistics. An example is the distance induced by the *Spectral Norm* (Golub and Loan, 1997).

Definition 5.1. *Let $\boldsymbol{K}_1$ and $\boldsymbol{K}_2$ be two kernel matrices in $\mathbb{R}^{n\times n}$, then the distance $d_S(\boldsymbol{K}_1, \boldsymbol{K}_2)$ induced by the Spectral norm is given by*

$$d(\boldsymbol{K}_1, \boldsymbol{K}_2) = \sqrt{\lambda^1_{max}(\boldsymbol{K}_1, \boldsymbol{K}_2)}, \tag{5.2}$$

where $\lambda^1_{max}(\boldsymbol{K}_1, \boldsymbol{K}_2)$ represents the largest eigenvalue of $\boldsymbol{K}_1 - \boldsymbol{K}_2$.

The most usual proximity kernel measure in statistics and pattern recognition is the Kernel Alignment (KA) (Cristianini and Shawe-Taylor, 2002). It can be interpreted as a measure of linear relationship between two given kernel matrices.

Definition 5.2. *Let $\boldsymbol{K}_1$ and $\boldsymbol{K}_2$ be two kernel matrices in $\mathbb{R}^{n\times n}$, then the empirical Alignment is defined as*

$$A(\boldsymbol{K}_1, \boldsymbol{K}_2) = \frac{\langle \boldsymbol{K}_1, \boldsymbol{K}_2 \rangle_F}{\sqrt{\langle \boldsymbol{K}_1, \boldsymbol{K}_1 \rangle_F \langle \boldsymbol{K}_2, \boldsymbol{K}_2 \rangle_F}}. \tag{5.3}$$

where $\langle \boldsymbol{K}_1, \boldsymbol{K}_2 \rangle_F = \sum_{i,j} (\boldsymbol{K}_1)_{ij} (\boldsymbol{K}_2)_{ij}$ represents the Frobenius product.

This measure has several interesting properties and it has been used to optimize linear kernels combinations (Joachims, 2002).

Kernel Procrustes (KP) (Martín de Diego and Muñoz, 2006), can also be applied to measure the distance between two kernel matrices. Given two kernel matrices $\mathbf{K}_1$ and $\mathbf{K}_2$, the idea of kernel procrustes is to search for a matrix rotation $\mathbf{Q}$ matrix for $\mathbf{K}_2$ that makes it comparable to $\mathbf{K}_1$. Then the Frobenious norm is calculated as follows.

Definition 5.3. *Let $\boldsymbol{K}_1$ and $\boldsymbol{K}_2$ be two kernel matrices in $\mathbb{R}^{n\times n}$, then the kernel procrustes (KP) is given, in terms of the Frobenius norm by*

$$\begin{aligned} KP(\boldsymbol{K}_1, \boldsymbol{K}_2) = min_{\boldsymbol{Q}} &\quad \|\boldsymbol{K}_1 - \boldsymbol{Q}^T\boldsymbol{K}_2\boldsymbol{Q}\|_F^2 \\ s.t. &\quad \boldsymbol{Q}^T\boldsymbol{Q} = \boldsymbol{I}_n \end{aligned}$$

where $\boldsymbol{I}_n \in \mathbb{R}^{n\times n}$ is the diagonal matrix of ones.

The problem has a close solution given by $\mathbf{Q} = \mathbf{V}\mathbf{U}^T$, being $\mathbf{V}$ and $\mathbf{U}$ the corresponding matrices of eigenvectors of $\mathbf{K}_1$ and $\mathbf{K}_2$.

Next we detail a new disimilarity measure for kernel matrices based on the simultaneous diagonalization of $\mathbf{K}_1$ and $\mathbf{K}_2$.

5.2.1 Spectral framework for kernel matrices comparison

In this section we propose a new dissimilarity measure for kernel matrices. The key ingredient will be the generalised eigenvalues and eigevectos of pairs of kernel matrices ($\mathbf{K}_1$, $\mathbf{K}_2$). To define the new dissimilarity we start by introducing the concept of matrix pencil for general matrices $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n\times n}$.

Definition 5.4. *Given two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$, the matrix-valued function $L(\lambda) = \boldsymbol{A} - \lambda\boldsymbol{B}$ is called **matrix pencil**. The Pencil is represented through the pair $(\boldsymbol{A}, \boldsymbol{B})$.*

Definition 5.5. *A pencil $(\boldsymbol{A}, \boldsymbol{B})$ is called **definite pencil** if the matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ are symmetric and positive definite.*

Since we are interested in kernel matrices (that are always symmetric and semi-definite positive) then the main concern of this section will be positive definite pencil kernels of the form $(\mathbf{K}_1, \mathbf{K}_2)$. Next we introduce the concept of generalized eigenvalues of pencils.

Definition 5.6. *Let $\boldsymbol{A}$ and $\boldsymbol{B}$ be two matrices of dimensions $n \times n$. Then, the generalized eigenvalues of the pair $(\boldsymbol{A}, \boldsymbol{B})$ are the roots of the polynomial $det(\boldsymbol{A} - \lambda\boldsymbol{B}) = 0$.*

Notice that while the eigenvalues of a single matrix $\mathbf{A}$ are the roots of $det(\mathbf{A} - \lambda\mathbf{I}_n) = 0$ the generalized eigenvalues of the pair $(\mathbf{A}, \mathbf{B})$ are the roots of the polynomial $det(\mathbf{A} - \lambda\mathbf{B}) = 0$. Therefore, the generalized eigenvalues suffer from a lack of symmetry since the roots of $det(\mathbf{A} - \lambda\mathbf{B}) = 0$ are different from the roots of $det(\mathbf{A} - \lambda\mathbf{B}) = 0$.

Next we afford the task of diagonalizing two kernels in the same base of vectors. The following theorem ensures the existence of bases in which any pair of kernels $\mathbf{K}_1$ and $\mathbf{K}_2$ diagonalize.

Theorem 5.1. *(Parlett, 1997) Let be $(\mathbf{K}_1, \mathbf{K}_2)$ a definite pencil. Then, there is a nonsingular matrix* **V** *such that the matrices* $\mathbf{K}_1$ *and* $\mathbf{K}_2$ *can be simultaneously diagonalized.*

$$\begin{aligned} \boldsymbol{V}^T \boldsymbol{K}_1 \boldsymbol{V} &= \boldsymbol{\Lambda} \\ \boldsymbol{V}^T \boldsymbol{K}_2 \boldsymbol{V} &= \boldsymbol{\Sigma} \end{aligned}$$

being $\boldsymbol{\Lambda} = diag(\lambda_1, ..., \lambda_n)$, $\boldsymbol{\Sigma} = diag(\sigma_1, ..., \sigma_n)$ *and where the generalized eigenvalues* λ_i/σ_i *are real and finite.*

Then the diagonalized versions of $\mathbf{K}_1$ and $\mathbf{K}_2$ under $\mathbf{V}$ are $\boldsymbol{\Lambda}$ and $\boldsymbol{\Sigma}$ respectively. Notice that while the generalized eigenvalues of the pencil $(\mathbf{K}_1, \mathbf{K}_2)$ are λ_i/σ_i, the generalized eigenvalues of $(\mathbf{K}_2, \mathbf{K}_1)$ are σ_i/λ_i.

The base of vectors given by the columns of $\mathbf{V}$ is not necessarily orthonormal unless the kernels commute, thus if only if

$$\mathbf{K}_1\mathbf{K}_2 = \mathbf{K}_2\mathbf{K}_1.$$

In this case, the columns of $\mathbf{V}$ can be obtained by solving equations $\mathbf{K}_1\mathbf{v}_j = \lambda_j\mathbf{K}_2\mathbf{v}_j$ for $j = 1 \ldots, n$, that is, by calculating the eigenvectors of $\mathbf{K}_2^{-1}\mathbf{K}_1$ if $\mathbf{K}_2$ is not singular.

If the kernels do not commute, the matrix $\mathbf{V}$ is no uniquely defined. However, the values λ_i/σ_i are invariant under the choice of $\mathbf{V}$. In many real applications it is interesting to perform simultaneous diagonalization forcing $\boldsymbol{\Sigma} = \mathbf{I}_n$ (Epifanio et al., 2003). Several algorithms have been proposed for this case. See (Hua, 1991) for some examples. In our particular problem we will see how only the generalized eigenvalues of the pencil $(\mathbf{K}_1, \mathbf{K}_2)$ are needed to build kernel dissimilarities. Therefore we only need algorithms to compute $\boldsymbol{\Lambda} = diag(\lambda_1, ..., \lambda_n)$ and $\boldsymbol{\Sigma} = diag(\sigma_1, \ldots, \sigma_n)$.

In most cases, $\mathbf{K}_1$ and $\mathbf{K}_2$ are not full rank, and algorithms based on the QR decompositions are not stable (Golub and Loan, 1997). Let $r_1 = rank(\mathbf{K}_1)$ and $r_2 = rank(\mathbf{K}_2)$. In this work we use the *Direct Matrix Pencil Algorithm* (Hua, 1991) that uses the truncated Singular Value Decomposition (SVD) of the matrices to estimate their generalized eigenvalues. Let the SVD of $\mathbf{K}_1$ and $\mathbf{K}_2$ be

$$\begin{aligned} \mathbf{K}_1 = &\quad \mathbf{U}_1\boldsymbol{\Sigma}_1\mathbf{V}_1^T \\ \mathbf{K}_2 = &\quad \mathbf{U}_2\boldsymbol{\Sigma}_2\mathbf{V}_2^T, \end{aligned}$$

where $\boldsymbol{\Sigma}_1$ is a diagonal $r_1 \times r_1$ matrix and $\boldsymbol{\Sigma}_2$ is diagonal of dimensions $r_2 \times r_2$. $\mathbf{V}_1$ and $\mathbf{V}_2$ are the r_1 and r_2 left eigenvectors of $\mathbf{K}_1$ and $\mathbf{K}_2$, and $\mathbf{U}_1$ and $\mathbf{U}_2$ the corresponding right eigenvectors. Based on the above SVD decompostions, the pencil $(\mathbf{K}_1, \mathbf{K}_2)$ can be written as:

$$\mathbf{K}_1 - \lambda\mathbf{K}_2 = \mathbf{U}_1\boldsymbol{\Sigma}_1\mathbf{V}_1^T - \lambda\mathbf{U}_2\boldsymbol{\Sigma}_1\mathbf{V}_2^T. \tag{5.4}$$

Since the rank of $\mathbf{K}_2$ is r_2 then the pencil $(\mathbf{K}_1, \mathbf{K}_2)$ has r_2 generalizaed eigenvalues. If we multiply eq. (5.4) by $\mathbf{U}_2^T$ form the left and and by $\mathbf{V}_2^T$ form the right, we are not changing the eigenvalues and we obtain

$$\mathbf{U}_2^T\mathbf{U}_1\boldsymbol{\Sigma}_1\mathbf{V}_1^T\mathbf{V}_2 - \lambda\boldsymbol{\Sigma}_2, \tag{5.5}$$

what is a $n \times n$ matrix pencil. The generalized eigenvalues of this new pencil are easy to compute without stability problems. In addition, it can be proven that the generalized eigenvalues of the pencil in eq. (5.5) are equal to the eigenvalues of the matrices $\boldsymbol{\Sigma}_1^{-1}(\mathbf{U}_2^T\mathbf{U}_1\boldsymbol{\Sigma}_1\mathbf{V}_1^T\mathbf{V}_2)$ and $(\mathbf{U}_2^T\mathbf{U}_1\boldsymbol{\Sigma}_1\mathbf{V}_1^T\mathbf{V}_2)\boldsymbol{\Sigma}_2^{-1}$ that can be easily computed (Zolteoski, 1988) as well.

A Pencil Dissimilarity (PD) Measure for Matrices

Next we propose a new dissimilarity kernel measure that is based on the generalized eigenvalues of the pencil $(\mathbf{K}_1, \mathbf{K}_2)$. Since $\mathbf{K}_1$ and $\mathbf{K}_2$ are interchangeable, the measure should be invariant to the order. In our case, we fix $\Sigma = \mathbf{I}_n$, and therefore $\lambda_i/\sigma_i = \lambda_i$. It is clear that, if the order of the kernels in the pencil changes, the corresponding new eigenvalues become $1/\lambda_i$. For this reason, any measure based on the numbers $\lambda_1, \ldots, \lambda_r$ should be invariant under reciprocication of the eigenvalues. Then, the transformed eigenvalues to consider are

$$\lambda_i^* = \frac{1 + \lambda_i}{\sqrt{1 + \lambda_i^2}}, \tag{5.6}$$

where we use the notation $\lambda_i^* = (1/\lambda_i)^*$.

Once $\mathbf{K}_1$ and $\mathbf{K}_2$ are expressed in the same base, we can define a a kernel distance in terms of the generalized eigenvalues:

$$PD(\mathbf{K}_1, \mathbf{K}_2) = \sum_{i=1}^{r} \left(\lambda_i^* - 2/\sqrt{2}\right)^2, \tag{5.7}$$

where r is the number of different from zero generalized eigenvalues. This **Pencil Dissimilarity** is equivalent to $\|\Lambda - \mathbf{I}_n\|_F$ once the correction under reciprocication has been applied to the components of diagonal matrices Λ and $\mathbf{I}_n$. As we will show in the experimental section, this measure is consistent with the kernel alignment and also detects similar cluster structures in two kernel matrices.

Proposition 5.1. *Let $\boldsymbol{K}_1$ and $\boldsymbol{K}_2$ two kernel matrices in $\mathbb{R}^{n\times n}$. Then the $PD(\boldsymbol{K}_1, \boldsymbol{K}_2)$ defined in eq. (5.7) is a symmetric dissimilarity function.*

5.2.2 Other spectral measures

The use of the spectral (individual) decomposition of the original kernel matrices $\mathbf{K}_1$, $\mathbf{K}_2$ is also useful way to definite new similarity/disimilarity measures between the matrices. In the following sections we define some of them and we relate their spectral version with some well known existent measures.

Kernel Alignment

Consider the spectral decomposition of two kernel matrices $\mathbf{K}_1$, $\mathbf{K}_2$. Then

$$\begin{aligned}\mathbf{K}_1 &= \mathbf{U}\mathbf{D}_1\mathbf{U}^T = \mathbf{U}\mathbf{D}_1^{1/2}\mathbf{D}_1^{1/2}\mathbf{U}^T = \mathbf{U}^*(\mathbf{U}^*)^T\\ \mathbf{K}_2 &= \mathbf{V}\mathbf{D}_2\mathbf{V}^T = \mathbf{V}\mathbf{D}_2^{1/2}\mathbf{D}_2^{1/2}\mathbf{V}^T = \mathbf{V}^*(\mathbf{V}^*)^T.\end{aligned}$$

where $\mathbf{U}$ and $\mathbf{V}$ are the matrix whose columns are the eigenvectors of $\mathbf{K}_1$ and $\mathbf{K}_2$, and $\mathbf{D}_1$ and $\mathbf{D}_2$ the diagonal matrices with the eigenvalues in the diagonal. However we define $\mathbf{U}^* = \mathbf{U}\mathbf{D}_1^{1/2}$ and $\mathbf{V}^* = \mathbf{V}\mathbf{D}_2^{1/2}$.

Canonical Correlations (Hotelling, 1936) can be applied in order to estimate the degree of similarity of $\mathbf{U}^*$ and $\mathbf{V}^*$. This procedure calculates the angles between the spaces respectively generated by the columns of $\mathbf{U}^*$ and $\mathbf{V}^*$ by searching for maximal linear correlations over combinations of the variables. Unfortunately, if both kernels are full rank, the spanned spaces are the same and differences cannot be found. Nevertheless, the technique can be generalized calculating the sum of the cross correlations among the variables of the two basis. This can be done with the squared Frobenius norm of the matrix $(\mathbf{U}^*)^T\mathbf{V}^*$. Normalizing and rewriting in terms of the original decompositions we can define the following spectral similarity:

$$S_1(\mathbf{K}_1, \mathbf{K}_2) = \frac{\|\mathbf{D}_1^{1/2}\mathbf{U}^T\mathbf{V}\mathbf{D}_2^{1/2}\|_F^2}{\|\mathbf{D}_1\|_F\|\mathbf{D}_2\|_F} \tag{5.8}$$

Proposition 5.2. *The kernel alignment is equivalent to the S_1 measure.*

Kernel Procrustes

Let two kernels $\mathbf{K}_1$ and $\mathbf{K}_2$. Their diferences can also be obtained working direcly with the eigenvalues of the matrices. Let $\mathbf{D}_1$, $\mathbf{D}_2$ the diagonal matrices containing the eigenvalues of $\mathbf{K}_1$ and $\mathbf{K}_2$. Such eigenvalues represent the weight of the corresponding eigenvector in the spectral decompositions. Without taking the basis of eigenvectors into account, a disimilarity kernel measure can be defined as

$$S_2(\mathbf{K}_1, \mathbf{K}_2) = \|\mathbf{D}_1 - \mathbf{D}_2\|_F^2. \tag{5.9}$$

This measure is always positive and unbounded. It takes value zero when both kernels are equal.

Proposition 5.3. *The kernel similarity meassure S_2 is equivanlent to the kernel procrustes when $\boldsymbol{Q} = \boldsymbol{V}\boldsymbol{U}^T$.*

To end this section we include an example where the previously defined spectral measures are compared in a simulated example.

Example 5.1. In this example we compare, the behavior of three measured defined in this section: S_1 (or the Kernel Procrustes), S_2 (or the Kernel Alignment) and the new Pencil Dissimilarity (PD) that we will denote by S_3 in this example. The idea is to check the sensitivity of these measures when a they are used to evaluate the proximity between a kernel matrix $\mathbf{K}^*$ and some controlled perturbations of it.

We generate a sample s_n of $n = 100$ data made up from bivariate normal distribution $N(\boldsymbol{\mu}, \mathbf{I}_2)$ for $\boldsymbol{\mu} = (0, 0)$ and $\mathbf{I}_2$ the identity matrix. We fix $\mathbf{K}^*$ to be the kernel matrix of a Gaussian kernel $K(\mathbf{x}, \mathbf{y}) = exp(-\rho\|x - y\|^2)$ with $\sigma = 1$ evaluated on s_n. We modify $\mathbf{K}^*$ increasing and decreasing the parameters of the Gaussian kernel used to calculate the kernel matrix. In particular we consider 75 different values for ρ selected in a grid in the interval $[0.5, 3]$ obtaining $\mathbf{K}_1, \ldots, \mathbf{K}_{75}$. Finally we calculate the three proximities measures between $\mathbf{K}^*$ and each $\mathbf{K}_t$ for $t = 1, \ldots, 75$. To make results comparable we normalize the proximity measures as follows:

- $\widetilde{S}_1(\mathbf{K}^*, \mathbf{K}_t) = 1 - S_1(\mathbf{K}^*, \mathbf{K}_t)/\max_i\{S_1(\mathbf{K}^*, \mathbf{K}_i)\}$,
- $\widetilde{S}_2(\mathbf{K}^*, \mathbf{K}_t) = S_2(\mathbf{K}^*, \mathbf{K}_t)/\max_i\{S_2(\mathbf{K}^*, \mathbf{K}_i)\}$,
- $\widetilde{S}_3(\mathbf{K}^*, \mathbf{K}_t) = S_3(\mathbf{K}^*, \mathbf{K}_t)/\max_i\{S_3(\mathbf{K}^*, \mathbf{K}_i)\}$,

for $t = 1, \ldots, 75$. Notice that the measure S_1, that is a similarity, is transformed to dissimilarity.

