Variable Weight Kernel Density Estimation

by

Efrén N. Cruz Cortés

**A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical Engineering: Systems)
in the University of Michigan
2017**

Doctoral Committee:

Associate Professor Clayton D. Scott, Chair
Assistant Professor Laura K. Balzano
Associate Professor XuanLong Nguyen
Professor Jun Zhang

Efrén Cruz Cortés

encc@umich.edu

ORCID iD: 0000-0002-2062-6444

Dedication

To the proletarians of all countries.

Acknowledgments

I want to thank all people who oppose and have opposed sexism, racism, capitalism, and all forms of oppression. I acknowledge that if I was able to do anything is because of their commitment and sacrifice. We will all keep the struggle. I thank all the workers that facilitated the material means of my research and I hope to repay the debt one way or another. I also thank the psychological support from my friends and mentors. Finally, I thank my adviser Professor Clay Scott for his guidance.

Preface

No scientific endeavor is free of bias, pitfalls, and unintended consequences, nor is it free of a history. Unfortunately, engineering has a dark history of workers' exploitation, resource extraction, dispossession, population control, militaristic invasions, etc. As such, working in this field should be done with care and with the explicit determination to end these practices. Machine Learning, in specific, has been used recently to target the poor and the racially oppressed to further exploitation and reproduce Machiavellian power dynamics, as well as economic inequality. I must responsibly address this issue. My research is intended to aid, among others, medical, environmental, and economic problems for the betterment of society, it is NOT intended to aid parasitic financial institutions, murderous military activity, discriminatory oppression by states, or similar malpractices. I strongly condemn any misuse of my research, be it by individuals, governments, powerful corporations, or the like.

Contents

List of Figures

List of Tables

Abstract

Nonparametric density estimation is a common and important task in many problems in machine learning. It consists in estimating a density function from available observations without making parametric assumptions on the generating distribution. Kernel means are nonparametric estimators composed of the average of simple functions, called kernels, centered at each data point. This work studies some relatives of these kernel means with structural similarity but which assign different weights to each kernel unit in order to attain certain desired characteristics. In particular, we present a sparse kernel mean estimator and a consistent kernel density estimator with fixed bandwidth parameter.

First, regarding kernel means, we study the kernel density estimator (KDE) and the kernel mean embedding. These are frequently used to represent probability distributions, unfortunately, they face scalability issues. A single point evaluation of the kernel density estimator, for example, requires a computation time linear in the training sample size. To address this challenge, we present a method to efficiently construct a sparse approximation of a kernel mean. We do so by first establishing an incoherence-based bound on the approximation error. We then observe that, for any kernel with constant norm (which includes all translation invariant kernels), the bound can be efficiently minimized by solving the k-center problem. The outcome is a linear time construction of a sparse kernel mean, which also lends itself naturally to an automatic sparsity selection scheme. We demonstrate the computational gains of our method by looking at several benchmark data sets, as well as three applications involving kernel means: Euclidean embedding of distributions, class proportion estimation, and clustering using the mean-shift algorithm.

Second we address the bandwidth selection problem in kernel density estimation. Consistency of the KDE requires that the kernel bandwidth tends to zero as the sample size grows. In this work, we investigate the question of whether consistency is still possible when the bandwidth is fixed, if we consider a more general class of *weighted* KDEs. To answer this question in the affirmative, we introduce the fixed-bandwidth KDE (fbKDE), obtained by solving a quadratic program, that consistently estimates any continuous square-integrable density. Rates of convergence are also established for the fbKDE for radial kernels and the box kernel under appropriate smoothness assumptions. Furthermore, in a simulation study we demonstrate that the fbKDE compares favorably to the standard KDE and the previously proposed variable bandwidth KDE.

Chapter 1

Introduction

1.1 Kernel Density Estimator

Given an iid sample $x_1, \ldots, x_n \in \mathcal{X} \subset \mathbb{R}^d$ drawn from a probability distribution with associated density function f, the kernel density estimator (KDE) is an estimator for f given as

$$f_{KDE} = \frac{1}{n} \sum_{i=1}^{n} k(\cdot, x_i), \tag{1.1}$$

where k is a kernel function with parameter σ. Shortly we will define a kernel function in detail and provide some examples. An important and recurrent example is the Gaussian kernel, defined as

$$k(x, x') = (2\pi\sigma^2)^{-d/2} \exp(-\|x - x'\|^2/2\sigma^2).$$

The birth of the kernel density estimator was presaged by the unpublished report of Evelyn Fix [1] on nonparametric density estimators. The report emphasized the importance of these estimators for discriminant analysis (used for plug-in rules) and outlined their desired consistency properties. A few years later a full development of the KDE followed by Rosenblatt and Parzen [2, 3]. Since then, the KDE has found numerous applications across a broad range of quantitative fields and it has become a key ingredient in many machine learning methodologies. For example, a common approach to classification is a plug-in rule that estimates the class-conditional densities with separate KDEs [4, 5]. In anomaly detection, a detector of the form $f_{KDE}(x) \gtrless \gamma$ is commonly employed to determine if a new realization comes from f [6, 7, 8, 9]. In clustering, the mean-shift algorithm forms a KDE and associates each data point to the mode of the KDE that is reached by hill-climbing [10]. Parallel to its many applications, the KDE has also been the subject of extensive theoretical investigations, spawning several books (see, e.g, [11, 12, 13, 14]) and hundreds of research articles.

Although the KDE is a simple and mathematically founded nonparametric density estimator, two main drawbacks make its use sometimes undesirable. First, it is not scalable. As seen in equation (1.1), a point evaluation of the KDE requires $O(n)$ computations, where n is the size of the data. Evaluating the KDE just on the original points already requires $O(n^2)$ computations, hence an efficient way of computing or approximating the KDE is necessary. In this work, we present a sparse approximation to the KDE which can be computed efficiently. Second, each kernel has a smoothing or "bandwidth" parameter, the selection of which turns out to be more important than the selection of the kernel

itself [13]. Automatically selecting a bandwidth parameter is a hard task and the subject of much research [15]. In order for the KDE to be consistent, the bandwidth is required to approach zero. We address this issue not by presenting a new way of finding the smoothing parameter but by proposing a consistent kernel density estimator with fixed bandwidth. Note that both of the estimators we present are linear combinations of kernel functions, just as the KDE, but the weights of each function differ, contrary to the KDE whose weights are uniform.

1.2 Kernels

In this section we define different types of kernels. In the context of kernel density estimation, the kernel function satisfies that for all x', $\int k(x, x')dx = 1$. In addition, k is sometimes also chosen to be nonnegative, although this is not necessary for theoretical properties such as consistency. In Chapter 2 we will be concerned with a kernel sum more general than the KDE. We will be required, however, to work in inner product spaces. This motivates the following general definition.

Definition 1. *Let $\mathcal{X} \subset \mathbb{R}^d$. We say that $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a* kernel *if there exists an inner product space $\mathcal{H}$ such that for all x in $\mathcal{X}$, $k(\cdot, x) \in \mathcal{H}$.*

In the case of kernel density estimation, all commonly used kernels satisfy $k(\cdot, x) \in L^2(\mathbb{R}^d)$ for all $x \in \mathbb{R}^d$. Here, $L^2(\mathbb{R}^d)$ is the space of equivalence classes of square integrable functions. When we write $k(\cdot, x) \in L^2(\mathbb{R}^d)$, we view $k(\cdot, x)$ as a representative of its equivalence class. In the case of a positive definite kernel (defined below), $\mathcal{H}$ can be the associated reproducing kernel Hilbert space (defined in Section 1.5).

Definition 2. *For $\boldsymbol{x} := \{x_1, \ldots, x_n\}$ a subset of $\mathbb{R}^d$ and k a kernel with associated inner product space $\mathcal{H}$, we call*

$$K_{\boldsymbol{x}} := \left(\langle k(\cdot, x_i), k(\cdot, x_j)\rangle_{\mathcal{H}}\right)_{i,j\in[n]}$$

the kernel matrix of k.

We have used the notation $[n] := \{1, \ldots, n\}$. When the set $\boldsymbol{x}$ is understood, it is convenient to use the notation K instead of $K_{\boldsymbol{x}}$. When no ambiguity arises, for $\mathcal{I} \subset [n]$, the kernel matrix of the set $\{x_i \in \boldsymbol{x} | i \in \mathcal{I}\}$ is denoted as $K_{\mathcal{I}}$.

Definition 3. *A* symmetric positive definite kernel *is a kernel that is symmetric and for which the matrix $K_{\boldsymbol{x}}$ is positive semidefinite for all finite subsets $\boldsymbol{x} \subset \mathcal{X}$.*

For convenience we will sometimes refer to symmetric positive definite kernels as PD kernels or, for reasons which will become clear later, as reproducing kernels. Every symmetric positive definite kernel is associated to a unique Hilbert space of functions called a reproducing kernel Hilbert space (RKHS). Further properties of PD kernels and their RKHS's will be discussed in Section 1.5.

Although some of our results hold more generally, we are interested in a particular class of kernels, the radial kernels, which we now define.

Definition 4. *We say $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a* radial kernel *if k is a kernel as in Def. 7 and there exists a strictly decreasing function $g : [0, \infty) \to \mathbb{R}$ such that, for all $x, x' \in \mathbb{R}^d$,*

$$\langle k(\cdot, x), (\cdot, x')\rangle_{\mathcal{H}} = g(\|x - x'\|_2).$$

We now review some common examples of kernels. The Gaussian kernel with parameter $\sigma > 0$ has the form

$$k(x, x') = c_\sigma \exp\left(-\frac{\|x - x'\|_2^2}{2\sigma^2}\right).$$

The Laplacian kernel with parameter $\gamma > 0$ has the form

$$k(x, x') = c_\gamma \exp\left(-\frac{\|x - x'\|_2}{\gamma}\right).$$

The Student-type kernel with parameters $\alpha, \beta > 0$ has the form

$$k(x, x') = c_{\alpha,\beta}\left(1 + \frac{\|x - x'\|_2^2}{\beta}\right)^{-\alpha}.$$

The uniform kernel with parameter u has the form

$$k(x, x') = c_u \mathbf{1}_{\{\|x-x'\|_2 < u\}}$$

The Epanechnikov kernel with parameter a has the form

$$k(x, x') = c_a\left(1 - \frac{\|x - x'\|_2^2}{a^2}\right)\mathbf{1}_{\{\|x-x'\|_2 < a\}}.$$

The parameters $c_\sigma, c_\gamma, c_{\alpha,\beta}, c_u$ and c_a can be set, if desired and depending on the application, so as to normalize k to be a density estimation kernel.

These examples illustrate that the space $\mathcal{H}$ such that $k(\cdot, x) \in \mathcal{H}$ is not unique. Indeed, on one hand we may select $\mathcal{H} = L^2(\mathbb{R}^d)$. On the other hand, the first three kernels are symmetric positive definite kernels, and therefore we may take $\mathcal{H}$ to be the RKHS associated with k [16, 17]. Note that the last two kernels do not fit our definition of a radial kernel.

Each of the three PD kernel examples is also a radial kernel. If we take $\mathcal{H}$ to be the RKHS, then by the reproducing property we simply have $\langle k(\cdot, x), k(\cdot, x')\rangle = k(x, x')$, and in each case, $k(x, x') = g(\|x - x'\|_2)$ for some strictly decreasing g. These three kernels are also radial if we take $\mathcal{H} = L^2(\mathbb{R}^d)$. For example, consider the Gaussian kernel, and let us write $k = k_\sigma$ to indicate the dependence on the bandwidth parameter. Then $\langle k_\sigma(\cdot, x), k_\sigma(\cdot, x')\rangle_{L^2} = k_{\sqrt{2}\sigma}(x, x')$. Similarly, for the Student kernel with $\alpha = (1 + d)/2$ (the Cauchy kernel), we have $\langle k_\beta(\cdot, x), k_\beta(\cdot, x')\rangle_{L^2} = k_{2\beta}(x, x')$. For other kernels, although there may not be a closed form expression for g, it can still be argued that such a g exists, which is all we will need.

1.3 Efficient Computation

A quick note on notation. Throughout this section and in Chapter 2 we denote a kernel function by ϕ and reserve the letter k for an integer. Also, the results of Chapter 2 are applicable to another type of kernel sum of the form (1.1), the empirical kernel mean embedding. Therefore we refer to both the KDE and the empirical kernel mean embedding as kernel means.

As mentioned before, evaluating the KDE at a single test point requires $O(n)$ kernel evaluations, which for some applications is undesirable and perhaps computationally prohibitive. Several approaches have been taken to address this issue, many of which we review in Chapter 2. In this work we are concerned about the efficient computation of a sparse approximation of a KDE taking the form

$$\sum_{i=1}^{n} \alpha_i \phi(\cdot, x_i) \tag{1.2}$$

where $\alpha_i \in \mathbb{R}$ and $k := |\{i : \alpha_i \neq 0\}| \ll n$. In other words, given $x_1, \ldots, x_n$, a kernel ϕ, and a target sparsity k, we seek a sparse linear combination of kernels (2.2) that accurately approximates the KDE (1.1). A sparse kernel sum can be evaluated or manipulated much more efficiently. In the large n regime, the sparse approximation algorithm itself must be scalable, and as we argue in Chapter 2, existing sparse approximation strategies are too slow.

In this work we propose an efficient algorithm for sparsely approximating a kernel mean. This algorithm results from minimizing an "incoherence" based bound and holds for both types of kernel means. Several experiments show the advantage of our approach.

We reiterate that point evaluations of the KDE require $O(n)$ kernel evaluations, which is prohibitive for large n. On the other hand, a sparse approximation with sparsity k requires only $O(k)$ kernel evaluations. In some instances we need to evaluate the KDE multiple times for each query point, enhancing the problem further.

1.4 Consistent KDE with Fixed Bandwidth

One reason for the KDE's success is that it is nonparametric and makes few if any assumptions about f. Furthermore, the KDE is consistent, meaning that it converges to f as $n \to \infty$ [18]. To prove that the KDE is consistent it will be necessary for the convolution of f and k to approach f under some norm, which in turns requires the smoothing parameter σ to approach 0.

Correctly choosing the bandwidth is critical for practical applications of the KDE. Indeed, choice of bandwidth is the primary factor that determines the performance of the KDE, and is much more important than the choice of kernel [13]. At the same time, bandwidth selection is a notoriously difficult problem, indeed, a recent survey cites a couple dozen bandwidth selection methods [15].

In this work we do not propose a method of bandwidth selection. Instead, in Chapter 4 we aim at addressing the criticality of finding the optimal bandwidth by presenting an estimator different to the KDE which is consistent for a fixed bandwidth. The estimator takes the form

$$f_\alpha = \sum_{i=1}^{n} \alpha_i k(\cdot, z_i),$$

where $\{z_i\}_{i \in [n]}$ are not necessarily the data points $\{X_i\}_{i=1}^n$ and $\alpha \in A_n$, A_n being an ℓ_1-ball of data dependent size. Note that contrary to the KDE the α coefficients are not required to be nonnegative nor to sum to 1.

1.5 Reproducing Kernels

In this section we list some useful properties of positive definite kernels and their associated reproducing kernel Hilbert spaces. We start by recalling the definition of a positive definite kernel.

Definition A *positive definite kernel* is a kernel that is symmetric and for which the matrix $K_{\boldsymbol{x}}$ is positive semidefinite for all finite subsets $\boldsymbol{x} \subset \mathcal{X}$.

We have been mentioning the so called reproducing kernel Hilbert space. We now define it (see [19, 16]).

Definition 5. *Let $\mathcal{X}$ be a set. We call a Hilbert space of real-valued functions $\mathcal{H}$ a* reproducing kernel Hilbert space *(RKHS) on $\mathcal{X}$ if for every $x \in \mathcal{X}$ the linear evaluation functional $\delta_x : \mathcal{H} \to \mathbb{R}$ defined as $\delta_x(f) = f(x)$ for $f \in \mathcal{H}$, is bounded (equivalently, continuous).*

Recall that k is a kernel of an inner product space $\mathcal{H}$ if $k(\cdot, x) \in \mathcal{H}$ for all $x \in \mathcal{H}$. In that case we have

Definition 6. *A kernel of $\mathcal{H}$ is called a* reproducing kernel *if the reproducing property*

$$f(x) = \langle f, k(\cdot, x) \rangle_{\mathcal{H}}$$

holds for all $f \in \mathcal{H}$ and all $x \in \mathcal{X}$.

The following properties will be used throughout the book:

- Every symmetric positive definite kernel is a reproducing kernel, and vice versa.
- Every reproducing kernel has a unique associated RKHS, and vice versa.
- Given a PD kernel k on $\mathcal{X}$ and its associated RKHS $\mathcal{H}$, the set

$$H^0 := \left\{ \sum_{i=1}^{n} c_i k(\cdot, x_i) \mid (c_i, x_i) \in \mathbb{R} \times \mathcal{X} \ \forall\, i \in [n], n \in \mathbb{N} \right\}$$

  is dense in $\mathcal{H}$, under the norm induced by $\langle \cdot, \cdot \rangle_{\mathcal{H}}$.

These properties are stated without proof, and they will be used throughout without further justification. For more details see [19, 16, 20].

Chapter 2

Sparse Approximation of a Kernel Mean

Overview

In this chapter we present a scalable sparse representation of kernel means. It is scalable in that it can be efficiently computed is linear time with respect to both the sample size and the dimension. To construct it we first bound the approximation error and observe that for constant norm kernels this bound can be minimized by solving the k-center problem. We then demonstrate experimentally that our method compares favorably to other sparse approximation methods.

2.1 Introduction

A *kernel mean* is a quantity of the form

$$\frac{1}{n}\sum_{i=1}^{n}\phi(\cdot, x_i), \tag{2.1}$$

where ϕ is a *kernel* and $x_1, \ldots, x_n \in \mathbb{R}^d$ are data points. We define kernels rigorously below. Our treatment includes many common examples of kernels, such as the Gaussian kernel, and encompasses both symmetric positive definite kernels and kernels used for nonparametric density estimation.

Kernel means arise frequently in machine learning and nonparametric statistics as representations of probability distributions. In this context, $x_1, \ldots, x_n$ are understood to be realizations of some unknown probability distribution. The kernel density estimator (KDE) is a kernel mean that estimates the density of the data. The kernel mean embedding (KME) is a kernel mean that maps the probability distribution into a reproducing kernel Hilbert space. These two motivating applications of kernel means are reviewed in more detail below.

This work is concerned with efficient computation of a sparse approximation of a kernel mean, taking the form

$$\sum_{i=1}^{n}\alpha_i\phi(\cdot, x_i) \tag{2.2}$$

where $\alpha_i \in \mathbb{R}$ and $k := |\{i : \alpha_i \neq 0\}| \ll n$. In other words, given $x_1, \ldots, x_n$, a kernel ϕ, and a target sparsity k, we seek a sparse kernel mean (2.2) that accurately approximates

the kernel mean (2.1). This problem is motivated by applications where n is so large that evaluation or manipulation of the full kernel mean is computationally prohibitive. A sparse kernel mean can be evaluated or manipulated much more efficiently. In the large n regime, the sparse approximation algorithm itself must be scalable, and as we argue below, existing sparse approximation strategies are too slow.

Our primary contribution is an efficient algorithm for sparsely approximating a kernel mean. The algorithm results from minimizing a sparse approximation bound based on a novel notion of incoherence. We show that for a broad class of kernels minimizing the sparse approximation bound is equivalent to solving the k-center problem on $x_1, \ldots, x_n$, which in turn leads to an efficient algorithm. An advantage of our approach is that it approximates an arbitrary kernel mean, so we need not address the KDE and KME problems independently, but through a shared methodology.

The rest of the paper is outlined as follows. In Section 2.2 we review the KDE and KME, which motivate this work, and also introduce a general definition of kernel that encompasses both of these settings. We review related work and its connection to our contributions in Section 2.3. In Section 2.4 we establish an incoherence-based sparse approximation bound. We then use the principle of bound minimization in Section 2.5 to derive a scalable algorithm for sparse approximation of kernel means, and present a sparsity auto-selection scheme. Finally, to demonstrate the efficacy of our approach, Section 2.6 applies our methodology in three different machine learning problems that rely on large-scale KDEs and KMEs, and also presents its performance on 11 benchmark datasets. A preliminary version of this work appeared in [21]. A Matlab implementation of our algorithm is available at [22].

2.2 Motivation and Formal Setting

Our work is motivated by two primary examples of kernel means. We review the KDE and KME separately, and then propose a general notion of kernel that encompasses the essential features of both settings and is sufficient for addressing the sparse approximation problem. By way of notation, we denote $[n] := \{1, \ldots, n\}$.

2.2.1 Kernel Density Estimation

Let $\{x_1, \ldots, x_n\} \subset \mathbb{R}^d$ be a random sample from a distribution with density f. In the context of kernel density estimation, a kernel is a function ϕ such that for all x', $\int \phi(x, x')dx = 1$. In addition, ϕ is sometimes also chosen to be nonnegative, although this is not necessary for theoretical properties such as consistency. The kernel density estimator of f is the function

$$\widehat{f} = \frac{1}{n} \sum_{i \in [n]} \phi(\cdot, x_i).$$

The KDE is used as an ingredient in a number of machine learning methodologies. For example, a common approach to classification is a plug-in rule that estimates the class-conditional densities with separate KDEs [4, 5]. In anomaly detection, a detector of the form $\widehat{f}(x) \gtrless \gamma$ is commonly employed to determine if a new realization comes from f [6, 7, 8, 9]. In clustering, the mean-shift algorithm forms a KDE and associates each data point to the mode of the KDE that is reached by hill-climbing [10].

Evaluating the KDE at a single test point requires $O(n)$ kernel evaluations, which is undesirable and perhaps prohibitive for large n. On the other hand, a sparse approximation with sparsity k requires only $O(k)$ kernel evaluations. This problem is magnified in algorithms such as mean-shift, where a (derivative of a) KDE is evaluated numerous times for each data point. In our experiments below, we demonstrate the computational savings of our approach in KDE-based algorithms for the embedding of probability distributions and mean-shift clustering.

2.2.2 Kernel Mean Embedding of Distributions

Let $\{x_1, \ldots, x_n\} \subset \mathbb{R}^d$ be a random sample from a distribution P. The idea behind the kernel mean embedding is to select a symmetric positive definite kernel ϕ, and embed P in the RKHS associated with ϕ via the mapping

$$\Psi(P) := \int \phi(\cdot, x) dP(x).$$

Since P is unknown, this mapping is estimated via the kernel mean

$$\widehat{\Psi}(P) := \frac{1}{n} \sum_{i \in [n]} \phi(\cdot, x_i).$$

The utility of the KME derives from the fact that for certain kernels, Ψ is injective. This permits the treatment of probability distributions as objects in a Hilbert space, which allows many existing machine learning methods to be applied in problems where probability distributions play the role of feature vectors [23, 24, 25, 26]. For example, suppose that random samples of size n are available from several probability distributions $P_1, \ldots, P_N$. A KME-based algorithm will require the computation of all pairs of inner products of kernel mean embeddings of these distributions. If $x_1, \ldots, x_n \sim P$ and $x'_1, \ldots, x'_n \sim P'$, then $\langle \widehat{\Psi}(P), \widehat{\Psi}(P') \rangle = \frac{1}{n^2} \sum_{i,j} \phi(x_i, x'_j)$ by the reproducing property. Therefore the calculation of all pairwise inner products of kernel mean embeddings requires $O(N^2 n^2)$ kernel evaluations. On the other hand, if we have sparse representations of the kernel means, these pairwise inner products can be calculated with only $O(N^2 k^2)$ kernel evaluations, a substantial computational savings. In our experiments below, we demonstrate the computational savings of our approach in KME-based algorithms for the embedding of probability distributions and class-proportion estimation.

2.2.3 Generalized Notion of Kernel

The problem of sparsely approximating a sample mean can be addressed more generally in an inner product space. This motivates the following definition of kernel, which is satisfied by both density estimation kernels and symmetric positive definite kernels.

Definition 7. *We say that $\phi : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is a* kernel *if there exists an inner product space $\mathcal{H}$ such that for all x in $\mathbb{R}^d$, $\phi(\cdot, x) \in \mathcal{H}$.*

In the case of kernel density estimation, all commonly used kernels satisfy $\phi(\cdot, x) \in L^2(\mathbb{R}^d)$ for all $x \in \mathbb{R}^d$. Here, $L^2(\mathbb{R}^d)$ is the space of equivalence classes of square integrable functions. When we write $\phi(\cdot, x) \in L^2(\mathbb{R}^d)$, we view $\phi(\cdot, x)$ as a representative of its equivalence class. In the case of the kernel mean embedding, we may simply take $\mathcal{H}$ to be the RKHS associated with ϕ.

Definition 8. *For $\{x_1, \ldots, x_n\}$ a subset of $\mathbb{R}^d$ and ϕ a kernel with associated inner product space $\mathcal{H}$, we call $K := (\langle \phi(\cdot, x_i), \phi(\cdot, x_j) \rangle_{\mathcal{H}})_{i,j \in [n]}$ the* kernel matrix.

Our proposed methodology applies to translation invariant kernels and beyond. For concreteness, however, we focus on radial kernels because of the connection to Euclidean geometry.

Definition 9. *We say $\phi : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is a* radial kernel *if ϕ is a kernel as in Def. 7 and there exists a strictly decreasing function $g : [0, \infty) \to \mathbb{R}$ such that, for all $x, x' \in \mathbb{R}^d$,*

$$\langle \phi(\cdot, x), \phi(\cdot, x') \rangle_{\mathcal{H}} = g(\|x - x'\|_2).$$

We now review some common examples of radial kernels. The Gaussian kernel with parameter $\sigma > 0$ has the form

$$\phi(x, x') = c_\sigma \exp\left(-\frac{\|x - x'\|_2^2}{2\sigma^2} \right),$$

the Laplacian kernel with parameter $\gamma > 0$ has the form

$$\phi(x, x') = c_\gamma \exp\left(-\frac{\|x - x'\|_2}{\gamma} \right),$$

and the Student-type kernel with parameters $\alpha, \beta > 0$ has the form

$$\phi(x, x') = c_{\alpha,\beta} \left(1 + \frac{\|x - x'\|_2^2}{\beta} \right)^{-\alpha}.$$

The parameters c_σ, c_γ and $c_{\alpha,\beta}$ can be set to 1 for the KME, or so as to normalize ϕ to be a density estimation kernel, depending on the application.