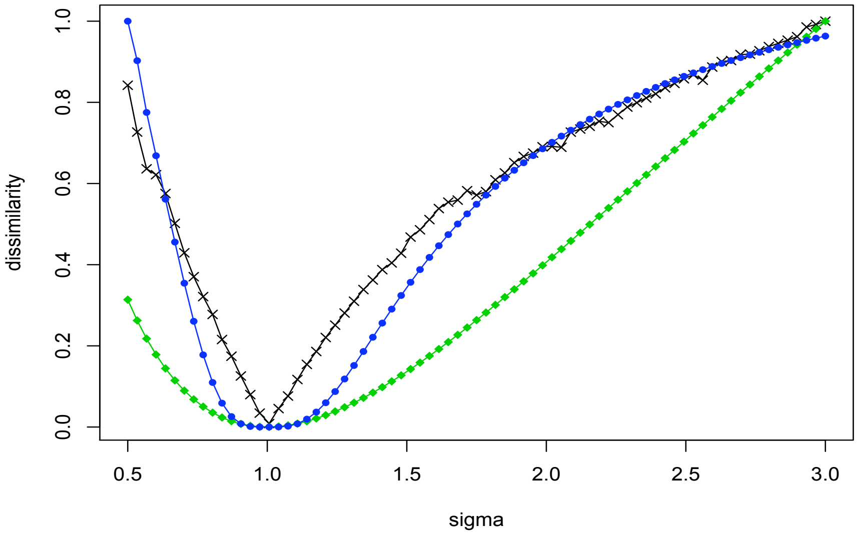


Figure 5.1: Comparison of the Kernel Alignment (S_1, green $\diamond$), the Pencil Dissimilarity (S_2, blue $\circ$) and the Kernel Procrustes (S_3, black x).

In Figure 5.1 we illustrate, for the three measures, the dissimilarity values between $\mathbf{K}^*$ and the set of matrices. The behavior is similar in the three cases. The dissimilarity decreases fast for values of ρ lower that 1 and grows slowly otherwise. It is remarkable that S_1 and S_2 describe a flat curve around 1 while for S_3 the the behavior of described curve is sharp. I addition, this measure is more irregular that S_1 and S_2.

• • •

5.3 Proximities combinations based on Joint Diagonalization Algorithms

In this Section we propose a new methodology to remove redundant information when several kernel matrices $\mathbf{K}_1, \ldots, \mathbf{K}_m$ are combined. The key idea is to diagonalize a set of kernels simultaneously obtaining a basis$\{\mathbf{v}_1^*, \ldots, \mathbf{v}_{n^*}^*\}$ of a general space of higher dimension that only includes the non redundant information among the original kernels.

As we studied in Section 5.2, the exact simultaneous diagonalization of two matrices is always possible (Parlett, 1997). The base of vectors given by the columns of $\mathbf{V}$ in Theorem 5.1 is not necessarily orthonormal unless the matrices $\mathbf{A}$ and $\mathbf{B}$ conmute, that is when $\mathbf{AB} = \mathbf{BA}$. In this case $\mathbf{V}$ is also unique. If $\mathbf{B}$ is non-singular, the problem can be solved as and ordinary eigenvalue problem where the target matrix is $\mathbf{B}^{-1}\mathbf{A}$. See (Hua, 1991; Epifanio et al., 2003) and references therein for further details. However, when more than two matrices are involved in the diagonalization process, if they do not commute, the diagonalization have to be approximated (Cardoso and Souloumiac, 1996).

In this Section we review the Approximate Joint Diagonalization algorithm of a set of matrices and we introduce a new procedure for kernel fusion based on it.

5.3.1 Approximate Joint Diagonalization of Matrices

Given a set of matrices $\{\mathbf{A}_1, ..., \mathbf{A}_m\}$ for $\mathbf{A}_t \in \mathbb{R}^{n\times n}$ it is not possible in general to achieve perfect joint diagonalization in a single step, unless $\mathbf{A}_i\mathbf{A}_j = \mathbf{A}_j\mathbf{A}_i$. Unfortunately these restrictions do not hold for most theoretical or practical problems. Therefore in practice we will have to find an orthonormal change of basis which makes the matrices "as diagonal as possible" in a sense that will be detailed right away. Some fields of application for these idea are, for instance, Blind Source Separation (Yeredor, 2002) and Independent Component Analysis (Bach and Jordan, 2002).

In this section we make use of the the Approximate Joint Diagonalization (AJD) of symmetric matrices (Wax and Sheinvald, 1997; Cardoso and Souloumiac, 1996; Yeredor, 2002). Given a square matrix $\mathbf{A}$ the notion of closeness to be diagonal can be defined in several ways. Here measure the deviation of A from diagonality by defining

$$off(\mathbf{A}) = \|\mathbf{A} - diag(\mathbf{A})\|_F^2 = \sum_{i\neq j} a_{ij}^2,$$

where $\|\mathbf{A}\|_F = \sum_i \sum_j a_{ij}^2$ is the Frobenius norm. If $\mathbf{A}$ is a diagonal matrix then $off(\mathbf{A}) = 0$, while $off(\mathbf{A})$ will take small positive values when the off-diagonal values of $\mathbf{A}$ are close to zero.

Given the set of matrices $\{\mathbf{A}_1, \ldots, \mathbf{A}_m\}$, the target is to find an orthonormal matrix $\mathbf{V}$ such that the departure from diagonality of the transformed matrices $\tilde{\mathbf{D}}_t = \mathbf{V}^T\mathbf{A}_t\mathbf{V}$ are as diagonal as possible $\forall i \in \{1, ..., m\}$. Therefore the goal will be to minimize

$$\begin{array}{rcl} J(\mathbf{V}) & = & \sum_{t=1}^m off(\mathbf{V}^T\mathbf{A}_t\mathbf{V}) \\ s.t. & & \\ & & \|\mathbf{V}^T\mathbf{V} - \mathbf{I}_n\|_F = 0 \\ & & \|diag(\mathbf{V} - \mathbf{I}_n)\|_F = 0, \end{array} \tag{5.10}$$

where the restrictions have to be included to achieve orthonormality and to avoid the trivial solution $\mathbf{V} = 0$. After solving eq. (5.10) we will obtain a set quasi diagonal matrices $\tilde{\mathbf{D}}_1, \ldots, \tilde{\mathbf{D}}_m$, where $\tilde{\mathbf{D}}_t = \mathbf{V}^T \mathbf{A}_t \mathbf{V}$ for $i = 1, ..., m$.

There is no closed solution for the problem in eq. (5.10) and some type of numerical approach has to be adopted. We will apply the algorithm described in (Cardoso and Souloumiac, 1996; Yeredor, 2002). The idea is to generate a sequence of similarity transformations of the initial matrices that drive to zero the off-diagonal entries. The convergence of the algorithm is proven to be quadratic and the obtained eigenvalues and eigenvectors are robust against small perturbations of the data.

5.3.2 Fusion Joint Diagonalization Algorithm (FJDA)

As already mentioned, Approximate Joint Diagonalization involves the computation of a base of orthogonal vectors in which the set of kernels approximately diagonalize. We will obtain relevant information about the data structure by analyzing the resulting eigenvalues, or equivalently, the diagonal matrices obtained from the joint diagonalization procedure. The ideas are similar to that used in Principal Components Analysis, where the covariance matrix is diagonalized and the resulting eigenvalues can be interpreted as the weights of the new variables.

Let $\{\mathbf{v}_1, ..., \mathbf{v}_n\}$ be the column vectors of the matrix $\mathbf{V}$ obtained from the JD algorithm (the $\{\mathbf{v}_i^*\}$ vectors in the introduction of this chapter). These vectors constitute the basis where the set of kernels diagonalize and can be interpreted as the *average eigenspace* of the kernels. A detailed analysis of the kernels redundancy can be done in terms of the values of the obtained quasi-diagonal matrices $\tilde{\mathbf{D}}_1, \tilde{\mathbf{D}}_2, \ldots, \tilde{\mathbf{D}}_m$ obtained. Given the kernel matrix $\mathbf{K}_t$, the components of the associated $\tilde{\mathbf{D}}_t$ can be interpreted as follows:

- $\tilde{\mathbf{D}}_t(i,i) = 0$: the vector $\mathbf{v}_i$ is irrelevant for the kernel $\mathbf{K}_t$. That is, the i-th variable v_i is in the null space of $\mathbf{K}_t$.
- $\tilde{\mathbf{D}}_t(i,i) \neq 0$: in this case $\mathbf{v}_i$ is a relevant component for $\mathbf{K}_t$.
- $\tilde{\mathbf{D}}_t(i,j)$: These values can be interpreted as the interactions among the new variables. Due to the JD operation, $\tilde{\mathbf{D}}_t(i,j) \approx 0$.

Given $\mathbf{V}$ and $\tilde{\mathbf{D}}_1, \tilde{\mathbf{D}}_2, \ldots, \tilde{\mathbf{D}}_m$, the straightforward sum of the kernel matrices can be reexpressed as:

$$\sum_{t=1}^{m} \mathbf{K}_t = \mathbf{V}^T \left(\sum_{t=1}^{m} \tilde{\mathbf{D}}_t \right) \mathbf{V} \tag{5.11}$$

Table 5.1: Scheme of the Fusion Joint Diagonalization Algorithm.

INPUT:	Kernel matrices $\mathbf{K}_1, \ldots, \mathbf{K}_m$	
OUTPUT:	Kernel combination $\mathbf{K}^*$	
STEP 1	$(\mathbf{V}, \tilde{\mathbf{D}}_1, \tilde{\mathbf{D}}_2, \ldots, \tilde{\mathbf{D}}_m)$	= $AJD(\mathbf{K}_1, \ldots, \mathbf{K}_m)$
STEP 2	$\mathbf{D}^*$	= $F(\tilde{\mathbf{D}}_1, \tilde{\mathbf{D}}_2, \ldots, \tilde{\mathbf{D}}_m)$
STEP 3	$\mathbf{K}^*$	= $\mathbf{V}^T\mathbf{D}^*\mathbf{V}$

Given that the *off-diagonal values* of $\tilde{\mathbf{D}}_1, \tilde{\mathbf{D}}_2, \ldots, \tilde{\mathbf{D}}_m$ are quite close to zero, $\tilde{\mathbf{D}}_t(i,i)$ can be interpreted as the weight that kernel $\mathbf{K}_t$ assigns to the i-th variable in the new basis. Since the new base is orthogonal, independent information is given by each component. The straightforward sum of kernels implies to include redundances in the operation and to overweight variables that appear in more than one kernel at the same time. In order to avoid these redundances, the sum of the quasi-diagonal matrices of expression eq. (5.11) can be replaced by the function $F(\tilde{\mathbf{D}}_1, \tilde{\mathbf{D}}_2, \ldots, \tilde{\mathbf{D}}_m)$ defined as follows:

$$F(\tilde{\mathbf{D}}_1, \tilde{\mathbf{D}}_2, \ldots, \tilde{\mathbf{D}}_m) = \begin{cases} \max\left\{\tilde{\mathbf{D}}_1(i,j), ..., \tilde{\mathbf{D}}_m(i,j)\right\} & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \tag{5.12}$$

The justification of this choice is as follows. The relevance of the $i-th$ variable in the basis induced by kernel $\mathbf{K}_t$ is given by $\tilde{\mathbf{D}}_t(i,i)$. The use of the max function guarantees that the i-th variable will be relevant in the resulting combined basis if this is the case for any of the individual representations. Thus, the weight of ith variable in the fusion kernel will be $max\{\tilde{\mathbf{D}}_1(i,i), \ldots, \tilde{\mathbf{D}}_m(i,i)\}$.

The final algorithm for kernel fusion is shown in Table 5.1 and it provides a global framework for kernel fusion. Notice that, since the matrix $\mathbf{V}$ is orthogonal and the diagonal matrices of $F(\tilde{\mathbf{D}}_1, \tilde{\mathbf{D}}_2, \ldots, \tilde{\mathbf{D}}_m)$ are positive, $\mathbf{K}^*$ is always demidefinte postive and therefore is a Mercer kernel matrix.

To conclude this section we include two examples where the behavior of Fusion Joint Diagonalization Algorithm is illustrated.

Example 5.2. In order to validate the utility of the approximate joint diagonalization algorithm to detect possible redundancies among kernels, we perform the following experiment. Consider the following data matrix $\mathbf{X}$ with 5 random observations and three orthogonal variables $\mathbf{x}_1$, $\mathbf{x}_2$ and $\mathbf{x}_3$:

$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3] = \begin{bmatrix} -0.2398 & -0.4738 & 0.2370 \\ 0.4249 & 0.3203 & 0.1753 \\ -0.3535 & 0.4904 & -0.0183 \\ -0.3284 & -0.1104 & -0.2079 \\ 0.4969 & -0.2266 & -0.1860 \end{bmatrix} \tag{5.13}$$

Let $\mathbf{K}_1$, $\mathbf{K}_2$ and $\mathbf{K}_3$ be three linear kernels calculated using the variable sets $\{\mathbf{x}_1\}$, $\{\mathbf{x}_1, \mathbf{x}_2\}$ and $\{\mathbf{x}_2, \mathbf{x}_3\}$ respectively. Notice that, in this example, $\mathbf{K}_1$ and $\mathbf{K}_3$ are based on independent variable sets while $\mathbf{K}_2$ and $\mathbf{K}_3$ share the variable $\mathbf{x}_2$. In addition let the linear kernel $\mathbf{K} = \mathbf{X}\mathbf{X}^T$ calculated with $\{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$ be:

$$\mathbf{K} = \begin{bmatrix} 0.3381 & -0.2121 & -0.1519 & 0.0818 & -0.0559 \\ -0.2121 & 0.3138 & 0.0037 & -0.2113 & 0.1059 \\ -0.1519 & 0.0037 & 0.3658 & 0.0658 & -0.2834 \\ 0.0818 & -0.2113 & 0.0658 & 0.1633 & -0.0995 \\ -0.0559 & 0.1059 & -0.2834 & -0.0995 & 0.3328 \end{bmatrix}.$$

The goal of this experiment is to show the utility of the joint diagonalization algorithm to detect the redundances in the battery of kernels $\{\mathbf{K}_1, \mathbf{K}_2, \mathbf{K}_3\}$. To this aim, we apply the joint diagonalization to $\mathbf{K}_1$, $\mathbf{K}_2$ and $\mathbf{K}_3$. We obtain three (in this case exact) diagonal matrices $\tilde{\mathbf{D}}_1$, $\tilde{\mathbf{D}}_2$ and $\tilde{\mathbf{D}}_2$ given by

$$\tilde{\mathbf{D}}_1 = diag(0.7178, 0.0000, 0.0000, 0.0000, 0.0000)$$

$$\tilde{\mathbf{D}}_2 = diag(0.7178, 0.6310, 0.0000, 0.0000, 0.0000)$$

$$\tilde{\mathbf{D}}_3 = diag(0.0000, 0.6310, 0.1651, 0.0000, 0.0000)$$

and a orthogonal matrix $\mathbf{V}$ whose columns are the common base of generalized eigenvectors $\mathbf{V} = \{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \mathbf{v}_4, \mathbf{v}_5\}$ given by:

$$\mathbf{V} = \begin{bmatrix} -0.6173 & 0.5964 & -0.4032 & 0.1389 & 0.2852 \\ -0.0451 & 0.5832 & 0.4315 & -0.5118 & -0.4579 \\ 0.4173 & 0.2830 & -0.5015 & 0.3877 & -0.5865 \\ -0.2938 & 0.0203 & 0.5800 & 0.7289 & -0.2133 \\ 0.5970 & 0.4729 & 0.2516 & 0.1927 & 0.5652 \end{bmatrix}$$

Notice that the number of non null eigenvalues of the matrices $\tilde{\mathbf{D}}_1$, $\tilde{\mathbf{D}}_2$ and $\tilde{\mathbf{D}}_2$ equals the number of variables used to define the corresponding kernel matrices. For instance $\tilde{\mathbf{D}}_1$ only weights with 0.7178 the variable $\mathbf{v}_1$ while $\tilde{\mathbf{D}}_1$ also weights $\mathbf{v}_2$ with 0.6310.

Let $\mathbf{K}^*$ be the kernel sum defined by $\mathbf{K}^* = \mathbf{K}_1 + \mathbf{K}_2 + \mathbf{K}_3$ or equivalently,

$$\mathbf{K}^* = \mathbf{V}^T(\tilde{\mathbf{D}}_1 + \tilde{\mathbf{D}}_2 + \tilde{\mathbf{D}}_2)\mathbf{V} = \begin{bmatrix} 0.6201 & -0.4657 & -0.2995 & 0.2128 & -0.0677 \\ -0.4657 & 0.5969 & 0.0105 & -0.3862 & 0.2445 \\ -0.2995 & 0.0105 & 0.7313 & 0.1278 & -0.5701 \\ 0.2128 & -0.3862 & 0.1278 & 0.2833 & -0.2377 \\ -0.0677 & 0.2445 & -0.5701 & -0.2377 & 0.6311 \end{bmatrix}$$

A desirable property of this sum would be that $\mathbf{K}^*$ matches the kernel $\mathbf{K}$ that has been calculated using the independent information of $\mathbf{x}_1$, $\mathbf{x}_2$ and $\mathbf{x}_3$. However $\mathbf{K}^*$ and $\mathbf{K}$ are different. This happen because variables $\mathbf{v}_1$ and $\mathbf{v}_2$ are twice its weight (the sum doubles the eigenvalue) in the direct sum of $\mathbf{K}_1$, $\mathbf{K}_2$ and $\mathbf{K}_2$. Consider now $\mathbf{K}^+$ the sum of $\mathbf{K}_1$, $\mathbf{K}_2$, and $\mathbf{K}_3$ calculated with the Fusion Joint Diagonalization Algorithm in Table 5.1. In this particular case we obtain that

$$F(\tilde{\mathbf{D}}_1, \tilde{\mathbf{D}}_2, \tilde{\mathbf{D}}_3) = diag(0.7178, 0.6310, 0.1651, 0.0000, 0.0000)$$

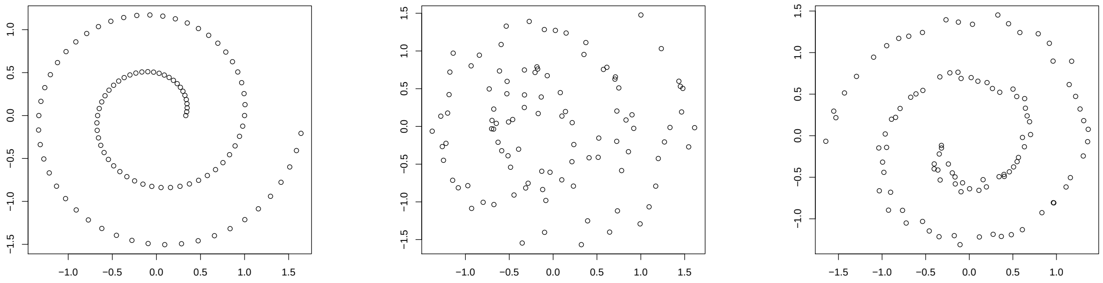
and therefore

$$\mathbf{K}^+ = \mathbf{V}^T(F(\tilde{\mathbf{D}}_1, \tilde{\mathbf{D}}_2, \tilde{\mathbf{D}}_3))\mathbf{V} = \begin{bmatrix} 0.3381 & -0.2121 & -0.1519 & 0.0818 & -0.0559 \\ -0.2121 & 0.3138 & 0.0037 & -0.2113 & 0.1059 \\ -0.1519 & 0.0037 & 0.3658 & 0.0658 & -0.2834 \\ 0.0818 & -0.2113 & 0.0658 & 0.1633 & -0.0995 \\ -0.0559 & 0.1059 & -0.2834 & -0.0995 & 0.3328 \end{bmatrix}.$$

Notice that $\mathbf{K}^+ = \mathbf{K}$ what shows that the proposed methodology is able to remove the redundant information (of variables $\mathbf{x}_1$ and $\mathbf{x}_2$) that however was included in the computation of $\mathbf{K}^*$.

• • •

Example 5.3. In this example we illustrate the performance of the new FJDA in a data structure recovery task. We consider two different one-dimensional random projections π_1 and π_2 of the spiral data in Figure 5.2 a) and calculate the kernel matrices $\mathbf{K}_1$ and $\mathbf{K}_2$ by applying the linear kernel $K(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T\mathbf{y}$ to the projected data points, that is, $K_i(\mathbf{x}, \mathbf{y}) = \pi_i(\mathbf{x})^T\pi_i(\mathbf{y})$. We add a corrupted (random) representation of the data and calculate $\mathbf{K}_3$ from this representation in the same way. $\mathbf{K}_3$ plays the role of a non informative (non-related) piece of information in the system. This situation happens when the distance function is not appropiate for the data set under consideration or when we try to use irrelevant information to solve a problem. The task is to recovery the original



(a) Spiral data.