These examples illustrate that the space $\mathcal{H}$ such that $\phi(\cdot, x) \in \mathcal{H}$ is not unique. Indeed, each of these three kernels is a symmetric positive definite kernel, and therefore we may take $\mathcal{H}$ to be the RKHS associated with ϕ [16, 17]. On the other hand, we may also select $\mathcal{H} = L^2(\mathbb{R}^d)$.

Each of these three examples is also a radial kernel. If we take $\mathcal{H}$ to be the RKHS, then by the reproducing property we simply have $\langle \phi(\cdot, x), \phi(\cdot, x') \rangle = \phi(x, x')$, and in each case, $\phi(x, x') = g(\|x - x'\|)$ for some strictly decreasing g. These kernels are also radial if we take $\mathcal{H} = L^2(\mathbb{R}^d)$. For example, consider the Gaussian kernel, and let us write $\phi = \phi_\sigma$ to indicate the dependence on the bandwidth parameter. Then $\langle \phi_\sigma(\cdot, x), \phi_\sigma(\cdot, x') \rangle_{L^2} = \phi_{\sqrt{2}\sigma}(x, x')$. Similarly, for the Student kernel with $\alpha = (1 + d)/2$ (the Cauchy kernel), we have $\langle \phi_\beta(\cdot, x), \phi_\beta(\cdot, x') \rangle_{L^2} = \phi_{2\beta}(x, x')$. For other kernels, although there may not be a closed form expression for g, it can still be argued that such a g exists, which is all we will need.

2.2.4 Abstract Problem Formulation

In the interest of generality and clarity, we consider the problem of sparsely approximating a sample mean in a more abstract setting. Thus, let $(\mathcal{H}, \langle \cdot, \cdot \rangle)$ be an inner product space with induced norm $\|\cdot\|_{\mathcal{H}}$, and let $\{z_1, \ldots, z_n\} \subset \mathcal{H}$. For $\alpha \in \mathbb{R}^n$, define $\|\alpha\|_0 := |\{i \mid \alpha_i \neq 0\}|$.

Given an integer $k \leq n$, our objective is to approximate the sample mean $\bar{z} = \frac{1}{n}\sum_i z_i$ as a k-sparse linear combination of $z_1, \ldots, z_n$. In particular, we want to solve the problem

$$\begin{aligned} &\text{minimize } \|\bar{z} - z_\alpha\|_{\mathcal{H}} \\ &\text{subject to } \|\alpha\|_0 = k \end{aligned} \tag{2.3}$$

where $z_\alpha = \sum_{i\in[n]} \alpha_i z_i$.

Note that problem (2.3) is of the form of the standard sparse approximation problem [27], where $\{z_1, \ldots, z_n\}$ is the so-called *dictionary* out of which the sparse approximation is built. Later we argue that existing sparse approximation algorithms are not suitable from a scalability perspective. Instead, we develop an approach that leverages the fact that the vector being sparsely approximated is the sample mean of the dictionary elements. We are most interested in the case where $z_i = \phi(\cdot, x_i)$ and ϕ is a kernel, but the discussion in Section 2.4 is held in this more abstract sense.

2.3 Related Work and Contributions

Problem (2.3) is a specific case of the sparse approximation problem. Since in general it is NP-hard many efforts have been made to approximate its solution in a feasible amount of time. See [27] for an overview. A standard method of approximation is Matching Pursuit. Matching Pursuit is a greedy algorithm originally designed for finite-dimensional signals, i.e., $\mathcal{H} = \mathbb{R}^d$. Following the notation of Problem (2.3) let $\bar{z}$ be the target vector we wish to approximate. In Matching Pursuit the first step is to pick an "atom" in $\{z_1, \ldots, z_n\}$ which captures most of $\bar{z}$ as measured by the magnitude of the inner product. After this first step the subsequent atoms are iteratively chosen according to which one captures more of the portion of $\bar{z}$ that hasn't been accounted for [28]. Note that just the first step of this algorithm requires to compute, for each z_i, the quantity $\langle \bar{z}, z_i \rangle = \frac{1}{n}\sum_{j\in[n]} \langle z_i, z_j \rangle$. Since we have n z_i's, the first step already takes $\Omega(n^2)$ inner product (kernel) evaluations, which is undesirable. A variant of matching pursuit specifically designed to approximate probability distributions through kernel means is kernel herding [29]. While [29] chooses the nonzero values of α_i to equal $1/k$, [30] proposes to use a line search to obtain nonuniform α_i values. In herding the complete kernel matrix is also computed, taking quadratic computational time. Another general common approach to sparse approximation, Basis Pursuit, has similar time complexity.

Several algorithms which focus specifically on the sparse KDE problem have been developed. In [31] a clustering method is used to approximate the KDE at a point by rejecting points which fail to belong to close clusters. In [32] a relevant subset of the data is chosen to minimize the L^2 error but at an expensive $O(n^2)$ cost. In [33, 34] a regression based approach is taken to estimate the KDE through its cumulative density function. These algorithms rely on the assumption that the kernel mean in question is a KDE, so cannot be generalized to other kernel means.

When the kernel mean is thought of as a mixture model, the model can be collapsed into a simpler one by reducing the number of its components through a similarity based merging procedure [35, 36, 37]. Since these methods necessitate the computation of all pairwise similarities, they present quadratic computational complexity. EM algorithms for this task result in similar computational requirements [38, 39].

A line of work that tries to speed up general kernel sums comes historically from n-body problems in physics, and makes use of fast multipole methods [40, 41]. The general

idea behind these methods is to represent the kernel in question by a truncated series expansion, and then use a space partitioning scheme to group points, yielding an efficient way to approximate group-group or group-point interactions, effectively reducing the number of kernel evaluations. These methods are usually kernel-dependent. For the case of the Gaussian kernel, see [42, 43] for two different space partitioning methods. Note that space partitioning schemes may suffer considerably in high dimensional settings. Also, since the kernel function is truncated through its series expansion, the resulting approximation may not integrate to 1. Contrary to these methods, our approach can still yield a valid density (discussed below).

The efforts of rapidly approximating general kernel based quantities have led to the use of ϵ-samples, or coresets. To define ϵ-samples, first denote the data $A := \{x_1, \ldots, x_n\}$ and the kernel quantity of interest $Q(A, x)$, where x is some query point (for example, the KDE is $Q(A, x) = \frac{1}{n}\sum_{i\in[n]} \phi(x_i, x)$). An ϵ-sample is a set $A' \subset A$ such that, for every query point x, $Q(A, x)$ and $Q(A', x)$ differ by less than ϵ with respect to some norm. See [44, 45] for the KDE case with ℓ_∞ norm. For the KME using the RKHS norm, see [46]. Both cases allow for constructions of ϵ-samples in near linear time with respect to the data size and $1/\epsilon$. Notice that our approach has the advantage that it handles both the KDE and KME cases simultaneously, and that if desired it can yield a valid density as the approximation.

Although most of the literature seems to concentrate on the KDE, there have also been efforts to speed up computation time in problems involving the KME. As in the ϵ-sample approach above, many of these problems require the distance between KMEs in the RKHS, so they focus on speeding up this calculation. In [47], for example, a fast method is devised for the specific case of the maximum mean discrepancy statistic used for the two-sample test.

Computing the kernel mean at each of the original points $\{x_1, \ldots, x_n\}$ can be thought of as a matrix vector multiplication, where the matrix in question is the kernel matrix. Therefore, an algebraic approach to this problem consists of choosing a suitable subset of the matrix columns and then approximating the complete matrix only through these columns. Among the most common of these is the Nyström method. In the Nyström method the kernel matrix K is approximated by the matrix $QW_r^+Q^T$, where Q is composed of a subset of the columns of K, indexed by $\mathcal{I}$, W is the matrix with entries K_{ij} for $(i, j) \in \mathcal{I} \times \mathcal{I}$, and W_r^+ is the best r-rank approximation to its pseudoinverse (see [48] for details). The columns composing Q are typically chosen randomly under some sampling distribution. See [49] for some examples of sampling distributions. As explained in Section 2.4.1, our approach is connected to the Nyström method and can be viewed as a particular scheme for column selection tailored to kernel means. The Nyström approximation of the kernel matrix is not the only one used though, and other algebraic approaches exist. In [50] for example, an interpolative decomposition of the kernel matrix is proposed.

In [51] a "coherence" based sparsification criterion is used in the context of one-class classification. The main idea is that each set of possible atoms $\{z_i|\alpha_i \neq 0\}$ can be quantified by the largest absolute value of the inner product between two different atoms. The method proposed requires the computation of the complete kernel matrix, and is therefore not suitable for our setting, which involves large data. The motivation for their coherence criterion, however, lies in the minimization of a bound on the approximation error. As seen in Section 2.4.2, we propose a similar bound as a starting point for our algorithm.

Contributions

We list a summary of contributions in this chapter.

- We present a bound on the sparse approximation error based on a novel measure of incoherence.
- We recognize that for radial kernels, and more generally for any kernel such that $\|\phi(\cdot, x)\|$ is constant (which includes all translation invariant kernels), minimizing the bound is equivalent to solving an instance of the k-center problem. The solution to the k-center problem, in turn, can be approximated by a linear running time algorithm.
- Our method for approximating the KDE can be implemented so that the sparse kernel mean is a valid density function, which is important for some applications. Note that some alternative methods cannot be adapted to do so.
- Our method provides amortization of computational complexity since the calculation of the set $\mathcal{I}$ (introduced below) is only done once. Many subsequent calculations (e.g., kernel bandwidth search) can then be performed at a relatively small or negligible cost.
- Our method is versatile in that it addresses different types of kernel means under a common framework. In particular, it can be used to approximate both KMEs and KDEs at the same time.
- Our method provides a scheme to automatically select the sparsity level.
- We demonstrate the improved performance of the proposed method in three different applications, Euclidean embedding of probabilities (using both the KDE and the KME), class proportion estimation (using the KME), and clustering with the mean-shift algorithm (using the KDE), as well as on several additional benchmark datasets.

2.4 Subset Selection and Incoherence-Based Bound

Let us now reformulate problem (2.3). Our approach will be to separate the problem into two parts: that of finding the set of indices i such that α_i is not zero, and that of finding the value of the nonzero α_i's. Letting $\mathcal{I} \subset [n]$ denote an index set, we can pose problem (2.3) as

$$\min_{\substack{\mathcal{I}\subseteq[n] \\ |\mathcal{I}|=k}} \min_{(\alpha_i)_{i\in\mathcal{I}}} \|\bar{z} - \sum_{i\in\mathcal{I}} \alpha_i z_i\|^2 . \tag{2.4}$$

Note that the inner optimization problem is unconstrained and quadratic, and its solution, which for fixed $\mathcal{I}$ and k we denote by $\alpha_{\mathcal{I}} \in \mathbb{R}^k$, is

$$\alpha_{\mathcal{I}} = K_{\mathcal{I}}^{-1} \kappa_{\mathcal{I}},$$

where $K_{\mathcal{I}} = (\langle z_i, z_j \rangle)_{i,j\in\mathcal{I}}$ and $\kappa_{\mathcal{I}}$ is the k-dimensional vector with entries $\frac{1}{n}\sum_{j\in[n]} \langle z_j, z_l \rangle$, $l \in \mathcal{I}$.

Let $\alpha_{\mathcal{I}} = (\alpha_{\mathcal{I},i})_{i\in\mathcal{I}}$ and $z_{\mathcal{I}} = \sum_{i\in\mathcal{I}} \alpha_{\mathcal{I},i} z_i$. Then we can rewrite problem (2.3) as

$$\min_{\substack{\mathcal{I}\subseteq[n]\\|\mathcal{I}|=k}} \|\bar{z} - z_{\mathcal{I}}\| . \tag{2.5}$$

2.4.1 Connection to the Nyström Method

Before continuing to the approximate solution of problem (2.5), we briefly highlight its relationship to the Nyström method. Given a set $\mathcal{I} \subset [n]$, let K be the kernel matrix of $\{z_i \,|\, i \in [n]\}$, $K := (\langle z_i, z_j\rangle)_{i,j\in[n]}$, and $K_{\mathcal{I}}$ the kernel matrix of $\{z_i \,|\, i \in \mathcal{I}\}$, $K_{\mathcal{I}} := (\langle z_i, z_j\rangle)_{i,j\in\mathcal{I}}$. Also, let $Q_{\mathcal{I}}$ be the binary matrix such that $KQ_{\mathcal{I}}$ is composed of the columns of K corresponding to $\mathcal{I}$. Then we can rewrite $\alpha_{\mathcal{I}}$ and $K_{\mathcal{I}}$ as $\alpha_{\mathcal{I}} = \left(Q_{\mathcal{I}}^T K Q_{\mathcal{I}}\right)^{-1} Q_{\mathcal{I}}^T K \mathbf{1}_n$ and $K_{\mathcal{I}} = Q_{\mathcal{I}}^T K Q_{\mathcal{I}}$, where $\mathbf{1}_n$ denotes the vector in $\mathbb{R}^n$ with entries $1/n$. By doing so, we can express the objective of (2.5) as

$$\begin{aligned}\|\bar{z} - z_{\mathcal{I}}\|^2 &= \mathbf{1}_n^T \left(K - KQ_{\mathcal{I}}K_{\mathcal{I}}^{-1}Q_{\mathcal{I}}^T K^T\right)\mathbf{1}_n \\ &= \mathbf{1}_n^T \left(K - \tilde{K}_{\mathcal{I}}\right)\mathbf{1}_n.\end{aligned}$$

where $\tilde{K}_{\mathcal{I}} := KQ_{\mathcal{I}}K_{\mathcal{I}}^{-1}Q_{\mathcal{I}}^T K^T$. We recognize $\tilde{K}_{\mathcal{I}}$ as the Nyström matrix from the Nyström method [49], which is the only term dependent on $\mathcal{I}$ in the objective. Therefore, our work can be interpreted from the Nyström perspective: choose suitable columns of K and approximate K through the Nyström matrix. The main difference is that the resulting approximation is assessed using the induced norm of the inner product space where the z_i's reside, instead of the commonly used spectral and Frobenius norms.