(b) Direct sum of kernels for the Spiral data.

(c) FJDA applied to the kernels for the Spiral data.

Figure 5.2: Original data and representations for the recovered data after the direct combination of three kernels and after the Fusion Joint Diagonalization Algorithm (FJDA)

data set from the three projections.

Two fusion schemes were compared in the experiment: The straightforward sum of kernels $\mathbf{K}_{sum} = \mathbf{K}_1 + \mathbf{K}_2 + \mathbf{K}_3$ and the combination $\mathbf{K}^*$ calculated with the Fusion Joint Diagonalization Algorithm. In Figure 5.2 b) and) the results are shown. It is clear that our procedure is able to recover the original data set structure while the straightforward sum of kernels fails on the task of recovering the data set structure.

5.4 Conclusions and final remarks

In this Chapter we have proposed an spectral framework for the analysis of redundancies in proximity matrices combinations. Two main issues have been studied: the problem of defining metrics for matrix spaces and the problem of redundant information in kernel combinations

In Section 5.2.1 we have proposed an spectral framework for the definition of metrics for matrix spaces. In particular we have proposed a new Pencil Dissimilarity (PD) based on the simultaneous diagonalization of kernel matrices. The new measure is easy to calculate and is proven to be consistent with the Kernel Alignment and the Kernel procrustes in a simulated experiment.

In addition, the proposed spectral framework can be used in information fusion when the sources of information are given by a set of kernel matrices. We have proposed in Section 6.7 and algorithm that, based on the Approximate Joint Diagonalization of matrices, produces a new representation of the data set in a Euclidean space where the basis is created from the representations induced by the individual kernels. The behavior of the fusion scheme is illustrated in two simulated examples and it will be tested in the experimental section of this thesis.

5.5 APPENDIX: Proofs

Proposition 5.1. A symmetric dissimilarity function for matrices is a application $d : \mathbb{R}^{n\times n} \times \mathbb{R}^{n\times n} \rightarrow \mathbb{R}$ such that given any $\mathbf{K}_1, \mathbf{K}_2 \in \mathbb{R}^{n\times n}$ then:

(i) $d(\mathbf{K}_1, \mathbf{K}_1) = 0$.

(ii) $d(\mathbf{K}_1, \mathbf{K}_2) \geq 0$ (non-negativity).

(iii) $d(\mathbf{K}_1, \mathbf{K}_2) = d(\mathbf{K}_1, \mathbf{K}_2)$ (symmetry).

It is straightforward that the function PD defined in eq. (5.7) satisfies i) and ii). In addition, if the generalized eigenvalues of the pencil $(\mathbf{K}_1, \mathbf{K}_2)$ are $\lambda_1, \ldots, \lambda_r$, those of the pencil $(\mathbf{K}_2, \mathbf{K}_1)$ are $1/\lambda_1, \ldots, 1/\lambda_r$. Since $\lambda_i^* = (1/\lambda_i)^*$ then

$$\sum_{i=1}^{r} \left(\lambda_i^* - 2/\sqrt{2}\right)^2 = \sum_{i=1}^{r} \left(2/\sqrt{2} - (1/\lambda_i)^*\right)^2 \tag{5.7}$$

Therefore $PD(\mathbf{K}_1, \mathbf{K}_2) = PD(\mathbf{K}_2, \mathbf{K}_1)$ and iii) is also satisfies, what concludes the proof. □

Proposition 5.2. Let $\mathbf{K}_1 = \mathbf{U}\mathbf{D}_1\mathbf{U}^T$ and $\mathbf{K}_2 = \mathbf{V}\mathbf{D}_2\mathbf{V}^T$ be the diagonalization of the two kernel matrices K_1 and K_2 such that $\mathbf{K}_1 = \sum_i \lambda_i \mathbf{u}_i \mathbf{u}_i^T$ and $\mathbf{K}_2 = \sum_j \mu_j \mathbf{v}_i \mathbf{v}_j^T$. Then:

$$\begin{aligned} S_1(\mathbf{K}_1, \mathbf{K}_2) &= \frac{\|\mathbf{D}_1^{1/2}\mathbf{U}^T\mathbf{V}\mathbf{D}_2^{1/2}\|_F^2}{\|\mathbf{D}_1\|_F \|\mathbf{D}_2\|_F} = \frac{\sum_{ij} \lambda_i \mu_j \left\langle \mathbf{u}_i, \mathbf{v}_j \right\rangle^2}{\sqrt{\sum_i \lambda_i^2}\sqrt{\sum_j \mu_j^2}} \\ &= \frac{\sum_{ij} \lambda_i \mu_j \langle \mathbf{u}_i, \mathbf{v}_j \rangle^2}{\sqrt{\sum_i \lambda_i \lambda_j \langle \mathbf{u}_i, \mathbf{u}_j \rangle}\sqrt{\sum_j \mu_i \mu_j \langle \mathbf{v}_i, \mathbf{v}_j \rangle}} \\ &= \frac{\langle \sum_i \lambda_i \mathbf{u}_i \mathbf{u}_i^T, \sum_j \mu_j \mathbf{v}_j \mathbf{v}_j^T \rangle}{\sqrt{\sum_i \lambda_i \lambda_j \langle \mathbf{u}_i \mathbf{u}_i^T, \mathbf{u}_j \mathbf{u}_j^T \rangle}\sqrt{\sum_j \mu_i \mu_j \langle \mathbf{v}_i \mathbf{v}_i^T, \mathbf{v}_j \mathbf{v}_j^T \rangle}} \qquad (5.8) \\ &= \frac{\langle \mathbf{K}_1, \mathbf{K}_2 \rangle_F}{\sqrt{\langle \mathbf{K}_1, \mathbf{K}_1 \rangle_F \langle \mathbf{K}_2, \mathbf{K}_2 \rangle_F}} \\ &= A(\mathbf{K}_1, \mathbf{K}_2). \end{aligned}$$

□

Proposition 5.3. Let $\mathbf{K}_1 = \mathbf{U}\mathbf{D}_1\mathbf{U}^T$ and $\mathbf{K}_2 = \mathbf{V}\mathbf{D}_2\mathbf{V}^T$ be the diagonalization of the two kernel matrices $\mathbf{K}_1$ and $\mathbf{K}_2$. Then:

$$\begin{aligned} S_2(\mathbf{K}_1, \mathbf{K}_2) &= \|\mathbf{D}_1 - \mathbf{D}_2\|_F^2 \\ &= \|\mathbf{U}(\mathbf{D}_1 - \mathbf{D}_2)\mathbf{U}^T\| \\ &= \|\mathbf{U}\mathbf{D}_1\mathbf{U}^T - \mathbf{U}\mathbf{D}_2\mathbf{U}^T\|_F^2 \\ &= \|\mathbf{K}_1 - \mathbf{U}\mathbf{D}_2\mathbf{U}^T\|_F^2 \\ &= \|\mathbf{K}_1 - \mathbf{U}\mathbf{V}^T\mathbf{V}\mathbf{D}_2\mathbf{V}^T\mathbf{V}\mathbf{U}^T\|_F^2 \\ &= \|\mathbf{K}_1 - \mathbf{U}\mathbf{V}^T\mathbf{K}_2\mathbf{V}\mathbf{U}^T\|_F^2 \\ &= \|\mathbf{K}_1 - \mathbf{Q}^T\mathbf{K}_2\mathbf{Q}\|_F^2 \\ &= KP(\mathbf{K}_1, \mathbf{K}_2), \end{aligned} \tag{5.9}$$

for $\mathbf{Q} = \mathbf{U}\mathbf{V}^T$. □

Chapter 6

Experiments

Abstract

This chapter in devoted to experimentation. We empirically test the procedures developed in Chapters 3, 4 and 5 and we compare them with the state of the art in data representation, cluster and classification techniques.

6.1 Introduction and methodology

In this chapter we present a wide range of experimental results devoted to evaluate the performance of the different methodologies developed in Chapters 3, 4 and 5. We will work with several simulated and real data sets where an adequate representation of the raw data is essential to obtain accurate and competitive results.

Whereas some cluster examples are analyzed, most of the experiments of this chapter are oriented to compare the performance of data representations systems in classification problems. To this aim, we will use four classification algorithms to study the effective accuracy each proposed data representation independently of the particular classification method employed. Thus, even when our techniques improve upon existing results, we are more interested in the comparative analysis between the data representations procedures than in giving an optimal results for each data set. Next we introduce the four classification algorithms we will consider:

- SVM, the Support Vector Machines described in the Section 2.6 of this thesis. In all the experiments we use a linear kernel and we fix the regularization parameter $C = 100$.

- FDA, the Flexible Discriminant Analysis proposed in (Hastie et al., 1994). We use two variants: FDA/BRUTO which is based on Additive Models and spline smoothing parameters and FDA/MARS which make use of the Multivariate Adaptative Regression Splines (Friedman, 1991).
- PLS/LDA, classification method described in (Boulesteix, 2004) which consists in Partial Least Squares dimension reduction and Linear Discriminant Analysis applied on the PLS components.

In all the simulations we divide the data in a set for training and a set for testing. In addition, the final errors of each procedure are always obtained as the averaged errors of a set of Monte Carlo simulations.

This chapter is organized as follows. In Section 6.2 we analyze several classification and cluster examples in the field of Functional Data Analysis. We also include an example to combine some of the ideas developed in chapters 3 and 4. In Section 6.3 we deal with two cluster examples of functional data. In section 6.4 we analyze a set of partially labeled classification problems. In 6.5 we show the examples of classification problems where the similarity between the data is asymmetric. In Section 6.6 some simulation results regarding the accuracy of the methodology proposed in Section 4.3.3 are included. In Section 6.7 we test the Pencil dissimilarity developed in Chapter 5 in a cluster example. We conclude this chapter in Section 6.8 where the Fusion Joint Diagonalization algorithm is tested in a real example

6.2 Classification experiements in FDA

In this section we show the perfomance of the methodology developed in in Chapter 3 in several classification and cluster problems where the nature of the data is functional. We start with a simulated example.

6.2.1 Waveform data

In this experiment we analyze a modified version of the three class waveform data (Breiman et al., 1984). We consider 400 predictors for each curve, instead the 21 of the original case. The three classes of the problem are defined by:

$$x(t) = uh_1(t) + (1-u)h_2(t) + \varepsilon(t) \text{ for class 1;}$$

$$x(t) = uh_1(t) + (1-u)h_3(t) + \varepsilon(t) \text{ for class 2;}$$

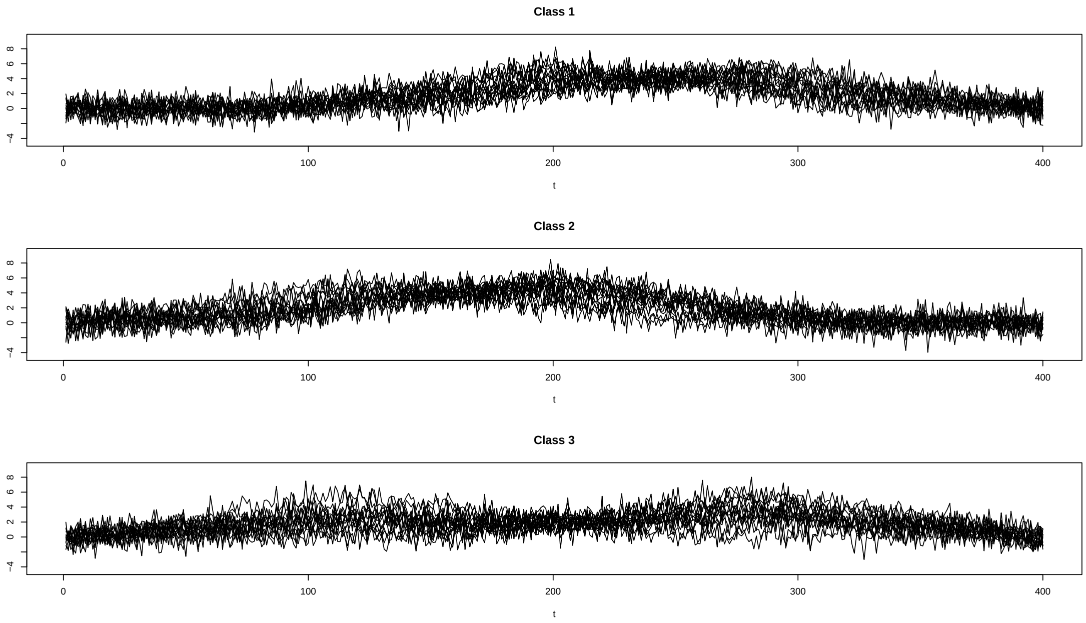


Figure 6.1: Three classes of the waveform data set.

$$x(t) = uh_2(t) + (1-u)h_3(t) + \varepsilon(t) \text{ for class 3;}$$

for u a uniform random variable on $(0, 1)$, $\varepsilon(t)$ standard normal variables, and the h_k function the shifted waveforms for $t \in [1, 21]$:

$$h_1(t) = max(6 - |t - 11|, 0), h_2(t) = h_1(t-4) \text{ and } h_3(t) = h_1(t+4).$$

We generate 1200 observations of the model (400 of each class), and we consider 450 for training the models and 750 as test sample. A plot of the three classes of the problem is shown in Figure 6.1. The objective of this example is to illustrate that an effective foregoing reduction of the dimension of the curves (projecting them onto certain RKHSs via eq. (3.2)) improves the classification errors of a variety of classification algorithms (those described in Section 6.1) compared to the same algorithms trained over the raw data.

We consider five different RHKSs where project the data. First, we use two basis of functions, both of dimension 10, to construct two kernel functions via eq. (3.50). The fist one is made up of a P-splines basis while for the second one a basis of B-splines is used. We also consider the data covariance function and two generalized covariance functions: a Gaussian kernel given by $K(\mathbf{x}, \mathbf{y}) = \exp\{-\rho\|\mathbf{x} - \mathbf{y}\|^2\}$ and a Laplace kernel

Table 6.1: Errors of the four classifications algorithms and 5 curves representations (+ the raw data) in the Waveform data after 100 runs. In italic letters the best technique of each row is remarked. In bold letters the best result is shown. In parenthesis the standard errors of the averaged errors are also shown.

Method	B-Splines	P- Splines	Cov.	RBF	Laplace	Raw data
SVM	0.0491 (0.0011)	0.0485 (0.0013)	0.0519 (0.0012)	0.0470 (0.0012)	*0.0467* *(0.0012)*	0.0628 (0.0014)
FDA_{bruto}	0.0293 (0.0010)	0.0353 (0.0010)	0.0313 (0.0009)	0.0289 (0.0010)	**0.0288** **(0.0010)**	0.0839 (0.0017)
FDA_{mars}	0.0399 (0.0014)	0.0362 (0.0013)	0.0413 (0.0014)	0.0449 (0.0014)	*0.0395* *(0.0013)*	0.1091 (0.0020)
LDA/PLS	0.0610 (0.0017)	0.0665 (0.0018)	*0.0606* *(0.0019)*	0.0612 (0.0018)	0.0613 (0.0018)	0.1675 (0.0026)

$K(\mathbf{x}, \mathbf{y}) = \exp\{-\rho\|\mathbf{x} - \mathbf{y}\|\}$ where, in both cases, $\rho = 1$. We project the data onto the RKHSs induced by the previous kernels using eq. (3.2) for $\gamma = 10^{-3}$.

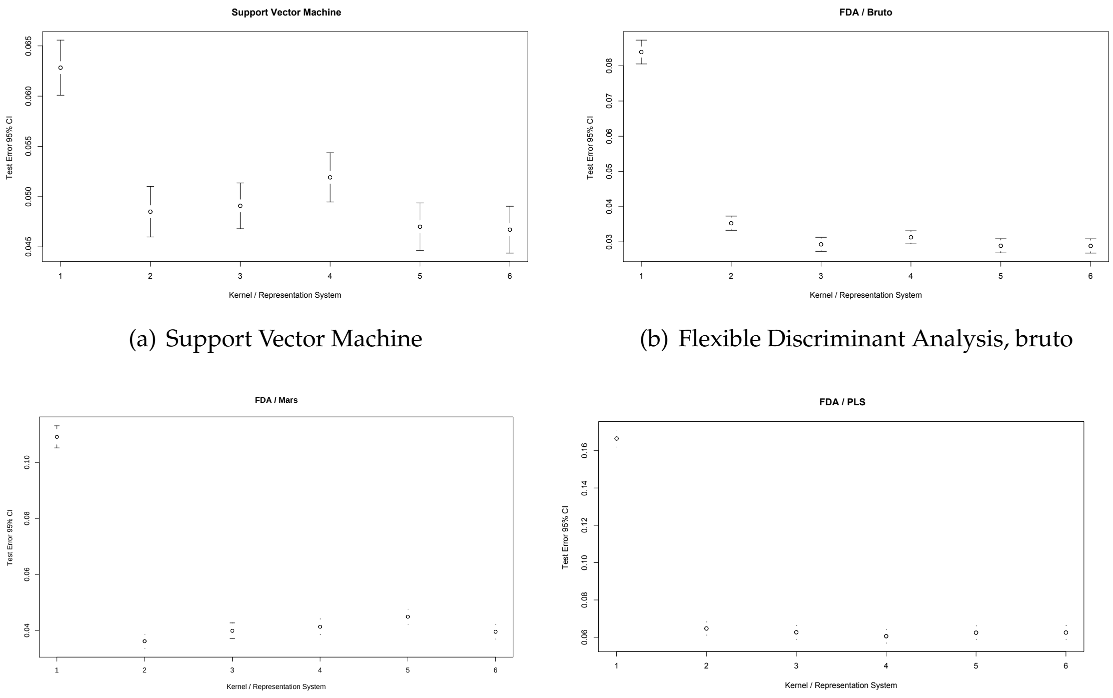
We train the SVM, FDA_{bruto}, FDA_{mars} and LDA/PLS using the five estimated projections and also using the raw data. In Table 6.1 we show the final averaged errors after 100 runs of the experiment. Regarding the projections, we decide the number of components to retain by cross validations over the errors. This means that the errors in Table 6.1 are selected as the best classification result when the only first d eigenfunctions of the proposed kernels for $d = 1, \ldots, 10$ are taken into account to represent the curves.

Results are shown in Table 6.1. It is clear that reducing the dimension of the curves by projecting them onto the proposed RKHSs always reduces significantly the classification errors of the four techniques compared to the same algorithms trained over the raw data. To illustrate better these differences we include Figure 6.2 where the confidence intervals of the errors for each representation are shown. The best algorithm-projection combination is a Laplace kernel with the FDA_{bruto} algorithm. It is also remarkable the good performance of the SVM trained over the raw data compared with the rest of the techniques.

6.2.2 Real classification examples

In this section we analyze the following three real functional data sets:

- *Growth data*: This data set consists on 93 growth curves for a sample of 54 boys and 39 girls (Ramsay and Silverman, 2006) (see Figure 6.3). The observations were



(a) Support Vector Machine

(b) Flexible Discriminant Analysis, bruto

(c) Flexible Discriminant Analysis, mars

(d) LDA/PLS

Figure 6.2: Confidence Intervals (95%) for the errors of the 5 representation (+ the raw data) in four classification techniques. The representation systems are: (1) Raw data, (2) B-splines, (3) P-splines, (4) Data covariance, (5) Gaussian Kernel and (6) Laplace Kernel.

measured at a set of twenty nine ages (from one to eighteen years old). The data were originally smoothed by using a spline basis and are available in the following web page: `http://ego.psych.mcgill.ca/misc/fda/`.

- *Phoneme data*: The third data set correspond to 800 discretized log-periodograms of the phonemes "aa" and "ao". Each phoneme is associated to a class of the experiment. A plot of 25 series of each class is shown in Figure 6.4. This data set is available in `http://www.math.univ-toulouse.fr/staph/npfda/`.

- *Spectrometric data*. This data set is made of 215 observation is the near infrared absorbance spectrum of a meat sample. Each observation consists in a 100 channel spectrum of absorbance in the wavelength range from 850 to 1050 nm recorded on a Tecator Infratec Food and Feed Analyzer. The two classes are determined by those samples with more (class 1) or less (class 2) than a 20% of fat content. In Figure 6.5 we show the original curves. This data set can also be download from `http://www.math.univ-toulouse.fr/staph/npfda/`.

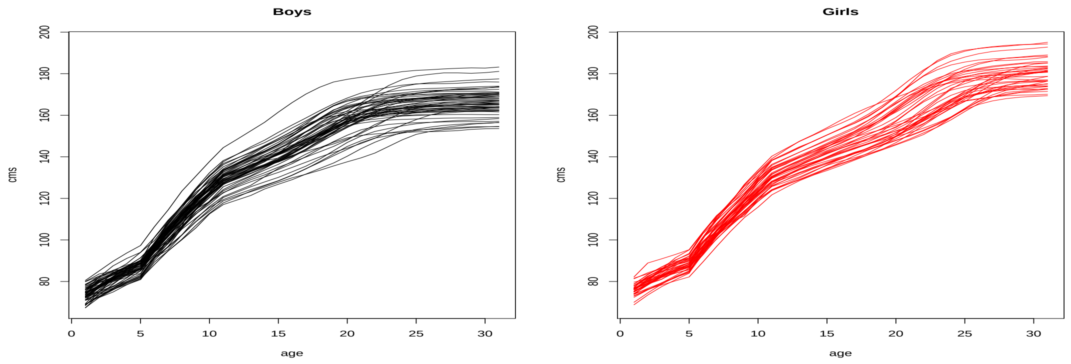


Figure 6.3: Growth data.

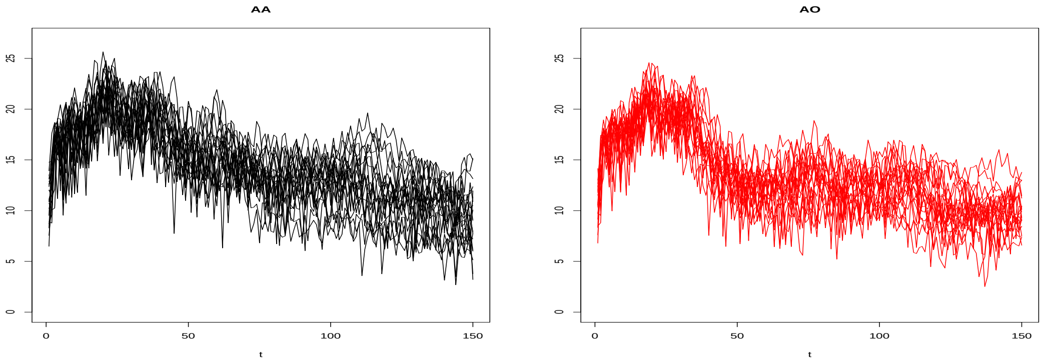


Figure 6.4: Phoneme data.

To test our methodology we follow the the same comparative scheme used in the previous section. However, in this case we optimize the parameters of the Gaussian and Laplace kernels by means of the SIC described in Section 3.4.2. We fix the penalization parameter $\gamma = 10^{-3}$ and we search the ρ parameter value (in both kernels) in a grid of 100 values in the interval $[10^{-4}, 10^{-1}]$. The optimal ρ^* is fixed as the value that minimizes the avaraged SIC for each set of sample curves $\{f_{n,1}, \ldots, f_{n,m}\}$. Denote by $f^{*l}_{K_{\rho_i},\gamma,n}$ the projection of f_ν (see eq. 3.1) onto the RKHS associated to the parameter dependent kernel K_{ρ_i} (Gausian or Laplace in this example) using $f_{n,l}$. Then the optimal ρ^* is given by

$$\rho^* = arg\min_{\rho_i} \frac{1}{m} \sum_{l=1}^{m} SIC(f^{*l}_{K_{\rho_i},\gamma,n}) \; for \; i = 1, \ldots, 100, \tag{6.1}$$

See eq. (3.25) for details regarding the estimation of the SIC.

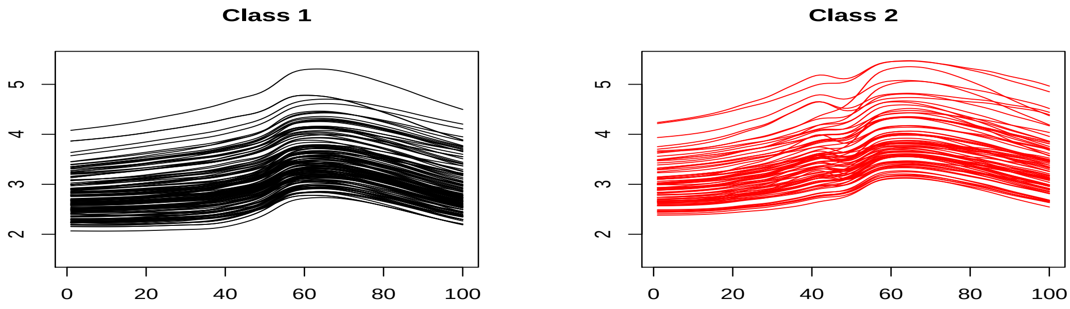


Figure 6.5: Spectrometric data.

Results are shown in Table 6.2. In agreement with the previous experiment, to project the curves onto the proposed RKHSs improves the results achieved by the classification procedures using the raw data. Just one exception appears, the Phoneme data using the LDA/PLS procure where non effective improvement is obtained. In this case, the best projection has an error of 19.35% misclassified curves (using S) while the error for the raw data is 19.13%.

Regarding the Growth data, the best result corresponds to the LDA/PLS technique combined with a P-splines kernel. It is remarkable that for this data set the projection using the P-Splines kernels achieve the minimum error in the four classification procedures.

The Support Vector Machine combined with Laplace kernel obtains the lower errors in the Spectrometric data (1.54%). Regarding the Phoneme data set, the best result is also obtained for the SVM (18.14%) but in this case combined with the Gaussian kernel. This is a clear example of how the use of generalized covariance functions is useful to improve the classical FDA approach (that focuses on specific basis of functions instead of generalized covariances).

To conclude the analysis we check the accuracy of the previous results by comparing the errors in Table 6.2 with those achieved by two techniques specificaly designed to deal with functional data:

- The P-spline Signal Regression (PSR) (Marx and Eilers, 1999).
- The Non Parametric Curves Discrimination (NPCD) (Ferraty and Vieu, 2003). This procedure uses a semi-metric to obtain the distance between the curves. We consider in the experiments the Partial Least Squares (for a number of components

fixed by cross validation for $p = 1, \dots, 10$) and the derivative semi-metrics (d_2). Only the result obtained with the best semi-metric is consider in each analysis for comparative purposes.

In Table 6.3 we compare the best results from Table 6.2 (for each data base) with the results obtained by previous techniques. It is clear that we are able to outperform their classification errors in the three cases and specially for the Growth data set. The PSR misclassifies a 5.21% of the curves, the MPLSR with a derivative semi-metric the 4.49% while we obtain an error of 1.16% using the LDA/PLS procedure using the P-splines kernel projection.

6.2.3 Combining Projections

The purpose of this experiment is to merge the ideas described in Chapter 3 and Section 4.3.3. Given a set of curves the main tasks are: (1) to project the curves onto a set of different RKHSs and (2) combine the set of projections in order to improve the classification results of a variety of classifications techniques. We will use the discrimination procedures described a the beginning of this chapter and we will analyze the Spectrometric data mentioned above.

We consider a set of five Gaussian kernels $K(x, y) = \exp\{-\rho\|x - y\|^2\}$ with parameters in a wide range of values. We fix $\rho_i \in \{2.5, 1, 0.5, 0.1, 0.01\}$ and $\gamma = 10^{-2}$ in eq. (3.2) to estimate a set of five projections of the curves. We calculate the five Empirical Regularized inner product matrices whose components are defined in eq. (3.14). We denote by $\mathbf{K}_1, \dots, \mathbf{K}_1$ the final five estimated matrices.

We consider the AKM, MAKM, Max-Min and AV methods (see Section 2.8 for details) to combine the previous inner product matrices (previously converted to similarities). The parameter τ in the MAKM and AV methods is fixed by cross validation. We transform each combination matrix to positive definite by Π^1_+ (see eq. 4.17) and we calculate the associated kernel functions K^*_{AKM}, K^*_{MAKM}, $K^*_{Max-Min}$ and K^*_{AV} as described in Section 4.3.3.

We compare the errors of SVM, FDA_{mars}, FDA_{bruto} and LDA/PLS by using the five original kernel matrices $\mathbf{K}_t$ for $t = 1, \dots, 5$ and the four combinations. We also classify the curves using the raw data to study how the new representations improve the performance of the original techniques. We use 80% of the data for training and 20% for testing. In Table 6.4 we show the results of the experiments for 100 runs. Regarding the representations induced by the oringinal Gaussian kernels we only include the performance of the best and the worst Gaussian kernel.

Table 6.2: Errors of the four classifications algorithms and 5 curves representations (+ the raw data) in three real data sets after 100 runs. In italic letters the best technique of each row is remarked while in bold letters we point out the best global result. In parenthesis the standard errors of the averaged errors are shown.

Growth data Method/RKHS	B-Splines	P- Splines	Cov.	RBF	Laplace	Raw data
SVM	0.0600 (0.0075)	*0.0158* *(0.0042)*	0.0326 (0.0052)	0.0568 (0.0071)	0.0400 (0.0059)	0.0811 (0.0076)
FDA_{bruto}	0.0368 (0.0055)	*0.0316* *(0.0048)*	0.0347 (0.0050)	0.0368 (0.0056)	0.0516 (0.0054)	0.3695 (0.0163)
FDA_{mars}	0.0463 (0.0058)	*0.0442* *(0.0049)*	0.0579 (0.0062)	0.0484 (0.0058)	0.0684 (0.0066)	0.0832 (0.0084)
LDA/PLS	0.0200 (0.0048)	**0.0116** **(0.0042)**	0.0211 (0.0040)	0.0200 (0.0048)	0.0305 (0.0047)	0.0379 (0.0056)
Spect. data Method/RKHS	B-Splines	P- Splines	Cov.	RBF	Laplace	Raw data
SVM	0.0179 (0.0025)	0.0833 (0.0051)	0.0162 (0.0027)	0.0183 (0.0025)	**0.0154** **(0.0024)**	0.0231 (0.0024)
FDA_{bruto}	0.0675 (0.0079)	0.0600 (0.0043)	0.0621 (0.0090)	*0.0571* *(0.0079)*	0.0617 (0.0096)	0.3367 (0.0117)
FDA_{mars}	0.0371 (0.0038)	0.0554 (0.0043)	*0.0296* *(0.0030)*	0.0358 (0.0031)	0.0325 (0.0031)	0.0733 (0.0052)
LDA/PLS	0.0896 (0.0053)	0.0925 (0.0061)	*0.0871* *(0.0049)*	0.1075 (0.0055)	0.0879 (0.0060)	0.0929 (0.0059)
Phoneme data Method/RKHS	B-Splines	P- Splines	Cov.	RBF	Laplace	Raw data
SVM	0.1835 (0.0036)	0.1842 (0.0033)	0.1924 (0.0035)	**0.1814** **(0.0036)**	0.1830 (0.0036)	0.2328 (0.0053)
FDA_{bruto}	0.1867 (0.0036)	0.1849 (0.0037)	0.1958 (0.0034)	*0.1831* (0.0035)	0.1872 *(0.0034)*	0.2187 (0.0037)
FDA_{mars}	0.1926 (0.0036)	0.2019 (0.0038)	0.2041 (0.0039)	*0.1918* (0.0033)	0.1964 *(0.0034)*	0.2695 (0.0050)
LDA/PLS	0.1990 (0.0039)	0.2263 (0.0036)	0.1935 (0.0035)	0.1966 (0.0037)	0.2006 (0.0037)	*0.1913* *(0.0039)*

The best result is achieved by the Support Vector Machine and the MAKM combination with a misclassification error of 1.69%. However the most interesting conclusions of this

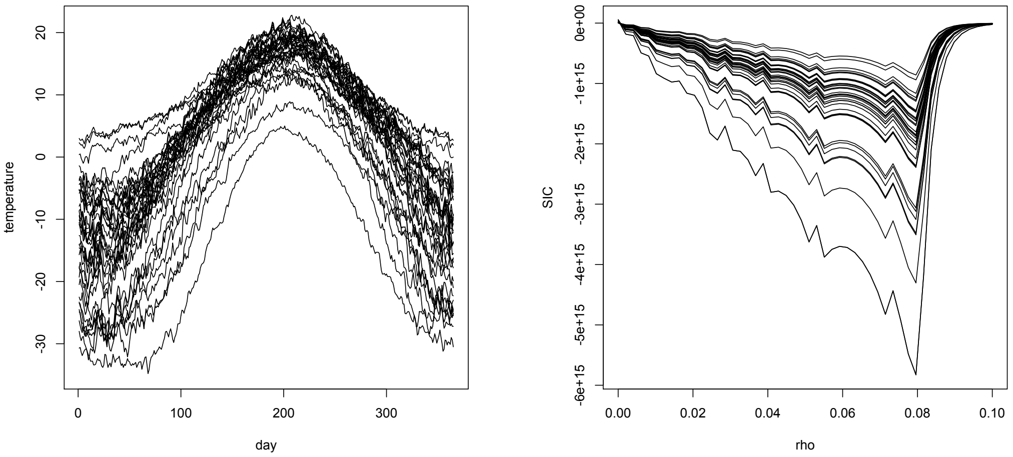
Table 6.3: Results for the best projection method and two techniques specially designed to work with functional data in the three data sets.

Growth data	PSR	$NPCD_{d^2}$	Best projection
Test Error	0.0521 (0.0045)	0.0494 (0.0400)	**0.0116** **(0.0042)**
Tecator data	PSR	$NPCD_{d^2}$	Best projection
Test Error	0.0736 (0.0039)	0.0218 (0.0021)	**0.0154** **(0.0027)**
Spect. data	PSR	$NPCD_{p=5}$	Best projection
Test Error	0.1866 (0.0085)	0.1928 (0.0031)	**0.1814** **(0.0036)**

Table 6.4: Results for the Spectrometric data using different projections and combinations.

	Raw data	Best G.	Worst G.	K^*_{AKM}	K^*_{MAKM}	$K^*_{Max-Min}$	K^*_{AV}
SVM	0.0231 (0.0024)	0.0554 (0.0066)	0.0892 (0.0066)	0.0175 (0.0034)	**0.0169** (0.0033)	0.0644 (0.0067)	0.0402 (0.0055)
FDA_{bruto}	0.3367 (0.0117)	0.0835 (0.0121)	0.3552 (0.0146)	**0.0488** (0.0058)	0.0508 (0.0063)	0.0600 (0.0064)	0.0583 (0.0068)
FDA_{mars}	0.0733 (0.0052)	0.0600 (0.0200)	0.3427 (0.0197)	0.0669 (0.0194)	**0.0588** (0.0143)	0.1354 (0.0258)	0.0688 (0.0129)
LDA/PLS	0.0929 (0.0058)	0.0829 (0.0089)	0.0838 (0.0089)	0.0831 (0.0089)	0.0829 (0.0089)	0.0727 (0.0072)	**0.0719** (0.0078)

experiment are obtained comparing the influence of the representation system (single kernels projections or combinations) in the classification results of the four discrimination techniques. The curves projection obtained via the Best Gaussian kernel improves, in three of the four cases, the raw data. The only exception is the the Support Vector Machine whose error using the raw data is 2.31% while the best Gaussian achieves an error of 5.54%. Nevertheless, always happen that some of the combinations improve both, the raw data and the best Gaussian. For instance the SVM using the MAKM combination achieves an error of 1.69% improving the error obtained by the best Gaussian, 5.54%. In addition, to combine the projection makes the representation methodology robust against wrong choices of the original projections. For the AKM, MAKM and AV methods, the inclusion of kernels with poor performance does not affect the final performance of the final combinations as is shown in Table 6.4.



(a) Series of averaged daily temperature in 35 cities of Canada

(b) Value of the SIC for the 35 projected series using a Gaussian kernel for different values of ρ.

Figure 6.6: Set of temperature series and values of the SIC for different values of ρ.

6.3 Cluster experiments in FDA

Next we include a couple of examples where the cluster structure of two different sets of funcional data is revealed by using the projection method described in Chapter 3.

6.3.1 Cluster of temperature series and model selection criteria

In this example we analyze the whole set of temperature curves described in Example 3.1. See Figure 6.6 a). The objective is to find the hidden cluster structure of the curves and to study it in terms of climate regions in Canada. To this aim we proceed in two steps: (1) we project the time series onto certain RKHS and (2) we apply a cluster procedure over the projections.

To select the RKHS where project the curves we use the SIC criteria described in Section 3.4.2. We optimize the parameter ρ of the Gaussian kernel from a set of 50 equally spaced values in the interval $[10^{-7}, 10^{-1}]$ and we fix $\gamma = 1$.

In this case the optimal value of ρ using the SIC (that one that minimized the averaged SIC for the set of series) is 0.0791. See Figure 6.6 b). We project the series using this

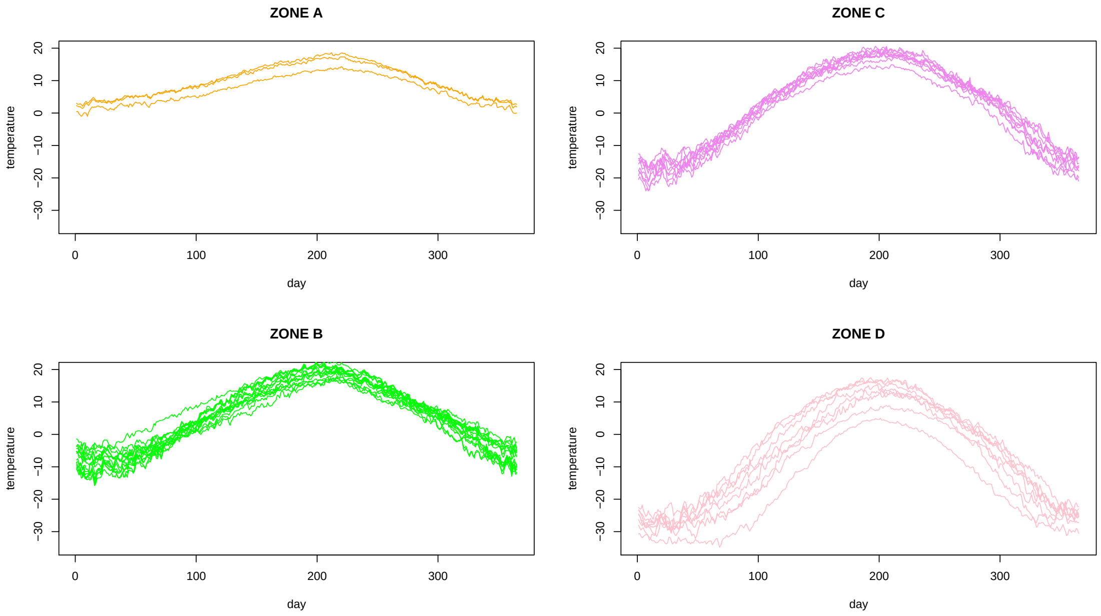


Figure 6.7: Clusters of the Canadian temperature data set using a Ward method over the projections obtained for a Gaussian kernel with $\rho = 0.0791$ and $\gamma = 1$.