2.4.2 An Incoherence-based Sparse Approximation Bound

We now present our proposed algorithm to approximate the solution of problem (2.5). Our strategy is to find an upper bound on $\|\bar{z} - z_{\mathcal{I}}\|$ which is dependent on $\mathcal{I}$ and then find the $\mathcal{I}$ that minimizes the bound. First, we present a lemma which will aid us in finding the bound.

Lemma 1. *Let $(\mathcal{H}, \langle\cdot,\cdot\rangle)$ be an inner product space. Let S be a finite dimensional subspace of $\mathcal{H}$ and P_S the projection onto S. For any $z_0 \in \mathcal{H}$*

$$\|P_S z_0\| = \max_{z\in S, \|z\|=1} \langle z_0, z\rangle .$$

Proof. First note that since S is finite dimensional, by the Projection Theorem $z_0 - P_S z_0$ is orthogonal to S. Now, for any $z \in S$ with $\|z\| = 1$, we have

$$\begin{aligned}\langle z_0, z\rangle &= \langle P_S z_0 + (z_0 - P_S z_0), z\rangle \\ &= \langle P_S z_0, z\rangle + \langle z_0 - P_S z_0, z\rangle \\ &= \langle P_S z_0, z\rangle \\ &\leq \|P_S z_0\| \, \|z\| = \|P_S z_0\| ,\end{aligned}$$

where we have used the Cauchy-Schwartz inequality. To confirm the existence of a vector z which makes it an equality and therefore reaches the maximum, just let $z = P_S z_0 / \|P_S z_0\|$. □

We can now present the theorem which will be the basis for our minimization approach. First, define

$$\nu_{\mathcal{I}} := \min_{j \notin \mathcal{I}} \max_{i \in \mathcal{I}} \langle z_i, z_j \rangle ,$$

which we can think of as a measure of the "incoherence" of $\{z_i \mid i \in \mathcal{I}\}$. It is now possible to establish a bound:

Theorem 1. *Assume that for some $C > 0$ $\langle z_i, z_i \rangle = C \ \forall i \in [n]$. Then for every $\mathcal{I} \subseteq [n]$,*

$$\|\bar{z} - z_{\mathcal{I}}\| \leq \left(1 - \frac{|\mathcal{I}|}{n}\right) \sqrt{\frac{1}{C}(C^2 - \nu_{\mathcal{I}}^2)}.$$

Proof. The beginning of this proof is similar to the one in [51]. Let $S_{\mathcal{I}} := \operatorname{span}(\{z_i \mid i \in \mathcal{I}\})$ and denote $P_{S_{\mathcal{I}}}$ the projection operator onto $S_{\mathcal{I}}$ and I the identity operator. We have

$$\|\bar{z} - z_{\mathcal{I}}\| = \|\bar{z} - P_{S_{\mathcal{I}}}\bar{z}\| = \frac{1}{n}\|\sum_{i \in [n]}(I - P_{S_{\mathcal{I}}})z_i\|$$

$$\leq \frac{1}{n}\sum_{i \in [n]}\|(I - P_{S_{\mathcal{I}}})z_i\| = \frac{1}{n}\sum_{i \notin \mathcal{I}}\|(I - P_{S_{\mathcal{I}}})z_i\|$$

where we have used the triangle inequality, and the last equality is due to the fact that $z_i = P_{S_{\mathcal{I}}} z_i$ when $z_i \in S_{\mathcal{I}}$.

Now, since $(z_i - P_{S_{\mathcal{I}}} z_i) \perp P_{S_{\mathcal{I}}} z_i$, we can use Pythagoras' Theorem in $\mathcal{H}$ to get $\|z_i - P_{S_{\mathcal{I}}} z_i\|^2 = \|z_i\|^2 - \|P_{S_{\mathcal{I}}} z_i\|^2$.

By Lemma 1, $\|P_{S_{\mathcal{I}}} z_i\| = \max_{z \in S_{\mathcal{I}},\, \|z\|=1} \langle z_i, z \rangle$. Therefore, for $i \notin \mathcal{I}$,

$$\begin{aligned}
\|P_{S_{\mathcal{I}}} z_i\| &= \frac{1}{\sqrt{C}} \max_{z \in S_{\mathcal{I}}, \|z\| = \sqrt{C}} \langle z_i, z \rangle \\
&\geq \frac{1}{\sqrt{C}} \max_{\ell \in \mathcal{I}} \langle z_i, z_\ell \rangle \\
&\geq \frac{1}{\sqrt{C}} \min_{j \notin \mathcal{I}} \max_{\ell \in \mathcal{I}} \langle z_j, z_\ell \rangle = \frac{1}{\sqrt{C}}\, \nu_{\mathcal{I}}.
\end{aligned}$$

Thus, for $i \notin \mathcal{I}$,

$$\|z_i\|^2 - \|P_{S_{\mathcal{I}}} z_i\|^2 \leq C - \frac{\nu_{\mathcal{I}}^2}{C}$$

and finally

$$\|\bar{z} - z_{\mathcal{I}}\| \leq \frac{1}{n}\sum_{i \notin \mathcal{I}} \sqrt{C - \frac{\nu_{\mathcal{I}}^2}{C}} = \left(1 - \frac{|\mathcal{I}|}{n}\right) \sqrt{\frac{1}{C}(C^2 - \nu_{\mathcal{I}}^2)}\,.$$

□

2.4.3 Application to Kernel Means

We now apply the previous result in the context of approximating a kernel mean based on a radial kernel. Recall that, in the kernel mean setting, $z_i = \phi(\cdot, x_i)$ and $\langle \phi(\cdot, x_i), \phi(\cdot, x_j) \rangle = g(\|x_i - x_j\|_2)$, where ϕ is a radial kernel, $\{x_1, \ldots, x_n\} \subset \mathbb{R}^d$, and g is strictly decreasing as in Definition 9. Also note that for any radial kernel the assumption in Theorem 1 is satisfied, since $\langle \phi(\cdot, x_i), \phi(\cdot, x_i) \rangle = g(0) = C > 0$. Here $\bar{f}$ and $f_{\mathcal{I}}$ are defined in an analogous way to $\bar{z}$ and $z_{\mathcal{I}}$, with $\bar{f}$ being a kernel mean and $f_{\mathcal{I}}$ its sparse approximation. The following corollary follows directly from Theorem 1.

Corollary 1. *Let ϕ be a radial kernel, with $\langle \phi(\cdot, x), \phi(\cdot, x) \rangle_{\mathcal{H}} = C \ \forall x \in \mathbb{R}^d$. Then for every $\mathcal{I} \subseteq [n]$,*

$$\left\| \bar{f} - f_{\mathcal{I}} \right\| \leq \left(1 - \frac{|\mathcal{I}|}{n} \right) \sqrt{\frac{1}{C} \left(C^2 - \nu_{\mathcal{I}}^2 \right)}.$$

□

For the case of symmetric positive definite kernels, with $\mathcal{H}$ the corresponding RKHS, bounding the $\mathcal{H}$ norm implies bounding the L^∞ norm, as stated in the following corollary.

Corollary 2. *Let ϕ be a symmetric positive definite kernel with associated RKHS $\mathcal{H}$, with $\langle \phi(\cdot, x), \phi(\cdot, x) \rangle_{\mathcal{H}} = C \ \forall x \in \mathbb{R}^d$. Then for every $\mathcal{I} \subseteq [n]$,*

$$\left\| \bar{f} - f_{\mathcal{I}} \right\|_\infty \leq \left(1 - \frac{|\mathcal{I}|}{n} \right) \sqrt{C^2 - \nu_{\mathcal{I}}^2}.$$

Proof.

$$\begin{aligned} \left\| \bar{f} - f_{\mathcal{I}} \right\|_\infty &= \max_x \left| \left(\bar{f} - f_{\mathcal{I}} \right)(x) \right| \\ &= \max_x \left| \left\langle \bar{f} - f_{\mathcal{I}}, \phi(\cdot, x) \right\rangle \right| \\ &\leq \max_x \left\| \bar{f} - f_{\mathcal{I}} \right\|_{\mathcal{H}} \left\| \phi(\cdot, x) \right\|_{\mathcal{H}} \\ &= \sqrt{C} \left\| \bar{f} - f_{\mathcal{I}} \right\|_{\mathcal{H}}, \end{aligned}$$

where the second line is due to the reproducing property and the third to the Cauchy-Schwarz inequality. □

For some applications, such as density estimation, one may desire the approximation to belong to $\Delta := \left\{ \sum_{i \in \mathcal{I}} \alpha_i z_i \mid \sum \alpha_i = 1, \alpha_i \geq 0 \right\}$. A similar bound can be derived, but is not as tight as the previous ones. It also suggests, however, the maximization of the term $\nu_{\mathcal{I}}$. In particular, we have that, under the assumptions of Corollary 1, and letting f_Δ be the projection of $\bar{f}$ onto Δ:

$$\left\| \bar{f} - f_\Delta \right\| \leq \sqrt{2 \left(C - \left(1 - \frac{|\mathcal{I}|}{n} \right) \nu_{\mathcal{I}} \right)}.$$

Details are shown in Chapter 3.

2.5 Bound Minimization Via k-center Algorithm

The bound in the previous corollaries can be minimized by maximizing the term $\nu_{\mathcal{I}}$. We now present a procedure to accomplish this for the case of radial kernels. Let ϕ be a radial kernel and define the set $\mathcal{I}^*$ as

$$\mathcal{I}^* := \arg \min_{\substack{\mathcal{I} \subseteq [n] \\ |\mathcal{I}| = k}} \max_{j \notin \mathcal{I}} \min_{i \in \mathcal{I}} \left\| x_i - x_j \right\|.$$

Then, since $\langle \phi(\cdot, x_i), \phi(\cdot, x_j) \rangle = g(\|x_i - x_j\|_2)$ for g strictly decreasing, $\mathcal{I}^*$ also maximizes $\nu_{\mathcal{I}} = \min_{j \notin \mathcal{I}} \max_{i \in \mathcal{I}} g(\|x_i - x_j\|)$. Therefore, $\mathcal{I}^*$ is the set that minimizes the bound in Theorem

input $x_1, \ldots, x_n, k$
$X \longleftarrow \varnothing$
$Y \longleftarrow \{x_1, \ldots, x_n\}$
Choose randomly a first index $u \in [n]$
$X \longleftarrow X \cup \{x_u\}$
$Y \longleftarrow Y \backslash \{x_u\}$
while $|X| < k$ **do**
  Choose the element $y \in Y$ for which $d(y, X)$ is maximized
  $X \longleftarrow X \cup \{y\}$
  $Y \longleftarrow Y \backslash \{y\}$
end while
output $\mathcal{I}_k = \{i \in [n] \,|\, x_i \in X\}$

Figure 2.1: A linear time 2-approximation algorithm for the k-center problem.

1. We have translated a problem involving inner products of functions to a problem involving distances between points in $\mathbb{R}^d$.

The problem of finding $\mathcal{I}^*$ is known as the k-center problem. To pose the k-center problem more precisely, we make a few definitions. For a fixed $\mathcal{I}$, let $X_\mathcal{I} = \{x_i \mid i \in \mathcal{I}\}$ and $Y_\mathcal{I} = \{x_j \mid j \notin \mathcal{I}\}$, and for all $x_j \in Y_\mathcal{I}$ define its distance to $X_\mathcal{I}$ as $d(x_j, X_\mathcal{I}) = \min_{x_i \in X_\mathcal{I}} \|x_i - x_j\|$. Furthermore, let $W(X_\mathcal{I}) = \max_{x_j \in Y_\mathcal{I}} d(x_j, X_\mathcal{I})$. Then, the k-center problem is that of finding the set $\mathcal{I}$ of size k for which $W(X_\mathcal{I})$ is minimized.

The k-center problem is known to be NP-complete [52]. However, there exists a greedy 2-approximation algorithm [53] which produces a set $\mathcal{I}_k$ such that $W(X_{\mathcal{I}_k}) \leq 2W(X_{\mathcal{I}^*})$. This algorithm is optimal in the sense that under the assumption that P$\neq$NP there is no ρ-approximation algorithm with $\rho < 2$ [54]. The algorithm is described in Fig. 2.1, and as can be seen, it has a linear time complexity in the size of the data n. In particular, the algorithm runs in $O(nkd)$ time.

To relate the output of the algorithm back to the bound of the theorem, note that $\nu_\mathcal{I} = g(W(X_\mathcal{I}))$. Since the k-center algorithm guarantees that $W(X_{\mathcal{I}_k}) \leq 2W(X_{\mathcal{I}^*})$, in the most general case we can say that $\sqrt{C^2 - \nu_{\mathcal{I}_k}^2} = \sqrt{C^2 - g(W(X_{\mathcal{I}_k}))^2} \leq \sqrt{C^2 - g(2W(X_{\mathcal{I}^*}))^2}$. Knowing more about the form of g yields more information. For example, for the Gaussian kernel we have

$$\begin{aligned}
&\sqrt{\frac{1}{C}\left(C^2 - \nu_{\mathcal{I}_k}^2\right)} \leq \sqrt{\frac{1}{C}\left(C^2 - \frac{1}{C^6}\nu_{\mathcal{I}^*}^8\right)} \\
&\leq \sqrt{\frac{1}{CC^6}\left(C^4 - \nu_{\mathcal{I}^*}^4\right)\left(C^4 + \nu_{\mathcal{I}^*}^4\right)} \\
&\leq \sqrt{\frac{2C^4}{CC^6}\left(C^2 - \nu_{\mathcal{I}^*}^2\right)\left(C^2 + \nu_{\mathcal{I}^*}^2\right)} \leq 2\sqrt{\frac{1}{C}\left(C^2 - \nu_{\mathcal{I}^*}^2\right)}.
\end{aligned}$$

2.5.1 Generalization to nonradial kernels

The preceding argument extends to certain nonradial kernels. For example, consider kernels satisfying $\langle \phi(\cdot, x), \phi(\cdot, x') \rangle = g(M(x, x'))$, where M is a non-Euclidean metric on $\mathbb{R}^d$. Examples include the Gaussian kernel with anisotropic covariance where M is a Mahalanobis distance, or a type of Laplacian kernel where M is the ℓ_1 distance. The

k-center algorithm described previously applies in these settings as well, where distance in the algorithm is computed using M.

Another example of a nonradial kernel is a *discriminative kernel* for classification problems. Let x denote a feature vector and $y \in \{-1, 1\}$ its label. If ϕ is a kernel on $\mathbb{R}^d$, then $\psi((x, y), (x', y')) = yy'\phi(x, x')$ is a kernel on $\mathbb{R}^d \times \{-1, 1\}$. A kernel mean based on this kernel is a well-known classification algorithm [**?**]. For the discriminative kernel, the problem of maximizing the incoherence reduces to a variation of the k-center problem: find k_1 points in one class and k_{-1} points in the other class such that $k_1 + k_{-1} = k$ and the maximum distance of a point to its nearest representative *in the same class* is minimized. The k-center algorithm described previously can be easily adapted to solve this problem.

More generally, let ϕ be any kernel such that $\|\phi(\cdot, x)\|^2 = C$ is independent of x. This includes any translation invariant kernel. Then

$$\langle \phi(\cdot, x), \phi(\cdot, x') \rangle = \frac{1}{2}(2C - \|\phi(\cdot, x) - \phi(\cdot, x')\|^2).$$

Hence, the problem of solving

$$\max_{\substack{\mathcal{I} \subseteq [n] \\ |\mathcal{I}|=k}} \min_{j \notin \mathcal{I}} \max_{i \in \mathcal{I}} \langle \phi(\cdot, x_i), \phi(\cdot, x_j) \rangle$$

reduces to solving

$$\min_{\substack{\mathcal{I} \subseteq [n] \\ |\mathcal{I}|=k}} \max_{j \notin \mathcal{I}} \min_{i \in \mathcal{I}} \|\phi(\cdot, x_i) - \phi(\cdot, x_j)\| .$$

In other words, it suffices to solve the k-center problem in the kernel feature space $\mathcal{H}$, using the induced $\mathcal{H}$ norm, which can be calculated efficiently using the kernel. Once again, the same k-center algorithm can be applied, where distances are now computed as $\|\phi(\cdot, x_i) - \phi(\cdot, x_j)\| = \sqrt{2C - 2\phi(x_i, x_j)}$. We also note that Corollaries 1 and 2 also hold in this generalized setting. It is worth noting that for the case of radial kernels the k-center algorithm can be used in both the feature space as in Euclidean space, which will potentially yield two different results. We have not empirically validated the nonradial kernel case.

2.5.2 Computation of $\alpha_{\mathcal{I}}$ and Auto-selection of k

The k-center algorithm allows us to find the set $\mathcal{I}$ on which our approximation will be based. After finding $\mathcal{I}$ we can determine the optimal coefficients $\alpha_{\mathcal{I}}$. Since the main computational burden is in the selection of $\mathcal{I}$, we now have the freedom to explore different values of $\alpha_{\mathcal{I}}$ in a relatively small amount of time. For example, we can compute $\alpha_{\mathcal{I}}$ for each of several possible kernel bandwidths σ.

The optimal way to compute $\alpha_{\mathcal{I}}$ depends on the application. If the user has a good idea of what the value of k is, then a fast way to compute $\alpha_{\mathcal{I}}$ for that specific value is to apply their preferred method to solve the equation $K_{\mathcal{I}}\alpha_{\mathcal{I}} = \kappa_{\mathcal{I}}$. For example, since for symmetric positive definite kernels the kernel matrix is positive semi-definite, the preconditioned conjugate gradient method can be used to quickly obtain $\alpha_{\mathcal{I}}$ to high accuracy. This approach has the advantages of being simple and fast.

A further advantage of our method is evident when the user can accept a maximum tolerance value of k, say k_{max}, but would prefer to stop at a value $k_0 \leq k_{max}$ that performs about as well as k_{max}. To do this, at iteration $m \geq 1$ in the k-center algorithm we compute

$\alpha_{\mathcal{I}_m}$ right after computing $\mathcal{I}_m$, which provides a record of all the $\alpha_{\mathcal{I}_j}$ for $1 \leq j \leq k_0$. To find k_0, we use the information from the computed coefficients to form an error indicator and stop when some error threshold is crossed. Before showing what these error indicators are, we first provide an update rule to efficiently compute the α coefficients at each iteration step.

Let $\mathcal{I}_m$ be the set of the first m elements chosen by the k-center algorithm, and let $\alpha_{\mathcal{I}_m}$, $K_{\mathcal{I}_m}$ and $\kappa_{\mathcal{I}_m}$ be obtained by using $\mathcal{I}_m$. If we increase the number of components to $m+1$, then as shown in [51] we have

$$K_{\mathcal{I}_{m+1}} = \begin{bmatrix} K_{\mathcal{I}_m} & b \\ b^T & \phi(x_{j_{m+1}}, x_{j_{m+1}}) \end{bmatrix}$$

where x_{j_ℓ} is the ℓ^{th} element selected by the k-center algorithm, and $b = (\phi(x_{j_{m+1}}, x_i))_{i \in \mathcal{I}_m}$. The resulting update rule for the inverse is

$$K^{-1}_{\mathcal{I}_{m+1}} = \begin{bmatrix} K^{-1}_{\mathcal{I}_m} & 0 \\ 0 & 0 \end{bmatrix} + q_0(qq^T)$$

where $q_0 = 1/(\phi(x_{j_{m+1}}, x_{j_{m+1}}) - b^T K^{-1}_{\mathcal{I}_m} b)$ and $q = \begin{bmatrix} -b^T K^{-1}_{\mathcal{I}_m} & 1 \end{bmatrix}^T$. From here the user can now compute $\alpha_{\mathcal{I}_{m+1}}$ by multiplying $K^{-1}_{\mathcal{I}_{m+1}}$ with

$$\kappa_{\mathcal{I}_{m+1}} = \begin{bmatrix} \kappa_{\mathcal{I}_m} \\ \frac{1}{n}\sum_{i=1}^n \phi(x_{j_{m+1}}, x_i) \end{bmatrix}.$$

Assuming we stop at k_{max}, the time complexity for computing all the $\alpha_{\mathcal{I}_m}$'s is $O(k^3_{max})$ and the necessary memory $O(k^2_{max})$.

Note that our incremental approach to construct $\alpha_{\mathcal{I}}$ does assume that the kernel matrix $K_{\mathcal{I}}$ is full rank, since it does compute $K^{-1}_{\mathcal{I}}$ explicitly, and not the pseudeoinverse (as opposed to, say, the Nyström method). For Gaussian and similar kernels, $K_{\mathcal{I}}$ is positive definite assuming the x_i's are distinct. Rank deficiency results from selecting centers that are very close to each other, however, the k-center algorithm does the opposite and selects elements far apart from each other, which supports the assumption of a full rank matrix $K_{\mathcal{I}}$.

To automatically stop at some $k_0 \leq k_{max}$ we need a stopping criterion based on some form of error. We propose the following: using the notation of problem (2.5) we have that

$$\begin{aligned} \|\bar{z} - z_{\mathcal{I}}\|^2 &= \left\langle \frac{1}{n}\sum_{\ell\in[n]} z_\ell, \frac{1}{n}\sum_{\ell'\in[n]} z_{\ell'} \right\rangle \\ &- 2\left\langle \frac{1}{n}\sum_{\ell\in[n]} z_\ell, \sum_{i\in\mathcal{I}} \alpha_{\mathcal{I},i} z_i \right\rangle + \left\langle \sum_{i\in\mathcal{I}} \alpha_{\mathcal{I},i} z_i, \sum_{j\in\mathcal{I}} \alpha_{\mathcal{I},j} z_j \right\rangle \\ &= \|\bar{z}\|^2 - 2\cdot \sum_{i\in\mathcal{I}} \alpha_{\mathcal{I},i} \cdot \frac{1}{n}\sum_{\ell\in[n]} \langle z_\ell, z_i\rangle + \alpha^T_{\mathcal{I}} K_{\mathcal{I}} \alpha_{\mathcal{I}} \\ &= \|\bar{z}\|^2 - \alpha^T_{\mathcal{I}} \kappa_{\mathcal{I}}. \end{aligned}$$

Since $\|\bar{z}\|^2$ is a constant independent of $\mathcal{I}$, we can avoid its $O(n^2)$ computation and only use the quantities $E_{|\mathcal{I}|} := -\alpha^T_{\mathcal{I}} \kappa_{\mathcal{I}}$ as error indicators. Note that E_t is nonincreasing with

respect to t. Based on this we choose k_0 to be the first value at which some relative error is small. In this paper we use the test

$$\frac{|E_{k_0-1} - E_{k_0}|}{|E_1 - E_{k_0}|} \leq \epsilon$$

for some small ϵ. The overall complexity is then reduced to $O(nk_0d + k_0^3 d)$.

A further consideration for computing $\alpha_{\mathcal{I}}$ should be made if the result is desired to be a probability density function. In this case a k-dimensional $\alpha_{\mathcal{I}}$ can be projected into the simplex $\Delta^{k-1} := \left\{\nu \in \mathbb{R}^k | \sum_{i=1}^k \nu_i = 1, \nu_i \geq 0 \ \forall \ 1 \leq i \leq k\right\}$ after being obtained by any of the discussed methods (see [55]). This extra step takes negligible time with respect to the rest of the computations. Alternatively, a quadratic program which takes into account the constraints of non-negativity and $\sum_{i=1}^k \alpha_{\mathcal{I},i} = 1$ can be solved.

A Matlab implementation of the complete Sparse Kernel Mean procedure can be found at [22].

2.6 Experiments: Speeding Up Existing Kernel Mean Methods

We have implemented our approach in three specific machine learning tasks that require the computation and evaluation of a mean of kernels. In the first of these, we apply our algorithm to the task of dimensionality reduction. In the second, we use it in the setting of class proportion estimation. In the third, we explore its performance when used as part of the mean shift algorithm. Finally, for 11 benchmark datasets we compare the performance of our approach to four other similar methods.

In the following we refer to our algorithm or to the resulting kernel mean as SKM (for Sparse Kernel Mean). We now provide a detailed description of each task and relevant results. The implementation has been done in Matlab.

2.6.1 Euclidean Embedding of Distributions

In this experiment we embed probability distributions in a lower dimensional space for the purpose of visualization. Given a collection of N distributions $\{P_1, \ldots, P_N\}$, the procedure consists of creating a dissimilarity matrix for some notion of dissimilarity among these distributions and then performing a dimensionality reduction method. We consider two cases. In the first case the dissimilarity matrix will be the distance between the kernel mean embeddings of the distributions in the RKHS (KME case), while in the second case it will be the (symmetrized) KL divergence between KDEs (KDE case). For dimensionality reduction we will use ISOMAP [56]. In the setup we have access to each of N distributions $\{P_1, \ldots, P_N\}$ through samples drawn from those distributions. The sample drawn from the ℓ^{th} distribution is denoted $\left\{x_i^{(\ell)}\right\}_{i=1}^{n_\ell}$.

Notice that in the KDE case, in order to compute the KL divergence it is necessary to obtain a valid density function. By choosing the coefficients as described in Section 2.5.2, the resulting sparse approximation is a density function.