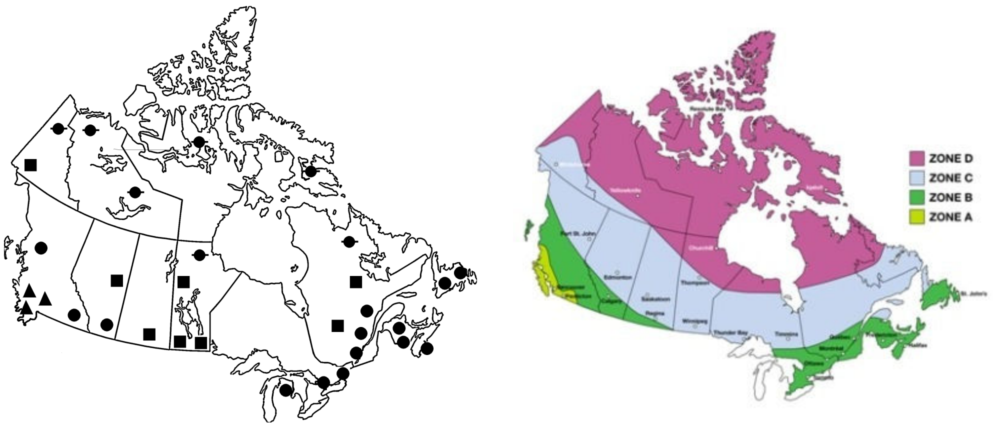
parameter and we apply a hierarchical cluster method over the projections (Ward method). Using a priori information about the climate in Canada (www.nrcan.gc.ca), we know that in this country there exist four climate zones (see Figure 6.8 b). Therefore we decide to retain 4 clusters. The series corresponding to each one of the obtained clusters are drawn in Figure 6.7. In addition, in Figure 6.8 a) we show the location of each series and we point out the clusters they belong to. The cities assigned to each cluster are detailed as follows:

- Zone A (Cirles with horizontal line): Scheffervll, Churchill, Uranium, Cty. Dawson, Yellowknife, Iqaluit, Inuvik, Resolute.

- Zone B (Squares): Arvida, Bagottville, Thunderbay, Winnipeg, The Pas, Regina, Pr. Albert, Edmonton, Whitehorse.

- Zone C (Triangles): Vancouver, Victoria, Pr. Rupert.

- Zone D (Circles): St. Johns, Halifax, Sydney, Yarmouth, Charlottvl, Fredericton, Quebec, Sherbrooke, Montreal, Ottawa, Toronto, London, Calgary, Kamloops, Pr. George.



(a) Stations and clusters

(b) Climate models

Figure 6.8: Map of the stations locations and map of the four climate zones in Canada. Image Source: Office of Energy Efficiency Canada

As can be seen in Figure 6.7, there is a perfect mach between the four discovered clusters and the four existing climate regions.

To check the effectiveness of the selected projection we repeat the previous analysis with a different value of ρ within the interval $[10^{-7}, 10^{-1}]$. In particular we fix $\rho = 10^{-5}$ and we show in Figure 6.9 the new clusters obtained this way. It is clear that the obtained clusters change and the four climate regions are not properly revealed in this case.

6.3.2 Cluster and classification of gene profiles

In this example we work with a data set previously analyzed in (Spellman et al., 1998) [1] and made up of a set of measurements of gene expressions. The original experiment deals with a set of yeast cells whose cycles were synchronized by a chemical process. The final data are therefore time series of cDNA micro-arrays gathered over 18 equally spaced time points for the full 6,178 genes in the yeast genome.

Following (Spellman et al., 1998), in this example we will focus on the $n = 612$ genes without missing values that present a clear pattern related to the cell cycle. In the original experiment, these genes were labeled (according to the phase of the yeast cell cycle) in five different categories: M, G1, G2, S, and M/G1. See Figure 6.10. In these plots the horizontal axis represents the time and the vertical axis the relative level of gene expression. The five genes groups are shown separately and the averaged profile for the each group is also included at the bottom (right) of the figure.

[1] Data available from the web site: http://www.stanford.edu/cellcycle/

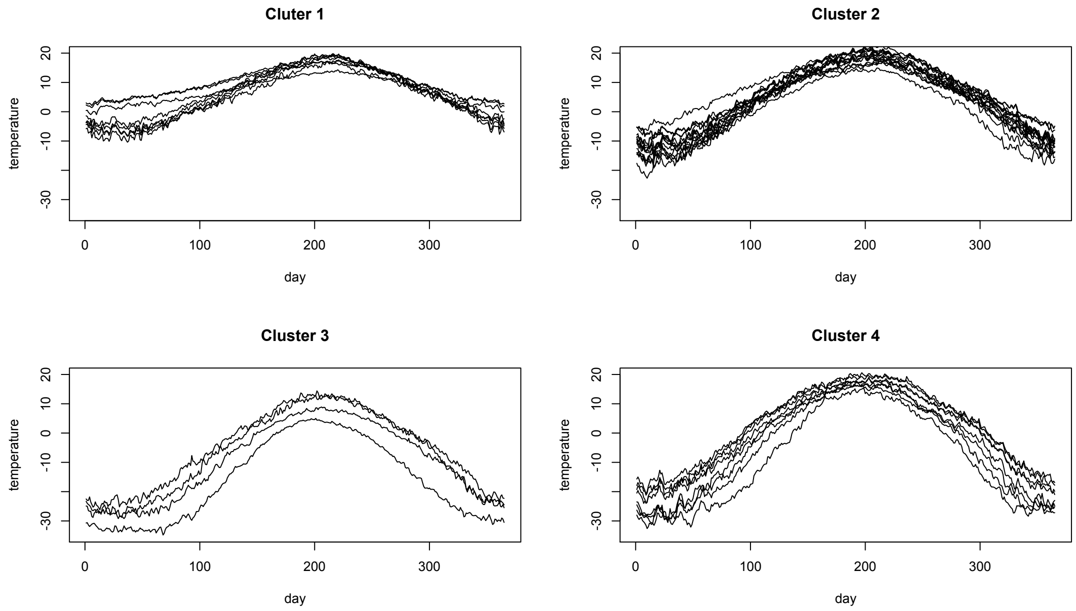


Figure 6.9: Clusters of the Canadian temperature data set using a Ward method over the projections obtained for $\rho = 10^{-5}$ and $\gamma = 1$.

Although the original grouping of the gene profiles was done considering the phase of the yeast cell cycle, the defined classes do not capture the different types of cycles exhibited in the genes profiles. For instance, in Figure 6.10 it is apparent that group S is formed by genes that presents at least two different patterns. As we will study next, this mixture of genes cause problems in the performance of discrimination procedures. Hence two are the tasks that we aim to afford in this example: First, to solve the genes profiles classification problem using the original labels and second to propose a new grouping that explain the cell cycle patters.

We project the profiles using eq. (3.2) for K a of Gaussian kernel with parameter $\rho = 0.07$ and for $\gamma = 0.00027$. These values are selected by cross validation over the classification errors. In Figure 6.11 (left) we show one gene profile and its projection onto the RKHS induced by K. In Figure 6.11 (rigth) we show the estimated weights (λ_j^*) for this gene.

We divide the sample in 80% of the data for training and 20 % for testing and classify the genes using a linear in two cases: first using the previous estimated projection and second using the raw data of the problem.

Classification results are shown in Table 6.5 (first two rows). The error after 100 runs is

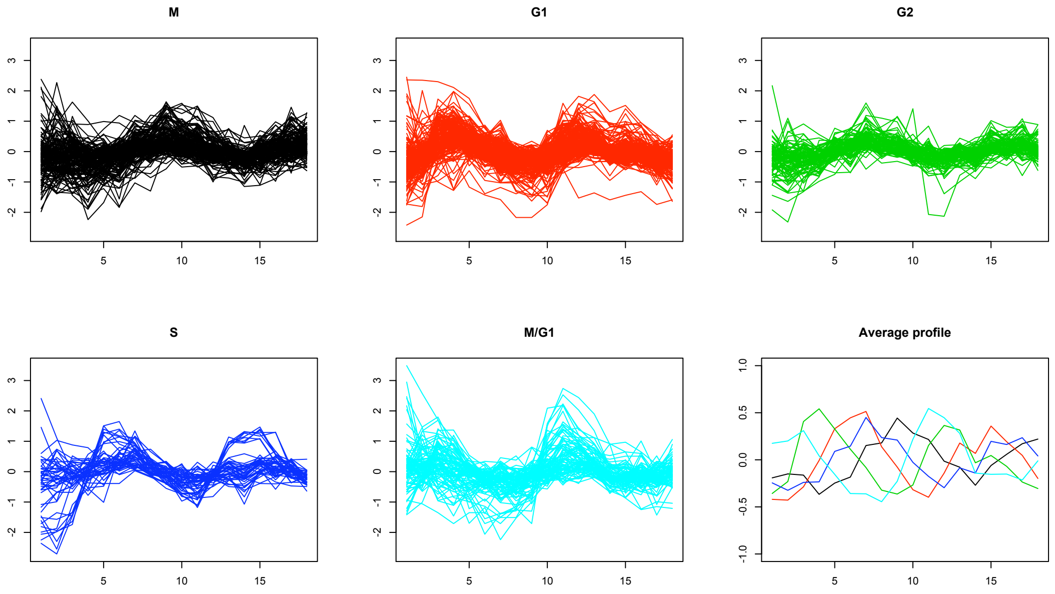


Figure 6.10: Raw time series view of the 612 gene profiles. Profiles for the groups M, G1, G2, S, and M/G1 are shown. A plot of the averaged profiles of each group is also shown.

Table 6.5: Comparative results for the Cell cycle data after 100 runs.

Method	Test Error	Std .Dev.
SVM, raw data	0.3174	0.0028
SVM, RKHS	**0.3115**	**0.0026**
SVM , raw data (new classes)	0.2362	0.0023
SVM, RKHS (new classes)	**0.1631**	**0.0021**

around 31% in both cases. Notice that, in contrast to the previous experiments where project the data always improved the classification errors, now the errors does not significantly change. This is due to the original genes labeling that does not reflect the natural cluster structure of the genes and that cause that we cannot improve the classification results.

To determine a more realistic genes labeling we proceed by applying a hierarchical cluster procedure over the projections of the genes. In particular we apply the Ward method and, based on the infomation provided by the cluster dendrogram (see Figure 6.12), we fix to 7 the number of clusters to retain. See Figure 6.12. Clusters 1, 2 and 7 present very flat, but different cycle patterns. The genes of these three classes seems to be on the

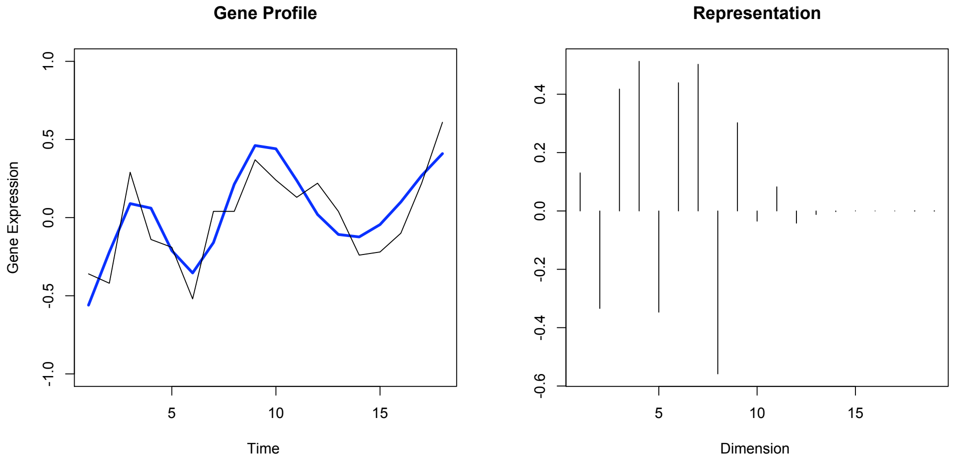


Figure 6.11: On the left, profile of one of the genes of the data base and its projection onto a RKHS of Gaussian kernel of parameter $\sigma = 0.07$. On the right, the RKHS representation of the gene is shown.

Table 6.6: Migration table beween the original and the new clusters.

Labels	1	2	3	4	5	6	7
M/G1	21	26	15	0	0	26	4
G1	48	83	64	0	2	8	18
S	26	6	0	0	9	2	4
G2	75	2	0	3	3	1	8
M	87	10	0	37	0	10	15

limit of being considered real cell cycle genes. However, clusters 3, 4, 5 and 6 present a more significance cycle trend of different frequency and and period. These clusters are homogeneous and their patterns very different.

To check the effect of the new grouping, we classify again the genes by using the new labels. Results are shown in the two last rows of Table 6.5. In this case, the projection clearly improve the classification results, being 16.13% the test errors using the projection and 23.62% for the raw data. Notice that projecting the gene profiles onto the RKHS induced by K, is useful in two senses: to identify new cluster structure in the data and improve the classification results. To conclude we include Table 6.6 where we illustrate how the new clusters have been created. Remark that the genes that originally belong to the group S migrate to clusters 1, 2 and 5 solving the initial problem of having genes of very different profiles within the same group.

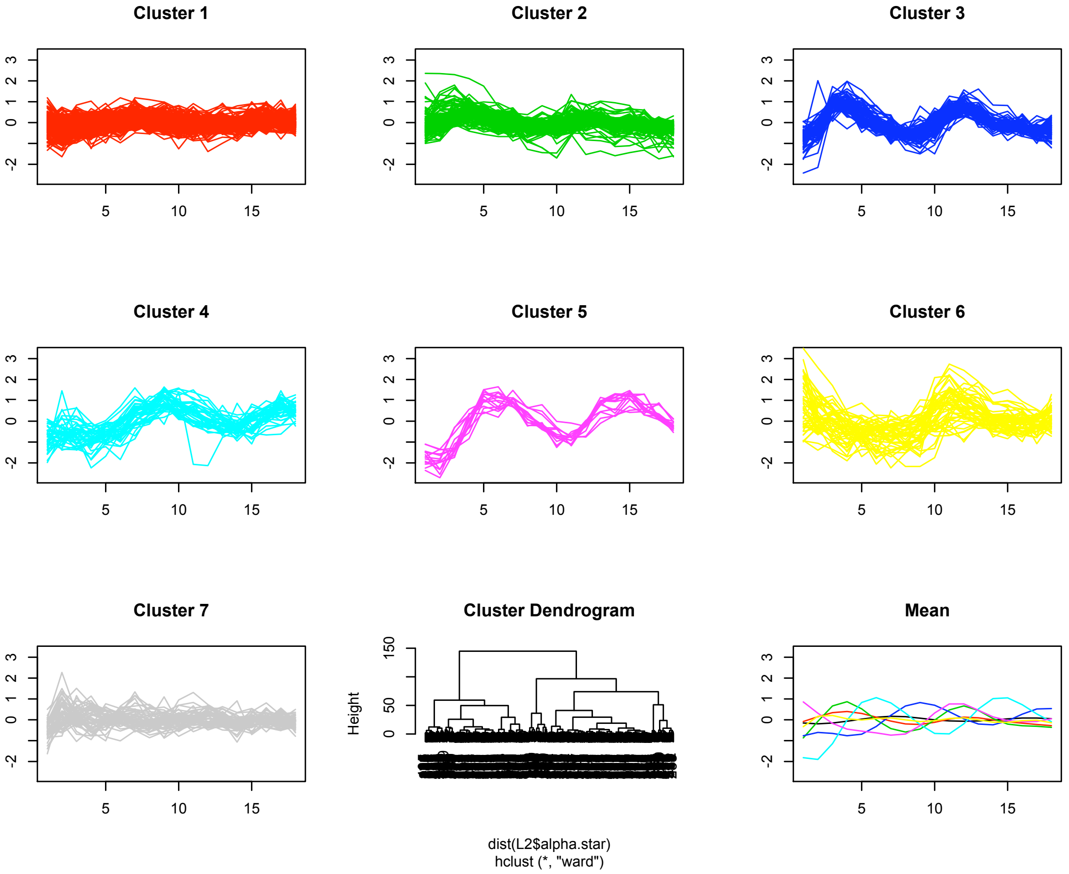


Figure 6.12: Seven clusters obtained after projecting the data onto the proposed RKHS. The obtained dendrogram (using the Ward method) and the averaged profile of the new classes are also inlcuded.

6.4 Partially labeled classification problems

In this section we check the performance of the methodology described in Section 4.3.1 in four real data sets. The idea is to show that the Linear Discriminant analysis can be improved in real examples using the information provided by unlabeled points. The four data sets used in this experiment are available in the the UCI repository [2] and they are named as: Iris, Connectionist Bench (Sonar, Mines vs. Rocks), Breast Cancer Wisconsin and Blood Transfusion. In the Iris data set we only considered the classes versicolor and virginica.

In the original data sets all the points are labeled. Hence, to test our procedure, we con-

[2] http://archive.ics.uci.edu/ml/datasets.html

struct artificially the partially labeled classification problem as follows. First, we divide each data set in two subsets, one of size n (training sample s_n in Section 4.3.1) to estimate the discrimination function and the other to calculate the classification errors (test sample). Within the training sample, we assume that only a randomly selected amount t of data are labeled (sample s_t) and we use the rest $n-t$ non labeled points to improve the classifier. We study the performance of the proposed methodology selecting different values of t for each data set (see Table 6.7 for details).

We estimate the LDA discrimination function in two cases. First, using only the labeled data, that is the matrix $S\big|_{s_t} = \mathbf{X}^T\mathbf{X}$, and second using the kernel matrix proposed in Section 4.3.1 (that is the matrix $K^*\big|_{s_t}$). In the estimation of K^* we fix $\gamma = \sigma$ and we select these parameters by cross validation in a range of 15 equidistant values in intervals specified in Table 6.7. In addition we fix by cross validation over the errors the number eigenfunctions used to reconstruct K^*. In the estimation of K^* we use the neighborhood information of the data. Hence we also include in the experiment a k-nearest neighbor algorithm for $k=1$ for comparative purposes.

In the four data sets some improvement is obtained using the proposed methodology. Regarding the Iris data set, the LDA using $K^*\big|_{s_t}$ as the input matrix always outperform both, the LDA with $S\big|_{s_t}$ and the K-nn. This behavior is also observed in the Transfusion data set where our proposal is the best technique in the three analyzed cases (for $t = 10, 50, 100$). In the Sonar and Cancer data set, there is not an clear best technique. In the first case the $K-nn$ wins in two of the three analyzed scenarios while for the Cancer data our proposal is the best technique for $t = 10, 50$. In this particular example, when $t = 100$, the LDA with $S\big|_{s_t}$ obtains the best results.

That the use of the non-labeled points helps us to improve the classification results of both, LDA and K-nn in 8 of the 11 proposed scenarios what clearly indicated the usefulness of our methodology.

6.5 Classification problems with asymmetry

In this section we present the analysis of three real examples where the asymmetric similarity between the data plays an important role. The objective of the experiments is twofold: First, to show the importance of considering asymmetric proximities between the data in classification problems. Second, to illustrate how the use of labels helps to significantly improve the classification results when the matrices in which **S** (the asymmetric data symmilarity) decomposes are combined. To this aim, the four statistical

Table 6.7: Comparison table of the performance of the FDA procedure using two different inputs matrices: $S\big|_{s_t} = \mathbf{X}^T\mathbf{X}$ and $K^*\big|_{s_t}$. Four real examples are shown for different values of the amount of labeled points. Results are obtained after 30 runs. Standard deviations are shown in parenthesis.