Let us start with the KME case, in which the dissimilarity is the difference between the distributions' KMEs. The first task is to estimate the KME using some symmetric

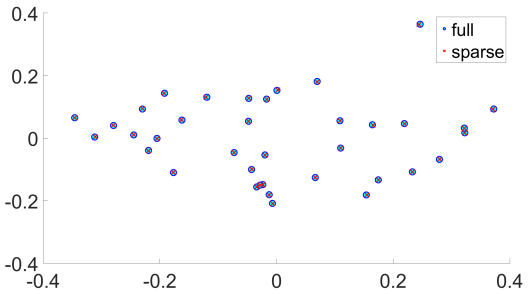

Figure 2.2: 2-dimensional representation of flow cytometry data - KME case. Each point represents a patient's distribution. The embeddings were obtained by applying ISOMAP to distances in the RKHS.

positive definite kernel ϕ. For the ℓ^{th} distribution, the empirical estimate of its KME is

$$\widehat{\Psi}(P_\ell) = \frac{1}{n_\ell}\sum_{i=1}^{n_\ell}\phi(\cdot, x_i^{(\ell)}),$$

with a sparse approximation

$$\widehat{\Psi}_0(P_\ell) = \sum_{i\in\mathcal{I}^{(\ell)}}\alpha_i^{(\ell)}\phi(\cdot, x_i^{(\ell)}),$$

for some set $\mathcal{I}^{(\ell)}$ and $\left\{\alpha_i^{(\ell)}|\alpha_i^{(\ell)}\in\mathbb{R}\right\}$, where the α coefficients have been computed according to the update method described in Section 2.5.2.

Given all the KMEs, we can now construct a distance matrix. Let $\mathcal{H}$ be the RKHS of ϕ. We can use the distance induced by the RKHS to create the matrix D, with entries

$$\begin{aligned} D_{\ell,\ell'} &:= \left\|\widehat{\Psi}(P_\ell) - \widehat{\Psi}(P_{\ell'})\right\|_{\mathcal{H}} \\ &= \left[\frac{1}{n_\ell^2}\sum_{i,j}\phi(x_i^{(\ell)}, x_j^{(\ell)}) - 2\frac{1}{n_\ell n_{\ell'}}\sum_{i,j}\phi(x_i^{(\ell)}, x_j^{(\ell')})\right. \\ &\quad \left. + \frac{1}{n_{\ell'}^2}\sum_{i,j}\phi(x_i^{(\ell')}, x_j^{(\ell')})\right]^{1/2}. \end{aligned}$$

We similarly define D_0 based on the sparse KMEs. With such matrices ISOMAP can now be performed to visualize the distributions in, say, $\mathbb{R}^2$.

Note that if the samples from P_ℓ and $P_{\ell'}$ have n_ℓ and $n_{\ell'}$ points, then $D_{\ell,\ell'}$ takes $\Theta(n_\ell^2 + n_\ell n_{\ell'} + n_{\ell'}^2)$ time to compute. Since we need all the pairwise distances, we need $\Theta(N^2)$ such computations. A sparse approximation of the KMEs of P_ℓ and $P_{\ell'}$ of sizes k_ℓ and $k_{\ell'}$ would instead yield a computation of $\Theta(k_\ell^2 + k_\ell k_{\ell'} + k_{\ell'}^2)$ for each entry. Assuming all samples have the same size n, and the sparse approximation size is k, then the computation of the distance matrix is reduced from $\Theta(N^2n^2)$ to $\Theta(N^2k^2)$.

Inspired by the work of [57], we have performed these experiments on flow cytometry data from $N = 37$ cancer patients, with sample sizes ranging from 8181 to 108343. We have used the Gaussian kernel, chosen $\mathcal{H}$ to be its RKHS, and computed the bandwidth based on the 'iqr' scale option in R's KernSmooth package. That is, we have computed the interquartile range of the data, averaged over each dimension, and divided by 1.35.

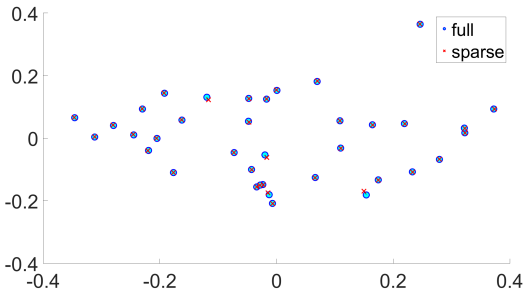

Figure 2.3: 2-dimensional representation of flow cytometry data - KME case. D_0 was found through kernel herding.

Table 2.1: Time comparison for the Euclidean embedding of the flow cytometry dataset - KME case.

	Approximation	D computation	Total
Full	0	3.04hrs	3.04hrs
SKM	1.29mins	1.6s	1.31mins
HERD	30.38mins	1.8s	30.42mins

After the embedding has been done, we have performed Procrustes analysis on the points so as to account for possible translation, scaling, and rotation.

To determine the maximum size k_ℓ of each sparse representation, we recall that the SKM procedure takes $O(n_\ell k_\ell + k_\ell^3)$ kernel evaluations, so in order to respect the $n_\ell k_\ell$ factor, we have chosen a small multiple of $\sqrt{n_\ell}$ for k_ℓ. In this case we picked k_ℓ to be the largest integer smaller than $3\sqrt{n_\ell}$ for each ℓ. We have implemented the auto-selection scheme described in Section 2.5.2. The results for the case of $\epsilon = 10^{-12}$ are shown in Fig. 2.2 and Table 2.1. As a comparison, we also compute an alternative D_0 based on kernel herding (see Section 2.3) and plot the result in Fig. 2.3. Note that although for this fixed ϵ HERD seems to fit better the projection resulting from using the full kernel mean, as seen in Table 2.1 it takes 30 minutes to do so. For the SKM an almost identical projection can be generated in just over a minute.

Although k_ℓ is the largest allowed sparsity, each algorithm stops at some $k_{0\ell} \leq k_\ell$. The values $\frac{k_{0\ell}}{k_\ell}$ averaged over all ℓ's are shown in Fig. 2.4. To determine how well D_0 approximates D, we have plotted the relative error $\frac{\|D-D_0\|_F}{\|D\|_F}$ for different values of ϵ. The result is shown in Fig. 2.4.

The KDE case is similar. The dissimilarity matrix is composed of the symmetrized KL divergence between the KDEs of the distributions, defined as $d_{KL}(p,q) := D_{KL}(p\|q) + D_{KL}(q\|p)$, where D_{KL} indicates the KL divergence. For the ℓ^{th} distribution, its KDE is

$$\widehat{f}_\ell = \frac{1}{n_\ell}\sum_{i=1}^{n_\ell}\phi(\cdot, x_i^{(\ell)}),$$

with a sparse approximation

$$\widehat{f}_{0\ell} = \sum_{i\in\mathcal{I}^{(\ell)}}\alpha_i^{(\ell)}\phi(\cdot, x_i^{(\ell)}).$$

for some set $\mathcal{I}^{(\ell)}$ and $\left\{\alpha_i^{(\ell)}|\alpha_i^{(\ell)} \geq 0 \ , \ \sum_i \alpha_i^{(\ell)} = 1\right\}$, which has again been calculated according to the update method described in Section 2.5.2. Note that the KL divergence

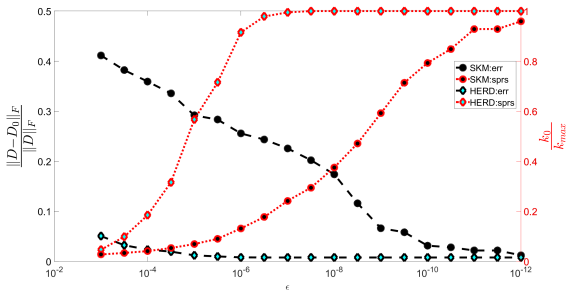

Figure 2.4: The relative error and sparsity incurred by the SKM-based and HERD-based matrices D_0 as a function of ϵ, KME case. Not included is the computation time, SKM achieves an error of 10^{-12} in 1.3 minutes, while HERD takes more than 30 minutes.

requires two density functions as input. We achieve this by projecting onto the simplex as indicated in 2.5.2. As in the KME case, we construct the dissimilarity matrix $(D_{\ell,\ell'}) := d_{KL}(\widehat{f_\ell}, \widehat{f_{\ell'}})$. Figures and Tables analogous to those for the KME case are shown in Chapter 3, with similar results.

Figure 2.2 shows us that the resulting embedded points using the sparse approximation keep the structure as of those using the full kernel mean. Notice also from Table 2.1 that the sparse approximation is many times faster than the full computation (about 400 times faster). Furthermore, the main computational investment is made in finding the elements of the sets $\mathcal{I}^{(\ell)}$, since the subsequent computation of D_0 is of negligible time.

In summary, for the case $\epsilon = 10^{-12}$, SKM takes only 1.3 minutes and the resulting embedding is almost identical to the full one, just as in Figure 2.3 shows for the herding case (which takes 30 minutes to compute). The SKM embedding is practically indistinguishable from that of the full one, being about 400 times faster than the full one and about 20 times faster than HERD.

2.6.2 Class Proportion Estimation

In this problem we are presented with labeled training data drawn from N distributions $\{P_1, \ldots, P_N\}$ and with further testing data drawn from a mixture of these distributions $P_0 = \sum_{i=1}^N \pi_i P_i$, where $\pi_i \geq 0$ and $\sum_i \pi_i = 1$. Each P_i represents a class in a multiclass classification problem and the goal is to estimate the mixture proportions $\{\pi_1, \ldots, \pi_N\}$ of each class in the unlabeled data set represented by P_0 (see [58, 59, 60]).

To do so we let $\widehat{\Psi}(P_\ell)$ represent the KME of P_ℓ for $0 \leq \ell \leq N$. We then find the proportions $\{\hat{\pi}_i\}_{i=1}^N$ that minimize the distance

$$\|\widehat{\Psi}(P_0) - \sum_{i=1}^N \pi_i \widehat{\Psi}(P_i)\|_{\mathcal{H}}^2,$$

where $\mathcal{H}$ is the RKHS of the kernel used to construct the KME. By setting the derivative to zero the optimal vector of proportions $\hat{\pi}_- := [\hat{\pi}_1, \ldots, \hat{\pi}_{N-1}]^T$, subject to $\sum_{i=1}^N \hat{\pi}_i = 1$ but not to $\hat{\pi}_i \geq 0$, satisfies

$$\hat{D}\hat{\pi}_- = \hat{e},$$

where

$$\hat{D}_{ij} = \left\langle \widehat{\Psi}(P_i) - \widehat{\Psi}(P_N), \widehat{\Psi}(P_j) - \widehat{\Psi}(P_N) \right\rangle_{\mathcal{H}}$$

and

$$\hat{e}_i = \left\langle \widehat{\Psi}(P_i) - \widehat{\Psi}(P_N), \widehat{\Psi}(P_0) - \widehat{\Psi}(P_N) \right\rangle_{\mathcal{H}}.$$

From here we can define

$$\hat{\pi} := \begin{bmatrix} \hat{\pi}_- \\ 1 - \sum_{i=1}^{N-1} \hat{\pi}_i \end{bmatrix}.$$

A parallel approach, using the KDE instead of the KME is shown in [59]. In that case the distance in $\mathcal{H}$ was changed to the L^2 distance.

Notice we have not enforced the constraint $\hat{\pi}_i \geq 0$, for $1 \leq i \leq N$. To do so a quadratic program can be set. For most of our simulations we did not encounter the necessity to do so, due to the fact that the true proportions we are approximating are also nonnegative, although this is an empirical observation without theoretical support. Therefore, for the few cases for which $\hat{\pi}$ lied outside of the simplex, we have projected onto it as described in [55].

We have used the handwritten digits data set MNIST, obtained from [61], which contains $60,000$ training images and $10,000$ testing images, approximately evenly distributed among its 10 classes (see [62] for details). We have only used the first five digits.

We present a comparison of the performance, measured by the ℓ_1 distance between the true π and the estimate $\hat{\pi}$, of the sparse KME compared to the full KME. We have done this for different values of π, meaning different locations of π inside the simplex. To do so, we sampled π from the simplex using the Dirichlet distribution with different concentration parameter ω. As a reminder to the reader, a small value of ω implies sparse values of π are most probable, $\omega = 1$ means any value of π is equally probable, and $\omega > 1$ means values of π for which all its entries are of similar value are most probable. We varied ω over the set $\{.1, .2, \ldots, 3.1\}$.

We have split the data in two and used the first half to estimate the kernel bandwidth through the following process. We first sample a true π, then we construct the KME and pick the bandwidth σ which minimizes $\|\pi - \hat{\pi}\|_{\ell_1}$. We performed the search on σ by using Matlab's function *fminbnd*. For both the SKM and the full case we allowed for 100 iterations. We have used the Gaussian kernel to create the sparse KME of the ℓ^{th} distribution, with sparsity level of $k_\ell = \sqrt{n_\ell}$, where n_ℓ is the size of the available sample from distribution ℓ. Since the α coefficients depend on σ, and for each set $\mathcal{I}^{(\ell)}$ we perform a search over several values of σ, we did not compute α iteratively as we constructed $\mathcal{I}^{(\ell)}$. Instead, once the construction of $\mathcal{I}^{(\ell)}$ was finished, we used the preconditioned conjugate gradient method to obtain α.

Once σ was estimated, we then accessed the second half of the data to test the performance for both the SKM and the full KME for different values of ω. The results are shown in Fig. 2.5. We have also plotted for perspective a "blind" estimation of π, which uniformly at random picks a vector $\hat{\pi}$. A comparison of the computation times for the sparse KME and the full KME is shown in Table 2.2, where we have averaged over all values of ω.

Notice from Table 2.2 that, in the SKM case, the estimation of σ takes about the same time as the computation of $\hat{\pi}$. This is due to the fact that the main bottleneck of the algorithm is the computation of the set $\mathcal{I}$ which is independent of σ. It is in the estimation of σ that the full kernel mean is many times slower than SKM, as seen in the Table, SKM is about 10 times faster for the whole process.

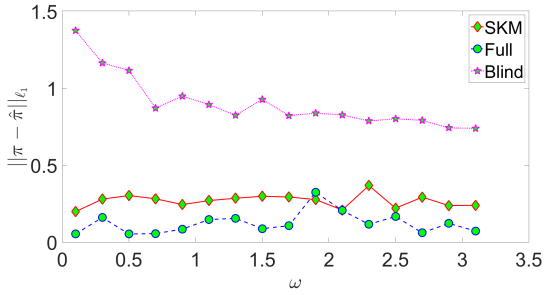

Figure 2.5: Class Proportion Estimation. ℓ_1 error of estimated proportions over a range of concentration parameters.

Table 2.2: Computation times for both full and sparse KME, averaged over all values of ω.

	σ estimation	$\hat{\pi}$ computation	Total
Full	110.4s	5.08s	1.9mins
SKM	5.9s	5.07s	11s

2.6.3 Mean-Shift Clustering

We have based this experiment on the mean-shift algorithm as described in [63]. This algorithm is used in several image processing tasks and we will use it in the context of image segmentation. The goal is to form a clustering of the image pixels into different segments.

Each pixel is represented by a 5-dimensional vector (3 dimensions to describe color, and 2 for the position in the image), and the distribution of these feature vectors is estimated by the KDE. Denote the image pixels as $\{x_i\}_{i=1}^n$, $x_i \in \mathbb{R}^5$. The mean-shift algorithm shifts each point lying on the surface of the density closer to its closest peak (mode). Given a starting point x, the algorithm iteratively shifts x closer to its mode until the magnitude of the shift is smaller than some quantity γ. The shift exerted on x at each iteration requires the computation of the gradient of the KDE at the current position, making mean-shift computationally expensive. Denote the shifted points as $\{y_i\}_{i=1}^n$. Once all points are shifted close to the different modes, then any clustering algorithm can be performed to find the clusters. A clustering algorithm is described in [63], based on merging the modes' neighborhoods which are close. We used a code following these guidelines found at [64], slightly modified by increasing the distance used for modes' neighborhoods to merge.

In our experiments we used a 500×487 image of a painting by Piet Mondrian (*Composition A*), and compared our algorithm with the full density estimation case. We chose k_{max} to be the largest integer smaller than $\sqrt{n}$ and we have used the method for auto-selecting k_0 outlined in Section 2.5.2, with $\epsilon = 10^{-8}$. We have used the Gaussian kernel and set the bandwidth according to Equation (18) in [65], which is specifically suggested for mode-based clustering. We compare the SKM approach to a method based on Locality Sensitive Hashing (LSH, see [66, 67]). This method finds for each point and with high probability its nearest neighbors. It then approximates the KDE locally by only using the effect from such neighbors. We chose 5 nearest neighbors and to implement LSH we used the Matlab version of LSH available at [68] (we have used the e2lsh scheme with three hash tables per picture). See [68, 69] for details on LSH.

We present two indicators to evaluate the performance between the clustering resulting from the full KDE and that resulting from the approximate KDE. In the following, let $\mathcal{B}$

Table 2.3: Time and Performance Comparison for Mean Shift algorithm.

	Time			Performance	
	Preparation	Mean Shift	Total	$d_i(\cdot, \mathcal{B})$	$\widehat{d}_H(\cdot, \mathcal{B})$
$\mathcal{B}$	0	2.1hrs	2.1hrs	0	0
$\mathcal{A}_{\text{SKM}}$	15.65s	39.12s	54.77s	0.006	0.015
$\mathcal{A}_{\text{LSH}}$	12.7s	7.1mins	7.32mins	0.034	0.02

be used to indicate that the full kernel density estimate has been used, while $\mathcal{A}$ indicates either the SKM or the LSH approaches. With a slight abuse of notation, let $\mathcal{A}$ and $\mathcal{B}$ also indicate their resulting clusterings.

Discrepancy Index. Our first performance measure, which we call the discrepancy index d_i, is somehow intuitive, and it describes the ratio of the number of vectors x_ℓ that the approximate methods shifted by more than δ away from their full method counterpart. δ is here some tolerance threshold, which we have set to three times the kernel bandwidth. More precisely, if $\{x_\ell\}_{\ell=1}^n$ indicate the picture pixels and $y_\ell^{\mathcal{A}}$, $y_\ell^{\mathcal{B}}$ are the shifted versions of x_ℓ according to density estimation methods $\mathcal{A}$ and $\mathcal{B}$ respectively, then

$$d_i(\mathcal{A}, \mathcal{B}) = \frac{1}{n} \sum_\ell \mathbf{1}_{\{\|y_\ell^{\mathcal{A}} - y_\ell^{\mathcal{B}}\| > \delta\}}.$$

Hausdorff Distance. The second performance measure, which describes the Hausdorff distance between clusterings, was obtained from [70] and is denoted by d_H. To define the Hausdorff distance, let P be a distribution on $\mathbb{R}^d$ (in our case, P is the distribution of the image pixels on $\mathbb{R}^5$). Furthermore, let $\mathcal{X}$ be the set of subsets of $\mathbb{R}^d$ such that the distance between two sets A and B is $\rho(A, B) := P(A \Delta B)$, where Δ is the symmetric difference (to be precise, we deal with equivalence classes, where two sets A and B are equivalent if $\rho(A, B) = 0$). Notice $\mathcal{X}$ is a metric space. Let $\mathcal{B} \subset \mathcal{X}$, and define $\rho(A, \mathcal{B}) := \min_{B \in \mathcal{B}} \rho(A, B)$. We interpret a subset $\mathcal{A}$ of $\mathcal{X}$ as a clustering, and an element A in $\mathcal{X}$ as a cluster. The Hausdorff distance between two clusterings is

$$d_H(\mathcal{A}, \mathcal{B}) = \max \left\{ \max_{A \in \mathcal{A}} \ \rho(A, \mathcal{B}), \ \max_{B \in \mathcal{B}} \ \rho(B, \mathcal{A}) \right\}.$$

In words, d_H measures the furthest distance between elements of $\mathcal{A}$ to the clustering $\mathcal{B}$ and elements of $\mathcal{B}$ to the clustering $\mathcal{A}$ (that is, d_H measures the less overlap between clusters of different clusterings, as measured by P). Since we don't have access to P, the empirical version of d_H proposed in [70] is obtained by replacing P by the empirical probability measure. Letting $\widehat{\rho}(A, \mathcal{B}) := \min_{B \in \mathcal{B}} \frac{1}{n} \sum_{i=1}^n \mathbf{1}_{\{A \Delta B\}}(x_i)$, we have

$$\widehat{d}_H(\mathcal{A}, \mathcal{B}) = \max \left\{ \max_{A \in \mathcal{A}} \ \widehat{\rho}(A, \mathcal{B}), \ \max_{B \in \mathcal{B}} \ \widehat{\rho}(B, \mathcal{A}) \right\}.$$

We use this latter quantity to measure the SKM performance.

The results are presented in Table 2.3. In the table $\mathcal{B}$ indicates the full kernel density estimate has been used, $\mathcal{A}_{\text{SKM}}$ indicates the k-center based algorithm and $\mathcal{A}_{\text{LSH}}$ the LSH method. Note that both the SKM and the LSH approach present significant computational advantages. The SKM approach, however, manages to be faster while incurring half the discrepancy of the LSH and about the same Hausdorff distance.

2.6.4 Comparison with Other Subset Selection Strategies

To further illustrate the performance of SKM, we look at the sparsity required to achieve a given accuracy ϵ, that is, the smallest value k_0/k_{max} for which the quantity $\bar{E}_{|\mathcal{I}|} := \|\bar{z} - z_{\mathcal{I}}\|^2 / \|\bar{z}\|^2$ is smaller than ϵ, where $k_0 = |\mathcal{I}|$, (see Section 2.5.2). Note we selected $\bar{E}_{|\mathcal{I}|}$ as opposed to the term $E_{|\mathcal{I}|}$ used for autoselection because $\bar{E}_{|\mathcal{I}|}$ is more interpretable and for these experiments we are not interested in efficient autoselection. We have applied our method for 11 distinct benchmark datasets, listed in Table 2.5. We present these sparsity values for $\epsilon = 10^{-3}$ and also the corresponding values for four other methods:1) RAND, which chooses the set $\mathcal{I}$ uniformly at random, 2) L2, which chooses $\mathcal{I}$ by sampling according to the squared norm of the columns of the kernel matrix (see [49]), 3) KMEANS which picks as the representative set not a subset of the data but the results from the k-means algorithm and constructs the kernel matrix according to these (see again [49]), and 4) HERD, which uses kernel herding (see [30]) to select $\mathcal{I}$ and $\alpha_{\mathcal{I}}$. Table 2.4 shows the time and memory complexities for these algorithms. The Wilcoxon signed rank test was performed pairwise comparing the performance of SKM to the other methods. The p-values for SKM with respect to RAND, L2, KMEANS and HERD are respectively .024, .001, .019, and .001 favoring SKM. Note that RAND and KMEANS have similar performance, and in fact the p-value between the two is .64. We also present the complete error plot for the banana dataset in Figure 2.6. Note that in this case other approximations show an initial advantage because they are more likely to pick elements from dense areas, which for small values of $|\mathcal{I}|$ represents better the full distribution. However, as the size of $\mathcal{I}$ increases, the fine structure (e.g., the distribution tails) is better captured by SKM, since the k-center algorithm picks points far apart from each other. Note that Table 2.5 shows that SKM achieves better sparsity for a given accuracy. Equivalently, we can say SKM achieves better accuracy for a fixed sparsity level. In Chapter 3, we also report the time required to achieve an accuracy of 10^{-3} and find a similar advantage for SKM.

The sparse approximation strategy proposed in this paper can be a valid density if the α_i's are set to satisfy $\alpha_i \geq 0$ and $\sum_i \alpha_i = 1$. Therefore, we also evaluate the performance of the proposed sparse approximation according to the KL divergence, a common metric between distributions whose arguments must be density functions.

For the same benchmark data sets listed above, we computed the KL divergences $D(\bar{z}\|z_{\mathcal{I}})$ and $D(z_{\mathcal{I}}\|\bar{z})$ between the sparse and the full kernel mean. We used the autoselection scheme proposed in Section 2.5.2 with $\epsilon = 10^{-7}$, and projected the resulting α onto the simplex to ensure we have a valid probability distribution. We have chosen a Gaussian kernel and used the Jaakkola heuristic [71] to compute the bandwidth. We compare the performance of SKM to that of RAND, which, as seen above, is similar to that of KMEANS. We have performed the Wilcoxon signed rank test [72] to determine if there is a significant advantage of the SKM. The test for both the case $D(\bar{z}\|z_{\mathcal{I}})$ and the case $D(z_{\mathcal{I}}\|\bar{z})$ yields a p-value of 0.0186, favoring the SKM method. The results are shown in Table 2.6. To understand this, note that in the extreme case in which one density is zero in a particular region while the other is positive, the KL divergence is infinite, so the KL divergence highly penalizes very low density approximations to positive density regions. The SKM accurately approximates low density regions since it selects outliers, while the random selection approach picks mostly points in populated regions.

Table 2.4: Time complexity and memory comparison among selected methods. n is the data size, k the approximation size, and T the number of iterations for the k-means algorithm from MATLAB.

Method	Selection time	α computation time	Memory
SKM	nk	k^3	k^2
RAND	$k \log n$	k^3	k^2
L2	n^2	$\leq k^3$	k^2
KMEANS	nkT	k^3	k^2
HERD	$n^2 + k^3$	(combined with selection)	k^2

Table 2.5: Sparsity level required for an accuracy of 10^{-3}.

	SKM	RAND	L2	KMEANS	HERD
banana	0.6443	0.8905	1	0.9068	1
image	0.7367	0.7933	1	0.8540	1
ringnorm	0.6603	0.6779	1	0.6787	.8325
breast-cancer	0.6768	0.6062	1	0.5905	1
heart	0.7455	0.8386	1	0.8384	1
thyroid	0.7257	0.9410	1	0.9383	1
diabetes	0.6405	0.7885	1	0.7859	1
german	0.5648	0.5900	1	0.5889	.8
twonorm	0.6202	0.5962	0.9863	0.6002	.7725
waveform	0.6557	0.6876	1	0.6843	1
iris	0.7667	0.8536	1	0.8782	1

2.6.5 Other Results

We have performed additional experiments that explore the performance of SKM as dimension increases. These results have been placed in Chapter 3. In conclusion, the results suggest that the sparsity required to achieve a given accuracy increases as a function of dimension and decreases as a function of bandwidth. We have also compared the performance of SKM for the Laplacian and the Student-t kernels. In general they both exhibit a similar performance as for the Gaussian kernel, in terms of relative error $\bar{E}$. For the Euclidean embedding and class proportion estimation experiments, however, it is harder to set an effective bandwidth for the Student-t kernel.