Data Set	**Params.** γ, σ	**Partition Train/Test**	**Partition Train (t, n-t)**	**Test Error** $LDA + S\big|_{s_t}$	**Test Error** $LDA + K^*\big|_{s_t}$	**Test Error** $1 - nn$
Iris	$[3.5, 6.5]$	(60,40)	(10,50)	0.0958 (0.0190)	**0.0833** **(0.0130)**	0.1008 (0.0430)
			(50,10)	0.0500 (0.0001)	**0.0250** **(0.0080)**	0.0766 (0.0063)
Sonar	$[0.05, 0.1]$	(170,38)	(75,95)	0.3491 (0.0140)	0.2964 (0.0152)	**0.2596** **(0.047)**
			(100,70)	0.3043 (0.0121)	**0.1605** **(0.0090)**	0.2368 (0.0512)
			(150,20)	0.2429 0.0063	0.2438 (0.0061)	**0.2228** **(0.0256)**
Cancer	$[1.5, 2.1]$	(400,283)	(10,390)	0.1253 (0.0114)	**0.0628** **(0.0086)**	0.0667 (0.0260)
			(50,350)	0.0502 (0.0031)	**0.0398** **(0.0018)**	0.0591 (0.0193)
			(100,300)	**0.0369** **(0.0011)**	0.0380 (0.0005)	0.0501 (0.0149)
Transf.	$[0.1, 1]$	(400,384)	(10,390)	0.2529 (0.0168)	**0.2274** **(0.0250**	0.2917 (0.0758)
			(50,350)	0.1845 (0.0037)	**0.1798** **(0.0007)**	0.2717 (0.0430)
			(100,300)	0.1800 (0.0034)	**0.1788** **(0.0008)**	0.2720 (0.0251)

classification techniques described at the beginning of this chapter are compared in two cases: when the asymmetric proximities between the data are considered and when the raw data (Euclidean distance) are used as input of the techniques.

In order to use asymmetric proximities to classify the data, we consider the two matrix decomposition (polar and traingular) described in Section 4.3.2. We classify the points by using separately $\mathbf{S}_1$ and $\mathbf{S}_2$ (or $\mathbf{M}_1$ and $\mathbf{M}_2$ in the polar case) and by their combination via $\frac{1}{2}(\mathbf{S}_1 + \mathbf{S}_2)$, the Semi-definite programming (S. D. P.) described in (Lanckriet et al., 2004) and the combination proposed in eq. (4.15). In this latest case we estimate the integral operator associated to the obtained similarity as described in Section 4.3.2 (Table 4.2) in order to estimate the value of the similarity it in test points. The choice of the parameter τ in eq. (4.15) is done by cross validation over the test error in a range of 25 equidistant values in the interval $[10^{-5}, 10^{-2}]$ for the first two experiments. In the third one the interval is fixed to $[10^{-5}, 10^{-3}]$. For the triangular decomposition we use both projections Π_+^1 and Π_+^2 (described in Section 4.3.2) to transform the matrices to positive definite.

The classification errors are obtained after 50 runs where 80% of the data are used for training and the remaining 20% for testing.

6.5.1 Term classification in text data bases

The goal of this first experiment is to classify the terms of the database described in Example 4.2 using the information provided by the matrix $\mathbf{S}$ whose components are given by eq. (4.23). Since $\mathbf{S}$ is asymmetric we consider its triangular ($\mathbf{S}_1$ and $\mathbf{S}_2$) and polar decomposition ($\mathbf{M}_1$ and $\mathbf{M}_2$). To classify the terms we consider independently the two sources of asymmetry (obtained via the two matrices decompositions) as well as their combinations.

Results are shown in Table 6.8. Several conclusions can be obtained. Fist of all it, is clear that the combination of the sources asymmetry generally improves the classification compared to the individual matrices ($\mathbf{S}_1$, $\mathbf{S}_2$ or $\mathbf{M}_1$, $\mathbf{M}_2$). In particular, the combination proposed in eq. (4.15) clearly shows the best performance in this experiment when it is used to fuse $\mathbf{M}_1$ and $\mathbf{M}_2$. Regarding the matrix projection method, we conclude that there is not a universal better method.

In Table 6.9 we compare the best test errors obtained for each classification procedure (results in bold in Table 6.8) with the errors obtained when the raw data are used to train the classifiers, that is using as input the df. representation of the terms. In addition we include the relative improvement obtained in each case. The four methods are clearly outperformed when the asymmetry is considered. The technique that is improved the most is the FDA_{bruto} (68%) while the best global method is the SVM with a 13.49% of misclassification data.

6.5.2 Classification of microarrays data

A genetic expression data base is usually a collection of DNA microarray experiments where each column represents an experiment and each row a different gene. Generally there are thousand rows and a few experiments. Each component of the data matrix measures the expression level of each gene in the target relative to each reference sample (experiment).

Table 6.8: Comparative results for the terms data after 50 runs using the triangular (with projections Π^1_+ and Π^2_+) and the polar decomposition in four classification techniques. Standard deviation of the errors are shown in parenthesis. In bold letters the best results for each technique is emphasized.

Triang. Dec. + Π^1_+	$\mathbf{S}_1$	$\mathbf{S}_2$	$\frac{1}{2}(\mathbf{S}_1 + \mathbf{S}_2)$	$\frac{1}{2}(\mathbf{S}_1 + \mathbf{S}_2) + \tau \mathbf{S}_y$	S.D.P
SVM	0.2424 (0.0098)	0.2278 (0.0093)	0.2357 (0.0078)	0.1506 (0.0061)	0.1819 (0.0068)
FDA_{bruto}	0.2345 (0.0095)	0.2333 (0.0093)	0.1792 (0.0082)	0.1541 (0.0062)	0.2580 (0.0077)
FDA_{mars}	0.2756 (0.0095)	0.2811 (0.0093)	0.2027 (0.0082)	0.1722 (0.0059)	0.2580 (0.0079)
LDA/PLS	0.2035 (0.0083)	0.1945 (0.0084)	0.1929 (0.0080)	0.1549 (0.0063)	0.1486 (0.0069)
Triang. Dec. + Π^2_+	$\mathbf{S}_1$	$\mathbf{S}_2$	$\frac{1}{2}(\mathbf{S}_1 + \mathbf{S}_2)$	$\frac{1}{2}(\mathbf{S}_1 + \mathbf{S}_2) + \tau \mathbf{S}_y$	S.D.P
SVM	0.2471 (0.0096)	0.2373 (0.0096)	0.2333 (0.0072)	0.1518 (0.0063)	0.1819 (0.0068)
FDA_{bruto}	0.2192 (0.0106)	0.2224 (0.0105)	0.1608 (0.0086)	0.1498 (0.0066)	0.2580 (0.0077)
FDA_{mars}	0.2796 (0.0106)	0.2576 (0.0105)	0.2360 (0.0086)	0.1746 (0.0060)	0.2580 (0.0077)
LDA/PLS	0.2086 (0.0093)	0.1878 (0.0082)	0.1780 (0.0067)	0.1525 (0.0063)	0.1486 (0.0069)
Polar Desc.	$\mathbf{M}_1$	$\mathbf{M}_2$	$\frac{1}{2}(\mathbf{M}_1 + \mathbf{M}_2)$	$\frac{1}{2}(\mathbf{M}_1 + \mathbf{M}_2) + \tau \mathbf{S}_y$	S.D.P
SVM	0.1851 (0.0069)	0.1965 (0.0078)	0.1678 (0.0066)	**0.1349 (0.0052)**	0.2521 (0.0074)
FDA_{bruto}	0.1690 (0.0064)	0.1784 (0.0078)	0.1494 (0.0067)	**0.1424 (0.0064)**	0.1898 (0.0077)
FDA_{mars}	0.2175 (0.0081)	0.2415 (0.0082)	0.2462 (0.0080)	**0.1623 (0.0054)**	0.1898 (0.0077)
LDA/PLS	0.1655 (0.0062)	0.2137 (0.0080)	0.1655 (0.0062)	**0.1404 (0.0056)**	0.1432 (0.0077)

Table 6.9: Comparative results for the textual data after 50 runs of the experiment. Four classification techniques using the original data (df. representation) as input and the results considering the asymmetric dissimilarities are compared. Best results for each classification methods are shown in bold letters.

Method	SVM	FDA_{bruto}	FDA_{mars}	LDA/PLS
Original data, df.	0.1957 (0.0012)	0.4520 (0.0007)	0.3416 (0.0013)	0.1522 (0.0009)
Best assym. result.	**0.1349 (0.0052)**	**0.1424 (0.0064)**	**0.1623 (0.0054)**	**0.1404 (0.0056)**
Relative improv.	31.06%	68.49%	52.48%	7.75%

In this experiment we work with a Human Microarray Cancer data set (Hastie et al., 2009). The data correspond to 64 samples where the level of expression of 6830 genes were measured. The range of the original data was from say -6 to 6 measuring the expression level of each gene. These values are recoded to 1 for expressed genes (positive values) and 0 for non expressed genes (negative values). Then there exits a correspondence with the previous example: the gene plays the role of the terms and the sample plays the role of the document. Hence it makes sense to use the asymmetric similarity s_{ij} defined in eq. (4.23) to analyze this type of data.

In this experiment we select randomly a total of 500 genes for the experiment and we label them by voting (based on the frequency) to the class of "renal" or "colon" cancer. We estimate their similarity matrix via eq. (4.23) and we follow the same comparative scheme of the previous example.

Classification results are presented in Table 6.10. Results are coherent with those obtained in the previous example with some exceptions. While it is true that to combine the sources of asymmetry improves the classification results of the four classification techniques, in this case the lowest errors are obtained combining $\mathbf{S}_1$ and $\mathbf{S}_2$ via eq. (4.15) with the Π_+^2 projection method. This results confirms that this combination method, that includes a term with the information of the data labels, obtains the best results when the sources of asymmetry has to be combined. In particular it always improves the average of the matrices and the S.D.P procedure.

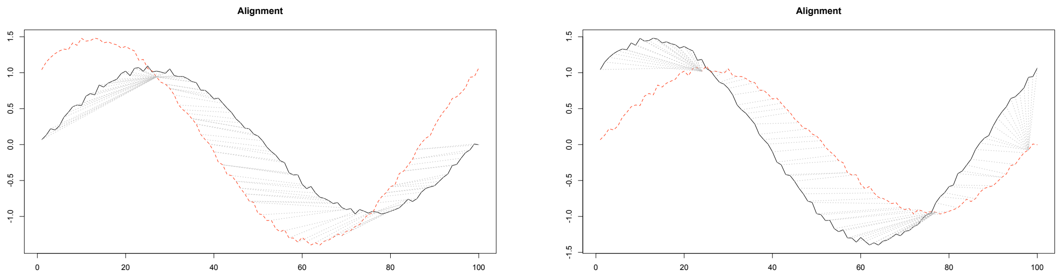
In Table 6.11 we show the relative improvements of the asymmetric schedule compared with the symmetric case (raw data). We obtaing significant improvements for the SVM (10.19%), the FDA_{mars} (2.55%) and the LDA/PLS (16.87%) method. For the FDA_{bruto} there is not significant improvement being the only case where the df. representation outperform the asymmetric schemes.

Table 6.10: Comparative results for the genetic data after 50 runs using the triangular (with projections Π_+^1 and Π_+^2) and the polar decomposition in four classification techniques. Standard deviation of the errors are shown in parenthesis. In bold letters the best results for each technique is emphasized.

Triang. Dec. + Π_+^1	$\mathbf{S}_1$	$\mathbf{S}_2$	$\frac{1}{2}(\mathbf{S}_1 + \mathbf{S}_2)$	$\frac{1}{2}(\mathbf{S}_1 + \mathbf{S}_2) + \tau\mathbf{S}_y$	S.D.P
SVM	0.0852 (0.0040)	0.0840 (0.0047)	0.0816 (0.0045)	0.0830 (0.0043)	0.1104 (0.0057)
FDA_{bruto}	0.1552 (0.0076)	0.1402 (0.0071)	0.1146 (0.0055)	0.1116 (0.0046)	0.2308 (0.0093)
FDA_{mars}	0.2052 (0.0076)	0.2078 (0.0054)	0.1876 (0.0054)	0.1371 (0.0034)	0.2936 (0.0100)
LDA/PLS	0.1036 (0.0043)	0.1010 (0.0047)	0.0974 (0.0045)	0.0950 (0.0038)	0.1576 (0.0067)
Triang. Dec. + Π_+^2	$\mathbf{S}_1$	$\mathbf{S}_2$	$\frac{1}{2}(\mathbf{S}_1 + \mathbf{S}_2)$	$\frac{1}{2}(\mathbf{S}_1 + \mathbf{S}_2) + \tau\mathbf{S}_y$	S.D.P
SVM	0.0964 (0.0049)	0.0848 (0.0044)	0.0812 (0.0046)	**0.0722 (0.0035)**	0.1104 (0.0057)
FDA_{bruto}	0.1486 (0.0072)	0.1358 (0.0057)	0.1156 (0.0055)	**0.0990 (0.0051)**	0.2316 (0.0089)
FDA_{mars}	0.2078 (0.0059)	0.2006 (0.0054)	0.1858 (0.0059)	**0.1258 (0.0043)**	0.3016 (0.0099)
LDA/PLS	0.1064 (0.0044)	0.0994 (0.0043)	0.0976 (0.0045)	**0.0898 (0.0033)**	0.1576 (0.0067)
Polar Desc.	$\mathbf{M}_1$	$\mathbf{M}_2$	$\frac{1}{2}(\mathbf{M}_1 + \mathbf{M}_2)$	$\frac{1}{2}(\mathbf{M}_1 + \mathbf{M}_2) + \tau\mathbf{S}_y$	S.D.P
SVM	0.0868 (0.0085)	0.0804 (0.0041)	0.0790 (0.0036)	0.0776 (0.0040)	0.4356 (0.0089)
FDA_{bruto}	0.1122 (0.0045)	0.1118 (0.0043)	0.1088 (0.0036)	0.1122 (0.0047)	0.1448 (0.0060)
FDA_{mars}	0.1772 (0.0045)	0.1814 (0.0043)	0.1784 (0.0036)	0.1486 (0.0050)	0.1448 (0.0060)
LDA/PLS	0.1188 (0.0122)	0.1186 (0.0118)	0.1188 (0.0122)	0.0952 (0.0069)	0.1432 (0.0067)

Table 6.11: Comparative results for the genetic data after 50 runs of the experiment. Four classification techniques using the original data (df. representation) as input and the results considering the asymmetric dissimilarities are compared. Best results for each classification methods are shown in bold letters.

Method	SVM	FDA_{bruto}	FDA_{mars}	LDA/PLS
Original data	0.0804 (0.0032)	0.1016 (0.0040)	**0.0784 (0.0048)**	0.1014 (0.0041)
Best asymmetric sim.	**0.0722 (0.0035)**	**0.0990 (0.0046)**	0.1258 (0.0043)	**0.0898 (0.0033)**
Relative improv.	10.19%	2.55%	-37.6%	11.43%



(a) Dynamic Time warping alignment with series 1 (black) as reference.

(b) Dynamic Time warping alignment with series 1 (red) as reference.

Figure 6.13: Alignment of two time series using the Dynamic Time Warping algorithm.

6.5.3 Time series classification and Dynamic Time Warping

The Dynamic Time Warping (Sakoe and Chiba, 1978) is an algorithm designed to measure the dissimilarity between two sequences of data including financial time series, spectrometric data or video patterns. The DTW algorithm works by solving an optimization problem with restrictions. Given two sequences i and j, they are "warped" in the time dimension and a measure d_{ij} of their dissimilarity is estimated.

To estimate the dissimilarity measure the DTW fix a reference sequence and finds the optimal match to the other sequence. This procedure results in an asymmetric dissimilarity: Consider for instance the two series in Figure 6.13. The alignment of the series is estimated taking as reference the series 1 (black) in Figure 6.13 a) and series 2 (red) in Figure 6.13 b). The shadow area represents in each case the dissimilarity measure between the series. In this case $d_{12} = 23.1043$ and $d_{21} = 26.273$.

In this example we work with the Phoneme data described in Section 6.2.2. For the set of curves we obtain the asymmetric matrix distance **D** between the series by the DTW algorithm. A transformation to similarity matrix is done by $s_{ij} = 1 - d_{ij}/max\{d_{ij}\}$. With **S** estimated this way, we perform the same experiment as in the previous examples. Result are shown in Tables 6.12 and 6.13.

The best results are always obtained when the combination in eq. (4.15) is applied. The triangular decompositions seems to be the best in this example. However the best projection technique is not the same for the four classification algorithms. For the SVM, FDA_{mars} and LDA/PLS the best projection method is given by Π_+^1 in contrast to the FDA_{bruto} that works better with Π_+^2.

Regarding the errors when original series are used as input to train the SVM, FDA_{bruto} and FDA_{mars} methods, they are significantly improved by considering asymmetric similarities. However the errors of the PLS/FDA procedure does not significantly change.

6.6 Proximity matrices combination and integral operators

In this example we show the performance of the kernel combination method proposed in Section 4.3.3 in a controlled two-class classification example. We generate 200 train point and 50 test points in $\mathbb{R}^2$ following:

- Class 1: $(x, y) = (u + 1, u^2 + e)$,
- Class 2: $(x, y) = (u + 7/5, -u^2 + 1 + e)$,

where $u \sim U(-1, 1)$ is a uniform random variable and $e \sim N(0, 0.1)$ a Gaussian one. The sample is shown in Figure 6.14.

We consider two kernels based on the projections of the data onto the two coordinate axis: $K_1(\mathbf{x}, \mathbf{y}) = \pi_x(\mathbf{x})\pi_x(\mathbf{y}) = x_1 y_1$ and $K_2(\mathbf{x}, \mathbf{y}) = \pi_y(\mathbf{x})\pi_y(\mathbf{y}) = x_2 y_2$. In order to estimate the accuracy of the procedure, we consider the battery of kernels of increasing complexity $K^*(d) = (K_1 + K_2)^d$ for $d = 1, \ldots, 15$.

Let $\mathbf{K}^*(d)$ denote the matrices obtained by applying the real combination kernels $K^*(d)$ to the points in the sample of 250 points. Let $K_F(d)$ denote the kernel function obtained (see Section 4.3.3) and $K_L(d) = \sum_{i=1}^{2} \lambda_i(d) K_i$ a linear combination of kernels defined as the best approximation (using the Frobenius norm) to $K^*(d)$ by using a linear combination of the K_i kernels (Muñoz and Martín de Diego, 2006). We want to compare the

Table 6.12: Comparative results for the Phoneme data with asymmetric kernel after 50 runs using the triangular (with projections Π_+^1 and Π_+^2) and the polar decomposition in four classification techniques. Standard deviation of the errors are shown in parenthesis. In bold letters the best results for each technique is emphasized.

Triang. Dec. + Π_+^1	$\mathbf{S}_1$	$\mathbf{S}_2$	$\frac{1}{2}(\mathbf{S}_1 + \mathbf{S}_2)$	$\frac{1}{2}(\mathbf{S}_1 + \mathbf{S}_2) + \tau \mathbf{S}_y$	S.D.P
SVM	0.2822 (0.0064)	0.3594 (0.0048)	0.2906 (0.0042)	**0.1931(0.0043)**	0.3072 (0.0046)
FDA_{bruto}	0.2238 (0.0049)	0.2676 (0.0052)	0.2200 (0.0047)	0.2025(0.0036)	0.4058 (0.0054)
FDA_{mars}	0.2402 (0.0044)	0.2496 (0.0048)	0.2295 (0.0041)	**0.2035 (0.0007)**	0.4195 (0.0054)
LDA/PLS	0.2215 (0.0045)	0.2461 (0.0045)	0.2182 (0.0044)	**0.1927 (0.0035)**	0.2531 (0.0071)
Triang. Dec. + Π_+^2	$\mathbf{S}_1$	$\mathbf{S}_2$	$\frac{1}{2}(\mathbf{S}_1 + \mathbf{S}_2)$	$\frac{1}{2}(\mathbf{S}_1 + \mathbf{S}_2) + \tau \mathbf{S}_y$	S.D.P
SVM	0.2677 (0.0070)	0.3541 (0.0086)	0.2878 (0.0049)	0.2007 (0.0036)	0.3072 (0.0046)
FDA_{bruto}	0.2264 (0.0043)	0.2612 (0.0058)	0.2179 (0.0034)	**0.2000 (0.0043)**	0.4037(0.0056)
FDA_{mars}	0.2301 (0.0041)	0.2485 (0.0051)	0.2284 (0.0038)	0.2105 (0.0008)	0.4256 (0.0065)
LDA/PLS	0.2185 (0.0042)	0.2357 (0.0043)	0.2139 (0.0040)	0.1966 (0.0040)	0.2531 (0.0071)
Polar Desc.	$\mathbf{M}_1$	$\mathbf{M}_2$	$\frac{1}{2}(\mathbf{M}_1 + \mathbf{M}_2)$	$\frac{1}{2}(\mathbf{M}_1 + \mathbf{M}_2) + \tau \mathbf{S}_y$	S.D.P
SVM	0.2450 (0.0042)	0.2550 (0.0040)	0.2288 (0.0041)	0.2187 (0.0043)	0.4770 (0.0066)
FDA_{bruto}	0.2108 (0.0036)	0.2196 (0.0039)	0.2140 (0.0043)	0.2284 (0.0047)	0.3402 (0.0083)
FDA_{mars}	0.2432 (0.0032)	0.2437 (0.0033)	0.2466 (0.0043)	0.2333 (0.0045)	0.3402 (0.0083)
LDA/PLS	0.2109 (0.0042)	0.2155 (0.0041)	0.2109 (0.0042)	0.2172 (0.0042)	0.2632 (0.0071)

Table 6.13: Comparative results for the Phoneme data after 50 runs of the experiment. Four classification techniques using the original data as input and the results considering the asymmetric dissimilarity obtained with the DTW algorithm are compared. Best results for each classification methods are shown in bold letters.

Method	SVM	FDA_{bruto}	FDA_{mars}	LDA/PLS
Original data	0.2328 (0.0053)	0.2191 (0.0005)	0.2619 (0.0007)	0.1928 (0.0006)
Best asymmetric sim.	**0.1931 (0.0043)**	**0.2000 (0.0043)**	**0.2035 (0.0007)**	**0.1927 (0.0035)**
Relative improv.	17.53%	9.5%	22.29%	0.001%

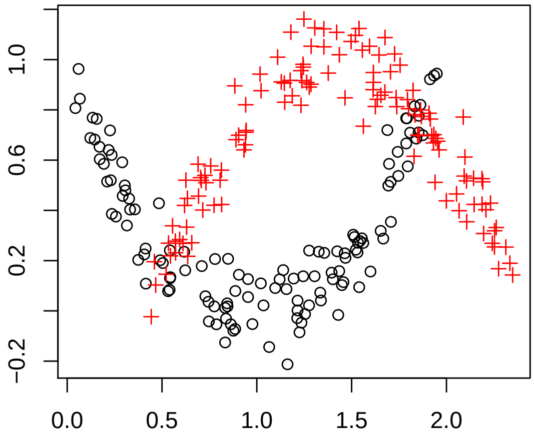


Figure 6.14: Plot of the generated data

reference matrices $\mathbf{K}^*(d)$ with the matrices $\mathbf{K}_F(d)$ and $\mathbf{K}_L(d)$ obtained applying $K_F(d)$ and $K_L(d)$ to the sample.

To this aim we use two goodness of fit measures. First we classify the test data using $\mathbf{K}_F(d)$ and $\mathbf{K}_L(d)$ and compare them with the results obtained using the reference matrix $\mathbf{K}^*(d)$. Second we measure the difference between matrices using the Frobenius norm. The results are shown in Figure 6.15. Fig. 6.15 left shows that, for every d, the kernel $K_F(d)$ obtains a quite similar performance to that obtained by the kernel it tries to reproduce (that is, $K^*(d)$). This is not the case for $K_L(d)$, a linear combination of kernels (instead a linear combination of eigenfunctions, as the $K_F(d)$ is). Thus, in a real case where we do not know in advance the results for the test data, we can expect a good behavior for $K_F(d)$. Fig. 6.15 right shows the adjustment of K_F and K_L to the

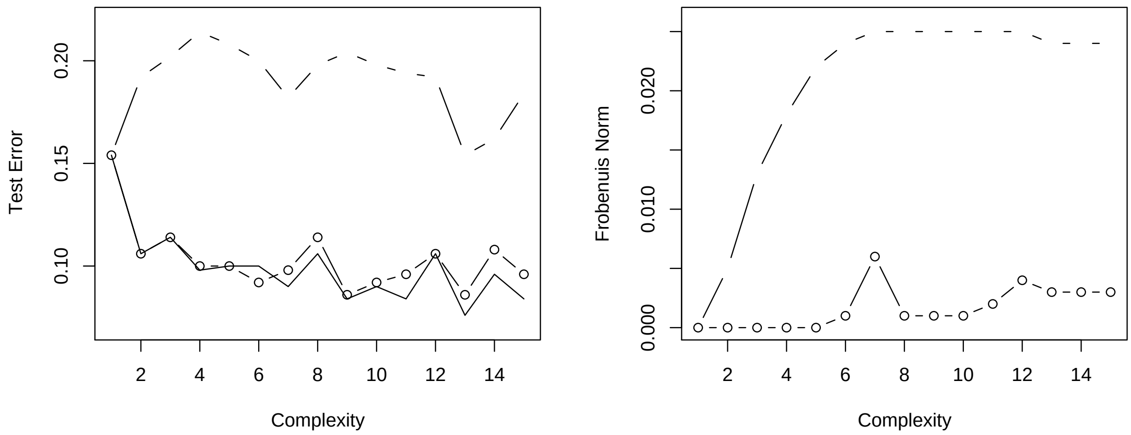


Figure 6.15: Comparison of the proposed method (line with circles) and the linear combination (dotted line) in an example of increasing complexity. Classification errors (left) and differences in terms of the frobenius norm (right) are shown.

true combination kernel as d increases. Again, $\mathbf{K}_F(d)$ remains similar to $\mathbf{K}^*(d)$.

6.7 Redundancies in kernel combinations

In this section we show the utility of the matrix similarities described in Chapter 5.2. The Body Mass Index (BMI) is a corporal index based on the weight and height of persons. It is a fast and inexpensive method for the assesment of overweight given by $weight/height^2$ (using kilograms and meters). The BMI induces the following taxonomy in human beings:

- Below 20: Underweight.
- 20-25: Normal.
- Above-30: Overweight.

In this experiment we consider a sample of 150 data with three apparent clusters. The BMI averages for each cluster are, respectively, 18, 22.5 and 28.5 (see Figure 6.16).

For the present case, six representations of the data are used. They are summarized in Table 6.14. The goal of this experiment is to compare the Pencil Dissimilarity measure (PD) and the Kernel Alignment (KA) in a case in which six linear kernels matrices

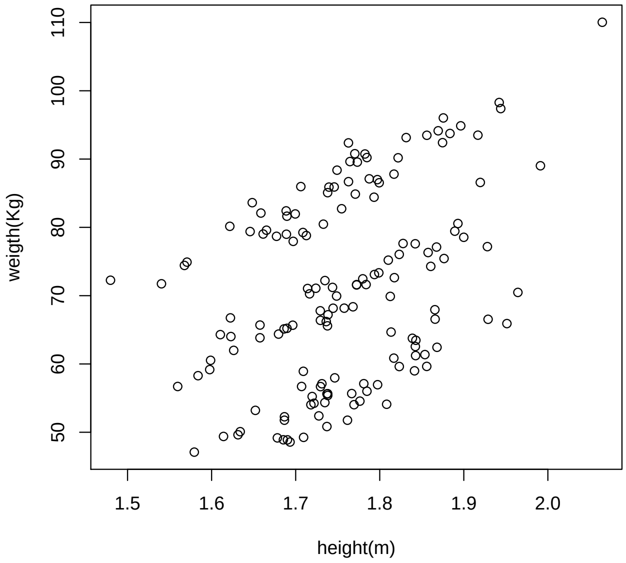


Figure 6.16: Body Mass Index example data.

Table 6.14: Six different representations of the data.

Representation	Variables	Units
1	(weigth, heigth)	kilos(normalized), meters(normalized)
2	(weigth, heigth)	kilos, meters
3	(weigth, heigth)	grams, meters
4	(weigth, heigth)	grams, centimeters
5	(weigth, heigth)	Mahalanobis transformation
6	$MBI = kg/m^2$	None

$\mathbf{K}_1, ..., \mathbf{K}_6$ are calculated according the representations of Table 6.14 (Figure 6.16 corresponds to $\mathbf{K}_2$). The example is favorable for the use of KA, because only linear transformations are involved and KA, being a correlation measure, invariant under linear transformations.

In this example, the reference kernel is assumed to be the one calculated with the BMI because it perfectly evidences the cluster structure of the data. In other words, some representations on Table 6.14 may be affected by the choice of the units, but the BMI is independent of the unit scaling. In order to estimate the similarity between the kernels $\mathbf{K}_1, ..., \mathbf{K}_5$ with $\mathbf{K}_6$, a k-means algorithm ($k = 3$) was applied to the six data representations. After clustering, the number of points that were missclassified with respect to

Table 6.15: Missclassified points for the six kernel representations with respect to the three true clusters calculated using the BMI.

Kernel	$\mathbf{K}_1$	$\mathbf{K}_2$	$\mathbf{K}_3$	$\mathbf{K}_4$	$\mathbf{K}_5$	$\mathbf{K}_6$
Errors	60	29	98	29	0	0

Table 6.16: Results for the three measures over the battery of 6 kernels with respect to K_6. Two normalization were applied to the kernels

Norm.	**Measure**	$\mathbf{K}_1$	$\mathbf{K}_2$	$\mathbf{K}_3$	$\mathbf{K}_4$	$\mathbf{K}_5$	$\mathbf{K}_6$
A	**KA**	0.961	0.973	0.940	0.973	0.993	1.000
	Procrustes	0.001	0.066	0.001	0.066	0.071	0.000
	PD	0.049	0.011	0.159	0.011	0.000	0.000
B	**KA**	0.395	0.259	0.007	0.256	0.741	1.000
	Procrustes	0.006	0.137	0.004	0.000	0.000	0.000
	PD	0.030	0.038	0.119	0.038	0.001	0.000

taxonomy induced by $\mathbf{K}_6$ is summarized in Table 6.15.

Based on Table 6.15, the ranking of kernels (regarding their similarity to $\mathbf{K}_6$), should be (begining with the most similar) $\mathbf{K}_5 \rightarrow (\mathbf{K}_4 = \mathbf{K}_2) \rightarrow \mathbf{K}_1 \rightarrow \mathbf{K}_3$ in decreasing order. Given that the measures involved in the kernels calculations are not bounded, it is convenient to perform some previous normalization. Thus we produce two standarized versions for each kernel $\mathbf{K}_i$, $i = 1, ..., 6$ given by:

$$(\mathbf{K}_i^A)_{lk} = \frac{(\mathbf{K}_i)_{lk} - min((\mathbf{K}_i))_{lk}}{max((\mathbf{K}_i))_{lk} - min((\mathbf{K}_i))_{lk}} \tag{6.2}$$

$$(\mathbf{K}_i^B)_{lk} = \frac{(\mathbf{K}_i)_{lk}}{\sqrt{(\mathbf{K}_i)_{ll}}\sqrt{(\mathbf{K}_i)_{kk}}} \tag{6.3}$$

Results of the experiment are shown in Table 6.16. It is clear that, kernel alignment and PD are equivalent. Since KA is a similarity and PD a dissimilarity, the corresponding values of the table should be interpreted in a diferent way: Kernels close to $\mathbf{K}_6$ should show large values for the alignment and small values for the PD. In contrast, the kernel procrustes measures fails in this example. It is not able to detect the relationships between the matrices being the proposed order are $(\mathbf{K}_4 = \mathbf{K}_2) \rightarrow \mathbf{K}_5 \rightarrow \mathbf{K}_3 \rightarrow \mathbf{K}_1$ (for the A normalization) and $\mathbf{K}_4 \rightarrow \mathbf{K}_2 \rightarrow \mathbf{K}_3 \rightarrow \mathbf{K}_1 \rightarrow \mathbf{K}_5$ (for B) what disagrees with the real order of the matrices.

In order to show graphically the concordance of the results for alignment and PD, Figure 6.17 shows a scatterplot of the values for each kernel. The relationship is non linear

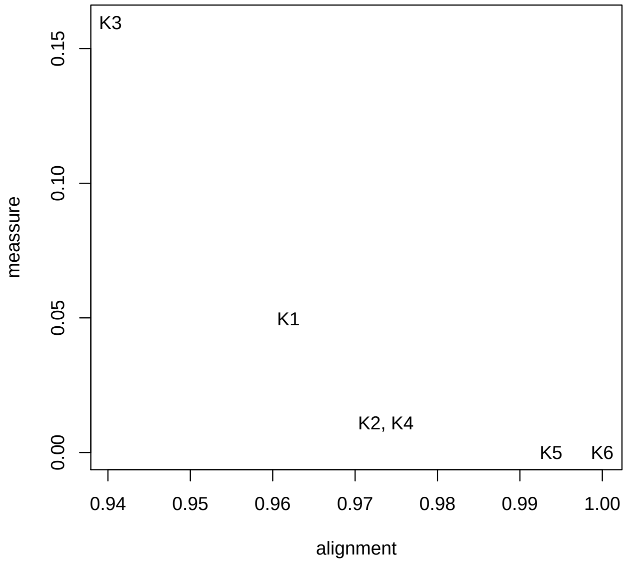


Figure 6.17: Scatterplot of the alignment and the new measure for the six kernels. This representation corresponds to first normalization method (A).

but it is always decreasing, just as one should expect for this example. Thus, we can concluded that DP performs as well as KA, the best available measure for this particular example.

6.8 The FJDA in a real example

In this example we perform a study of classification of sonar signals [3]. The goal is to discriminate between two types of signals: those bounced off a metal cylinder and those bounced off a roughly cylindrical rock. The data set has 208 observations measured on 60 variables that take values in the interval $[0, 1]$. Each value represents the energy within a particular frequency band, integrated over a certain period of time. The goal is classify the objects as rocks or mines.

We consider two Gaussian kernels $K_i(x, y) = e^{-\rho_i \|\mathbf{x}-\mathbf{y}\|^2}$, $i \in \{1, 2\}$, where $\rho_1 = 1$ and $\rho_2 = 0.1$. We want to combine K_1 and K_2 using the straightforward sum and the FJDA fusion method described in Section 5.3.2. In order to evaluate the performance of both

[3] http://www.ics.uci.edu/ mlearn/MLRepository.html

Table 6.17: Percentage of misclassified data, and percentage of support vectors for the Sonar data set after 10 runs. Standard deviations in brackets.

Kernel	**Train Error**	**Test Error**	%SV
K_1 ($\rho = 1$)	0.0114 (0.0046)	0.1595 (0.0037)	0.4000 (0.0001)
K_2 ($\rho = 0.1$)	0.0165 (0.0007)	0.2576 (0.0017)	0.4870 (0.0001)
KSum	0.0132 (0.0050)	0.1666 (0.0038)	0.7660 (0.0180)
KAJD	0.0078 (0.0040)	0.1523 (0.0040)	0.8290 (0.0220)

fusion approaches we will feed one SVM classifier with the resulting fusion kernels. The penalty value C is set to one in all the experiments.

Table 6.17 shows the classification results for the SVM classifier using four different kernels: the individual kernels K_1 and K_2, and the two fusion kernels: the straightforward sum and FJDA applied to the single kernels.

It is apparent from the results that K_1 performs better than K_2. When the straightforward sum is considered, the performance of the SVM is worse than in the case of using the Gaussian kernel with $\rho = 1$. It seems that the bad performance of K_2 damages the performance of the straightforward sum approach. On the other hand, the kernel obtained by the FJDA algorithm shows a better classification performance than the other fusion method and also than the individual kernels.

Chapter 7

Future lines of research

Next we conclude this work with the sketch of some lines of research we would like to afford in the near future.

7.1 Potential applications of functional data analysis

In fields like Chemometrics, Signal Extraction or Image Analysis, data are generally given by some measured spectra (considered as a function of the wavelength). A common problem in these areas is that the data are usually analyzed within the multivariate data analysis framework even when they violate standard assumptions.

In this thesis we have shown that the use of Regularization in Reproducing Kernel Hilbert Spaces is proven to provide appealing basis to find accurate representations of functional data. We strongly believe that problems like Regression with functional data (which is common in Chemometrics), Image Segmentation or Functional Analysis of Variance can be afforded under the approach described in Chapter 3.

7.2 Manifold Learning

Manifold Leaning has emerged as a new important topic of research with a wide range of applications. The underlying idea of manifold learning is that, while complex data often lie in very high dimensions, the number of degrees freedom is usually much less. Examples of this are human speech, data in Chemometrics or image data.

Traditional techniques such as Principal Components Analysis and Multidimensional Scaling have been extensively used for linear dimensionality reduction. However, these methods are inadequate when the data lie onto nonlinear manifolds. In recent years some new methods like Isomap (Tenenbaum et al., 2000), Locally Linear Embedding (Roweis and Saul, 2000) or and Laplacian Eigenmaps (Belkin and Niyogi, 2004) have been proposed for nonlinear dimensionality reduction tasks.

One of the key issues in manifold learning is to determine procedures able to obtain the natural distance between the objects of the sample (understanding by natural the geodesic distance calculated over the underlying manifold of the data). In this thesis we have done the first step in this sense using the theory from Reproducing Kernel Hilbert Spaces (RKHSs) to study the relationship between integral operators (associated to Mercer′s kernels) and proximity matrices. We strongly believe that this is a promising line of research we would like to investigate.

7.3 Kernel combinations and differential geometry

The performance of a kernel in classification problems depends directly on some properties that can be calculated via its associated metric (Burges, 1998). For instance, in classification problems, when data form different classes are mapped onto a manifold with high curvature, the performance of classification techniques is usually poor.

For kernels like the polynomial or Gaussian, there exist some expressions that allows works directly with the metric induced by the kernels. This property can be used to calculate the metric of kernel combinations and to optimize their parameters. We believe that an exhaustive study of kernel combinations in terms of the induced metrics is a promising line of research we would like to afford in the near future.

7.4 Spatio-temporal processes

Exploration of complex geosciences data demands the development of new and creative methods of data analysis. In the last years, strong efforts have been done in this sense specially in climate modeling. A climate model can be understood as a dynamical system in which a target variable is studied and some additional variables are considered as external forces. The state of the target variable (for instance CO_2) at a time t is represented by x_t for $t = 1, \ldots, T$. A climate model is a mapping F that relates the state of the variable from time t to $t+1$. It uses both, the current state of the climate system x_t, and the values of some external forces (location, radiation, precipitation, etc.) usually given by a vector $\mathbf{z_t}$, to produce a dynamical result for the next time step. Therefore the model states that $x_{t+1} = F(x_t, \mathbf{z_t}) + \epsilon_\mathbf{t}$ where ϵ_t is a random error (at time t). The choice of an appropriate mapping F it is crucial in climate modeling. A good model has

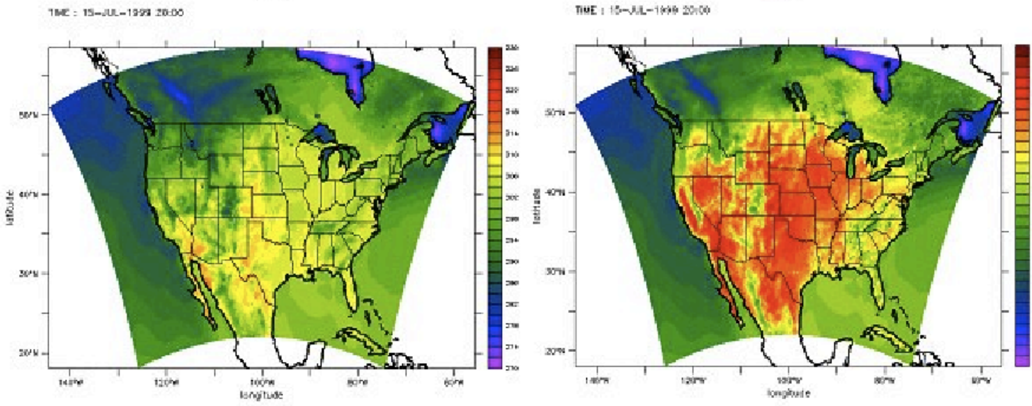


Figure 7.1: Two graphical results of the spacial estimated temperature by July 15, 1999, at 20:00 GTM. Two sources of information are available: (left) temperatures from simulation based on certain grid points (methoreologial stations); (right) temperatures from satellite assimilation data. Both results have complementary information. The left hand prediction is based on a small number of data (obtained form the stations) but generally very accurate. However databases obtained form satellites are always very large but they can be contaminated, for instance, for the presence of clouds. Source: *Development of Techniques for Assimilating GOES Satellite Data into Regional-Scale Photochemical Models. 2002, Progress Report*.

to be accurate in the description of the dynamic of the system, interpretable and fast to calculate.

7.4.1 Semi-parametric models of covariance functions

Geostatistical methods usually make the assumption that data are observations of stochastic variables. A spatial (and also temporal) variable can be considered as a realization of a random function represented by a stochastic model (estimated by F). One of the key steps in spatio-temporal modeling is to describe the statistical dependence of spatial variables and it iterations with the temporal components. Some prediction and interpolation methods, (i. e. kriging) manage these dependences assuming that the data are realizations of a Gaussian process with certain covariance function.

Although a lot work has been done, it is extremely difficult to specify realistic covariance models for complicated spatio-temporal processes. To simplify this problem, some assumptions like symmetry and separability are done. This allows to define parametric families of covariance models (Cressie and Huang, 1999) but they are inappropriate in some real examples. In this context it is of a great interest for us to investigate new tools for spatio-temporal modeling (González et al., 2009) able to deal with these and other related problems. In particular we strongly believe that this can be done under the theoretical framework the RKHS provide.

One of the main reasons why RKHSs represent a promising manner to deal with geostatistical problems is because, as we have studied in Chapters 3 and 4 of this thesis, they provide and excellent framework for information fusion. The relevance of this issue is twofold: First because it is common to find problems where several sources of information has to be combined. For instance when several collections of climate model outputs need to be ensemble to obtain a single representation that reflects the whole relevant information of the system (Sain and Furrer, 2009). See Figure 7.1 for a real example. Second, because RKHSs represent a natural way to define new semi-parametric families of combinations of covariance functions (Martín de Diego et al., 2009) that can be used in to study the dynamic complexity of atmospheric systems.

7.4.2 Extremes detection

Another important application of RKHS in climate modeling is extremes detection. While much of the work dealing with climate models concerns on interpolation or prediction (which means essentially to be focused on the means of the variables) some work has attempted to characterize extremes values (Cooley et al., 2007; Gilleland et al., 2005). In geostatistics, extremes are rare but potentially catastrophic events and their detection has several applications: identification of pollutant areas in cities, predictions of floods, heat waves, windstorms, etc.

The best-developed and most important mathematical models for rare events are based on probabilistic models usually fitted to the data using statistical techniques. However extremes are hard to measure and some open questions regarding their predictions in geostatistics remains open. For instance how does one estimates extremes where no observations are made of how it is possible to determine a possible "100 year event" from 50 years data measurements.

We stronlgy belive that RKHS can be used in conjunction with Extreme Value Theory to estimate distributional quantiles and small probabilities of appearance of extremely rare geophysical events as well as other problems in geostatistics.

References

Abney, S. (2008). Semisupervised learning for computational linguistics. *Chapman & Hall*.

Aizerman, M., Braverman, E., and Rozonoer, L. (1964). Theoretical foundations of the potential function method in pattern recognition. *Automation and Remote Control*, 25(6):821–837.

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723.

Amari, S. and Wu, S. (1999). Improving support vector machine classifiers by modifying kernel function. *Neural Networks*, 12:783–789.

Aroszajn, N. (1950). Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68(3):337–404.

Bach, F. R. and Jordan, M. I. (2002). Kernel independent component analysis. *Journal of Machine Learning Research*, 3:1–48.

Baker, C. (1977). The numerical treatment of integral equations. *Clarendon Press, Oxford*.

Bazaraa, M. S., Sherali, H. D., and Shety, C. M. (1993). Nonlinear programming: Theory and algorithms, 2nd ed. *Wiley, New York*.

Belkin, M. and Niyogi, P. (2004). Laplacian eigenmaps for dimensionality representation. *Neural Computation*, 15:1375–1396.

Boser, B., Guyon, I., and Vapnik, V. (1992). A training algorithm for optimal margin classifiers. *Proceedings of the Fifth ACM Workshop on Computational Learning Theory (COLT) ACM Press*, pages 144–152.

Boulesteix, A. L. (2004). Pls dimension reduction for classification with microarray data. *Statistical Applications in Genetics and Molecular Biology*, 3(1):article 33.

Bousquet, O. and Elisseeff, A. (2002). Stability and generalization. *Journal of Machine Learning Research*, 2:499–526.

Breiman, L., Friedman, J., Olshen, R., and Stone, C. (1984). Classification and regression trees. *Wadsworth, Belmont, CA*.

Burges, C. (1998). Geometry and invariance in kernel based methods. *Advanced in Kernel Methods-Support Vector Learning. MIT Press Cambridge, MA*, pages 89–116.

Cardoso, J. F. and Souloumiac, A. (1996). A jacobi angles for simultaneous diagonalization. *SIAM J. Mat. Anals. Applied*, 17(1):161–164.

Chapelle, O., Scholköpf, B., and Zien, A. (2006). Semi-supervised learning. *MIT Press, Cambridge, MA*.

Chen, Z. and Haykin, S. (2002). On different facets of regularization theory. *Neural Computation*, 14:2791–2846.

Cherkassky, V. and Ma, Y. (2004). Practical selection of svm parameters and noise estimation for svm regression. *Neural Networks*, 17:113–126.

Cherkassky, V., Shao, X., Mulier, F. M., and Vapnik, V. N. (1999). Model complexity control for regression using vc generalization bounds. *IEEE Transactions on Neural Networks*, 10(5):1075–1089.

Choi, M., Young, R., Nam, M.-R., and Kim, H. (2005). Fusion of multispectral and panchromatic satellite images using the curvelet transform. *Geoscience and Remote Sensing Letters*, 2(2):136–140.

Conway, F. (1990). A course in functional analysis. *Springer-Verlag*.

Cooley, D., Nychka, D., and Naveau, P. (2007). Bayesian spatial modeling of extreme precipitation return levels. *Journal of the American Statistical Association*, 102(479):824–840.

Cover, T. (1965). Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. *IEEE Transactions on Electronic Computers EC*, 14:326–334.

Cox, D. and O'Sullivan, F. (1990). Asymptotic analysis and penalized likelihood and related estimators. *Annals of Statistics*, 18:1676–1695.

Cox, T. and Cox, M. (2001). *Multidimensional Scaling*. Chapman & Hall.

Craven, P. and Wahba, G. (1979). Smoothing noisy data with spline functions: Estimating the correct degree of smoothing by the method of generalized cross validation. *Numerische Mathemetik*, 31:377–403.

Cressie, N. and Huang, H.-C. (1999). Classes of nonseparable, spatio-temporal stationary covariance functions. *Journal of the American Statistical Association*, 94:1330–1340.

Cristianini, N. and Shawe-Taylor, J. (2000). An introduction to support vector machines. *Cambridge University Press*.

Cristianini, N. and Shawe-Taylor, J. (2002). On the kernel target alignment. *Journal of Machine Learning Research*, 5:27–72.

Cucker, F. and Smale, S. (2001). On the mathematical foundations of learning. *Bulletin of the American Mathematical Society*, 39(1):1–49.

Cucker, F. and Zhou, D. X. (2007). *Statistical Learning*. Cambridge University Press.

de Jong, S. (1993). Simpls: an alternative approach to partial least squares regression. *Chemometrics and Intelligent Laboratory Systems*, 18:251–263.

Deutsch, F. R. (2001). Best approximation in inner product spaces. *Springer. CMS Books in Mathematics*.

Epifanio, I., Gutierrez, J., and Malo, J. (2003). Linear transform for simultaneous diagonalization of covariance and perceptual metric matrix in image coding. *Pattern Recognition*, 36:799–1811.

Ferraty, F. and Vieu, P. (2003). Curves discrimination: a nonparametric functional approach. *Computational Statistics & Data Analysis*, 44:161–173.

Ferraty, F. and Vieu, P. (2006). *Nonparametric Functional Data Analysis*. Springer.

Frey, P. W. and Slate, D. J. (1991). Letter recognition using holland-style adaptive classifiers. *Machine Learning*, 6(2):161–182.

Friedman, J. (1991). Multivariate adaptive regression splines. *Annals of Statistics*, 19:11–41.

Gasca and Sauer (2000). Multivariate polynomial interpolation. *Advances in Computational Mathematics*, 12:377–410.

Gilleland, E., Nychka, D., and Schneider, U. (2005). Spatial models for the distribution of extremes. *In Applications of Computational Statistics in the Environmental Sciences:*

Hierarchical Bayes and MCMC Methods ed. J.S. Clark and A. Gelfand, Oxford University Press.

Golub, G. and Loan, C. (1997). Matrix computations. *University Press, Baltimore.*

González, J., Sain, S. R., and Muñoz, A. (2009). Spatial temporal data analysis via reproducing kernel regularization. *Joint Statistical Meeting (JSM), Washington, USA.*

Gower, J. C. (1986). Metric and euclidean properties of dissimilarities coefficients. *Journal of Classification*, 3:5–48.

Gower, J. C. (2000). Multivariate polynomial interpolation. *Advances in Computational Mathematics*, 12:377–410.

Hall, P. and Vial, C. (2006). Assessing the finite dimensionality of functional data. *Journal of the Royal Statistical Society: Series B*, 68(4):689–705.

Hastie, T., Buja, A., and Tibshirani, R. (1994). Flexible discriminant analysis by optimal scoring. *Journal of the American Statistical Association*, 89:1255–1270.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). The elements of statistical learning: Data mining, inference, and prediction. second edition. *Springer Series in Statistics.*

Higham, N. J. (1986). Computing the polar decomposition with applications. *SIAM J. Sci. Statist. Comput.*, 7:1160–1174.

Higham, N. J. (2002). Computing the nearest correlation matrix. a problem from finance. *IMA Journal of Numerical Analysis*, 22:329–343.

Hochstadt, H. (1973). Integral equations. *John Wiley and Sons. New York*, 52:11503.

Horn, R. A. and Johnson, C. R. (1991). Topics in matrix analysis. *Cambridge University Press.*

Hotelling, H. (1936). Relation between two sets of variables. *Biometrika*, 28:321–377.

Hua, Y. (1991). On svd estimating generalized eigenvalues of singular matrix pencils in noise. *IEEE Transactions on Signal Processing*, 39(4):892–900.

Hyvärinen, A., Karhunen, J., and Oja, E. (2001). *Independent Component Analysis*. New York: Wiley.

Ivanov, V. (1976). The theory of approximate methods and their application to the numerical solution of singular integral equations. *Noordhoff International, Leyden.*

James, G. M. and Sugar, C. A. (2003). Clustering for sparsely sampled functional data. *Journal of the American Statistical Association*, 98:397–408.

Joachims, T. (2002). Learning to clasify text uning support vector machines. *Kluver*.

Jolliffe, I. (2002). *Principal Component Analysis*. Series: Springer Series in Statistics, 2nd ed.

Keerthi, S. S. and Lin, C.-J. (2003). Asymptotic behaviors of support vectormachines with gaussian kernel. *Neural Computation*, 15:1667–1689.

Kimeldorf, G. S. and Wahba, G. (1970). A correspondence between bayesian estimation on stochastic processes and smoothing by splines. *The Annals of Mathematical Statistics*, 41(2):495–502.

Kittler, J., Hatef, M., Duin, R. P. W., and Matas, J. (1998). On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine intelligence*, 20(3):226–239.

Kosko, B. (1991). Neural networks and fuzzy systems: A dynamical approach to machine intelligence. *Prentice Hall*.

Lanckriet, G., Cristianini, N., Bartlett, P.and El Ghaoui, L., and Jordan, M. (2004). Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, 5:27–72.

Lang, K. (1995). Newsweeder: Learning to filter netnews. *Proceedings of the Twelfth International Conference on Machine Learning*, pages 331–339.

Lin, Y. (2002). Support vector machines and the bayes rule in classification. *Data Mining Knowledge Discovery*, 6:259275.

Lorentz, R. A. (2000). Multivariate hermite interpolation by algebaic polynomials: a survey. *Journal of computational and applied mathematics*, 1:167 – 201.

Luenberger, D. G. (1969). Optimization by vector space methods. *New York: Wiley*.

Mallows, C. L. (1973). Some comments on cp. *Technometrics*, 15(4):661–675.

Mangasarian, O. and Wolberg, W. (1990). Cancer diagnosis via linear programming. *SIAM News*, 23 (5):1–18.

Martín de Diego, I., Muñoz, A., and Moguerza, J. M. (2009). Methods for the combination of kernel matrices within a support vector framework. *Machine Learning*, 78:137–174.

Martín de Diego, I. M. and Muñoz, A. (2006). Kernel procrustes. *Springer-Verlag, Lecture Notes in Computer Science*, 4:237–240.

Martín-Merino, M. and Muñoz, A. (2005). Visualizing asymmetric proximities with som and mds models. *Neurocomputing*, 63:171–192.

Marx, B. and Eilers, P. (1999). Generalized linear regression on sampled signals and curves: a p-spline approach. *Technometrics*, 41:1–13.

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

Moguerza, J. and Muñoz, A. (2006). Support vector machines with applications. *Statistical Science*, 21(4):322–336.

Muñoz, A. and González, J. (2008). Functional learning of kernels for information fusion purposes. *Springer-Verlag, Lecture Notes in Computer Science*, 5197:277–283.

Muñoz, A., González, J., and deDiego, I. M. (2006). Local linear approximation for kernel methods: The railway kernel. *Springer-Verlag, Lecture Notes in Computer Science*, 4225:936–944.

Muñoz, A. and Martín de Diego, I. (2006). From indefinite to semi-definite matrices. *Springer-Verlag, Lecture Notes in Computer Science*, 4109:764–772.

Muñoz, A. and Moguerza, J. (2006). Estimation of high density regions with one class neighbor machines. *IEEE Pattern Analysis and Machine Intelligence*, 28(3):476–480.

Mukherjee, S., Rifkin, P., and Poggio, T. (2002). Regression and classification with regularization. *Nonlinear Estimation and Classification', Lecture Notes in Statistics*, 171:107–124.

Omladic, M. and Semrl, P. (1990). On the distance between normal matrices. *Proceedings of the American Mathematical Society*, 110:591–596.

O'Sullivan, F. (1986). A statistical perspective on ill-posed inverse problems (with discussion). *Statistical Science*, 1:502–527.

Osuna, E., Freund, R., and Girosi, F. (1997). An improved training algorithm for support vector machines. *In Proc. IEEE Workshop on Neural Networks for Signal Processing. IEEE Press, New York.*

Parlett, N. B. (1997). The symmetric eigenvalue problem. *Classics in Applied Mathematics, SIAM.*

Parzen, E. (1970). Statistical inference on time series by rkhs methods. *In Proceedings 12th Biennial Seminar (R. Pyke, ed.) Canadian Mathematical Congress, Montreal.*, pages 1–37.

Philips, D. L. (1962). A technique for the numerical solution of certain integral equations of the first kind. *Journal of the Association for Computing Machinery*, 9:84–97.

Platt, J. C. (1999). Fast training of support vector machines using sequential minimal optimization. *Advanced in Kernel Methods-Support Vector Learning. MIT Press Cambridge, MA*, pages 41–66.

Poggio, T. and Gorosi, F. (1998). Notes on pca, regularization, sparsity and support vector machines. *Technical Report. A. I Memo No. 1632. C.B.C.L Paper No 161.*

Poggio, T., Mukherjee, S., Rifkin, R., Rakhlin, A., and A., V. (2001). b. *CBCL Paper 198/AI Memo 2001/011 Massachussets Institute of technology.*

Rakotomamonjy, A. and Canu, S. (2005). Frames, reproducing kernels, regularization and learning. *Journal of Machine Learning Research*, 6:1485–1515.

Ramsay, J. O. and Silverman, B. W. (2006). Functional data analysis. *Springer, New York, 2nd ed.*

Roweis, S. and Saul, L. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326.

Sain, S. and Furrer, R. (2009). A proposal for combining climate model output. *Working Paper*.

Sakoe, H. and Chiba, S. (1978). Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 26(1):43–49.

Schölkopf, B., Herbrich, R., Smola, A., and Williamson, R. (2000). A generalized representer theorem. *Springer-Verlag, Lecture Notes in Computer Science*, 2111.

Schölkopf, B., Smola, A., and Muller, K. R. (1999). Kernel principal component analysis. *Advanced in Kernel Methods-Support Vector Learning. MIT Press Cambridge, MA*, pages 327–352.

Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics*, 6:461–464.

Sigillito, V. G., Wing, S. P., Hutton, L. V., and Baker, K. B. (1989). Classification of radar returns from the ionosphere using neural networks. *Johns Hopkins APL Technical Digest*, 10:262–266.

Smale, S. and Zhou, D. X. (2007). Geometry on probability spaces. *Working Paper*.

Smola, A. and Schölkopf, B. (2003). A tutorial on support vector regression. *NeuroCOLT Technical Report TR-98-030*.

Spellman, P. T., Sherlock, G., Zhang, M. Q., Iyer, V. R., Anders, K., Eisen, M. B., Brown, P. O., Bostein, D., and Futcher, B. (1998). Comprehensive identification of cell cycle-regulated genes of the yeast saccharomyces cerevisiae by microarray hybridization. *Molecular Biology of the Cell*, 9:3273– 3297.

Sugiura, N. (1978). Further analysis of the data by akaikes information criterion and the finite corrections. *Communications in Statistics. Theory and Methods*, 7(1):13–26.

Sugiyama, M. and Muller, K. R. (2002). The subspace information criterion for infite dimensional hypothesis spaces. *Journal of Machine Learning Research*, 3:323–359.

Sugiyama, M. and Ogawa, H. (2001). Subspace information criterion for model selection. *Neural Computation*, 13(8):1863–1889.

Sugiyama, M. and Ogawa, H. (2002). Theoretical and experimental evaluation of the subspace information criterion. *Machine Learning*, 48:25–50.

Swindel, F. B. (1981). Geometry of ridge regression illustrated. *The American Statistician*, 35(1):12–15.

Tarantola, A. (2005). *Inverse Problem Theory and Methods for Model Parameter Estimation, 2nd edition*.

Tenenbaum, J. B., de Silva, V., and Langford, V. J. (2000). A global geometric framework for nonlinearity dimensionality reduction. *Science*, 290:2319–2323.

Tikhonov, A. and Arsenin, V. (1977). Solutions of ill-posed problems. *John Wiley and Sons, New York*.

Ua, P. C. and Pozdnoukhov, A. (2002). The analysis of kernel ridge regression learning algorithm. *Working Paper*.

Vapnik, V. (1995). The nature of statistical learning theory. *Springer, New York*.

Vapnik, V. (1998). Statistical learning theory. *John Wiley and Sons, New York*.

Vapnik, V. and Chervonenkis, A. (1964). A note on a class of perceptrons. *Automation and Remote Control*, 25:103–109.

Wahba, G. (1990). Spline models for observational data. *Series in Applied Mathematics, SIAM. Philadelphia*.

Wahba, G. (2003). Reproducing kernel hilbert spaces - two brief reviews. *Proceedings of the 13th IFAC Symposium on System Identification*, pages 549–559.

Wahba, G. (2006). Comment to "svm with applications". *Statistical Science*, 21(3):347–351.

Wax, M. and Sheinvald, J. (1997). A least-squares approach to joint diagonalization. *IEEE Signal Processing Lett*, 4:52–53.

Yeredor, A. (2002). Non-orthogonal joint diagonalization in the least-squares sense with application in blind source separation. *IEEE Transactions on Signal Processing*, 50 (7):1545–1553.

Zolteoski, M. (1988). Solving the generalize eigenvalue problem with singular forms. *Proceedings of EEE ICASSP*, 4:2861–2864.