2.7 Conclusion

We have provided a method to rapidly and accurately build a sparse approximation of a kernel mean. We derived an incoherence based bound on the approximation error and recognized that, for abroad class of kernels, including translation invariant kernels, its minimization is equivalent to solving the k-center problem either on the feature space or the Euclidean space where the data lies. If desired, our construction of the sparse kernel mean may be slightly modified to provide a valid density function, which is important in some applications. Hence, the algorithm is versatile in that it works for both kinds of kernel means: the KDE and the KME. Our method also naturally lends itself to a sparsity auto-selection scheme.

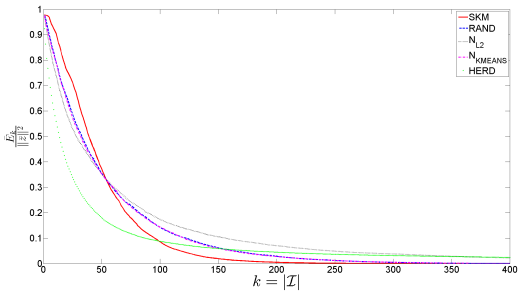

Figure 2.6: Error comparison among different methods for the banana data set. SKM incurs a larger initial error but quickly decreases as it captures more of the fine structure of the distribution. Note that RAND and $\mathrm{N}_{\mathrm{KMEANS}}$ perform very similarly, this can also be seen in table 2.5.

Table 2.6: Values of $\mathrm{D}(\bar{z}\|z_{\mathcal{I}})$ and $\mathrm{D}(z_{\mathcal{I}}\|\bar{z})$ for different data sets.

	$\mathrm{D}(\bar{z}\|z_{\mathcal{I}})$		$\mathrm{D}(z_{\mathcal{I}}\|\bar{z})$	
	RAND	SKM	RAND	SKM
banana	0.092597	0.001805	0.129183	0.001613
image	0.451205	0.041305	0.212585	0.061584
ringnorm	0.003983	0.031736	0.009253	0.02853
breast-cancer	0.358253	0.002546	0.345895	4.56E-05
heart	0.001918	6.35E-16	0.005228	2.91E-16
thyroid	0.177317	0.000594	0.034616	0.000289
diabetes	0.031366	0.005474	0.014635	0.000102
german	0.008711	0.003855	0.008742	0.00203
twonorm	0.000131	0.000243	4.59E-05	0.000372
waveform	0.011473	0.000177	0.015064	0.000404
iris	0.043924	0.000395	0.022519	0.000104

We showed its computational advantages and its performance qualities in three specific applications. First, Euclidean embedding of distributions (for both KDE and KME), in which, for the KDE case, a valid density is needed to compute the KL divergence. Second, class proportion estimation (for the KME), which presents the amortization advantages of the SKM approach, in this case with respect to the bandwidth σ. Finally, mean-shift clustering (for the KDE), in which with less computation time than the LSH-based approach, it performs better with respect to the discrepancy index and similar with respect to the Hausdorff distance. In most instances the proposed sparse kernel mean method has shown to be orders of magnitude faster than the approach based on the full kernel mean. Furthermore, we compared its performance in terms of error, sparsity, and time with respect to four other subset selection schemes for several benchmark datasets. We find that, with statistical significance, SKM outperforms these methods. Finally, we also observed its performance in different settings, concerning dimension and kernel variability. These latter results are presented in detail in Chapter 3.

Chapter 3

Further results for SKM

In this chapter we supplement many of the experimental results of Chapter 2, and provide a theoretical result related to the main theorem. In Section 3.1 we investigate the influence of bandwidth and dimensionality on the performance of SKM. We proceed to compare the performance among different kernels in Section 3.2. Section 3.3 contains the results of SKM on the embedding of distributions experiments of Section 2.6.1 of Chapter 2 for the KDE case. To supplement the results from section 2.6.4 of Chapter 2, in Section 3.4 we present Table 3.7, which contains a time comparison among different methods. Finally, in Section 3.5 we present a bound on the error between a kernel mean and a sparse approximation which is constrained to have nonnegative weights.

3.1 Dimension and Bandwidth

We first look at the performance of SKM, as measured by the error quantity $\bar{E}_k$ (defined in Section 2.5.2 of Chapter 2), for similar datasets of different dimension. We have generated five data sets of dimensions 1, 2, 10, 50, and 100, where the data points are randomly iid sampled from a standard Gaussian distribution (case 1), and the t-distribution with 1 degree of freedom (case 2). Figure 3.1 contains the error plots for both cases and for two heuristics for computing the bandwidth parameter: Scott's rule of thumb and Jaakkola's heuristic (see [12, 71]). As seen in the figures, for these bandwidths the relationship between performance and dimension is not straightforward, since in some instances higher dimensions yield a better error, while in other instances lower dimensions result in smaller error. To investigate this phenomena further, we look specifically at the relative error $\bar{E}$ committed at a sparsity level of 10% for a range of dimension and bandwidth parameters, listed in Table 3.1 for case 1. The values of the bandwidth parameter are 10 evenly spaced points between .01 and 12, the minimum and maximum bandwidths across all dimensions picked by the heuristics above for the data in question. The data size is $n = 500$. Figure 3.2 serves as a visualization aid for Table 3.1. A similar comparison is presented on Table 3.2 but regarding the sparsity needed for a relative error of 10^{-3}, with visualization aid in Figure 3.3. Table 3.1 suggests that, in case 1 and for fixed dimension, increasing the bandwidth reduces the error for 10% sparsity, while for small fixed bandwidth increasing dimension increases the relative error. Table 3.2 suggests that for fixed dimension higher bandwidth requires smaller k for an error of $\epsilon = 10^{-3}$ and for fixed bandwidth higher dimension requires a larger k to achieve the same error.

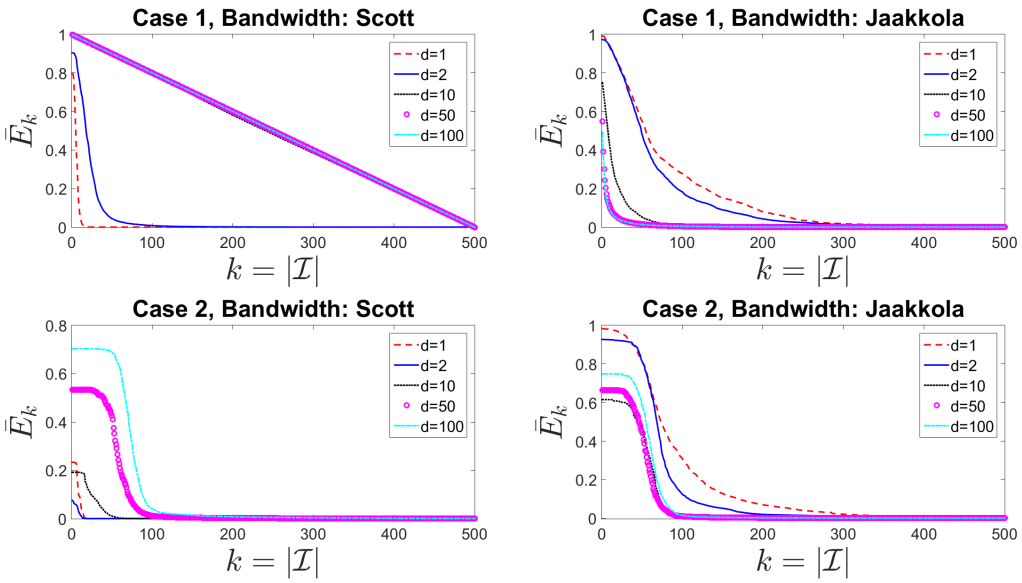

Figure 3.1: Error comparison among different dimensions. Case 1 indicates the data is drawn from a multivariate standard Gaussian distribution, while case 2 indicates the t-distribution with one degree of freedom. Ṣcott" and J̀aakkola" indicate, respectively, that the bandwidth has been chosen according to Scott's rule of thumb or Jaakkola's heuristic.

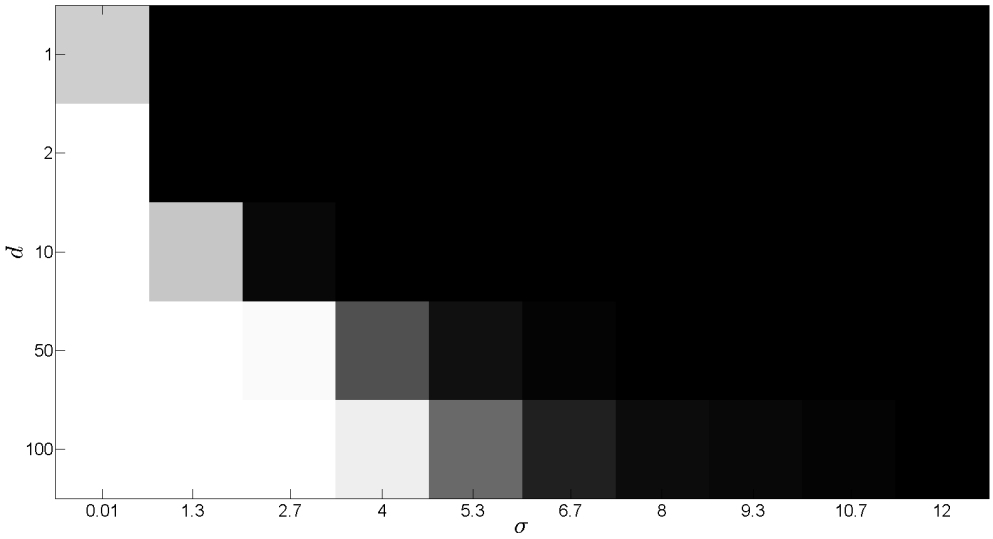

Figure 3.2: Visualization aid for Table 3.1

Table 3.1: d vs σ comparison of the relative error committed at a sparsity level of 10%. Gaussian case. The values of σ are evenly distributed between the minimum and maximum values obtained by the heuristics of Scott and Jaakkola.

	$\sigma = .01$	$\sigma = 1.34$	$\sigma = 2.67$	$\sigma = 4$	$\sigma = 5.34$	$\sigma = 6.67$	$\sigma = 8$	$\sigma = 9.34$	$\sigma = 10.67$	$\sigma = 12$
$d = 1$	0.7236	0.0001	0.0004	0.0002	0.0001	0.0002	0.0002	0.0002	0.0002	0.0001
$d = 2$	0.902	0.0005	0.0004	0.0004	0.0004	0.0004	0.0003	0.0005	0.0004	0.0003
$d = 10$	0.9	0.6936	0.0382	0.0047	0.0012	0.0009	0.0009	0.0009	0.0008	0.0008
$d = 50$	0.9	0.9	0.883	0.2874	0.0699	0.0242	0.0106	0.0062	0.0037	0.0025
$d = 100$	0.9	0.9	0.9005	0.8342	0.3681	0.1235	0.0540	0.0291	0.0173	0.0118

Table 3.2: d vs σ comparison of the sparsity required for an accuracy of $\epsilon = 10^{-3}$. Gaussian case. The values of σ are evenly distributed between the minimum and maximum values obtained by the heuristics of Scott and Jaakkola.

	$\sigma = .01$	$\sigma = 1.34$	$\sigma = 2.67$	$\sigma = 4$	$\sigma = 5.34$	$\sigma = 6.67$	$\sigma = 8$	$\sigma = 9.34$	$\sigma = 10.67$	$\sigma = 12$
$d = 1$	0.5802	0.011	0.007	0.0062	0.0056	0.0046	0.0048	0.004	0.004	0.004
$d = 2$	0.9940	0.0298	0.0176	0.0104	0.0088	0.0082	0.0064	0.005	0.0048	0.0044
$d = 10$	1	0.9296	0.395	0.1854	0.1074	0.0606	0.0408	0.0286	0.0234	0.0196
$d = 50$	1	1	0.998	0.9404	0.7836	0.585	0.4246	0.2958	0.215	0.1702
$d = 100$	1	1	1	0.996	0.9684	0.8914	0.7694	0.6342	0.5256	0.427

3.2 Kernel Choice

To illustrate the performance of SKM for different kernels, we show a comparison of the Gaussian, the Laplacian, and the Student-type kernels. For the student kernel we set its parameter $\alpha = (d + 1)/2$. Plots of error vs sparsity, sparsity vs accuracy, and time vs accuracy for the banana dataset, averaged over 50 runs, are shown in figures 3.4, 3.5, and 3.6. The bandwidth was chosen according the the Jaakkola heuristic. The error performance is comparable among these kernels, especially for the Gaussian and the Student-type kernels. As we can see from the figures, the Gaussian kernel requires relative less sparsity to achieve a given accuracy level, and therefore is also the fastest. We can then conclude that, at least for a standard bandwidth selection scheme, the Gaussian kernel is a reasonable choice.

Further, we ran our main experiments (Sections 2.6.1, 2.6.2, and 2.6.3 of Chapter 2) for the Laplacian and Student-type kernels, with similar results. The relevant information is shown in figures 3.7, 3.8, 3.9, 3.10 and Tables 3.3 and 3.4.

3.3 Euclidean Embedding of Distributions: KDE case

Here we present the results of our method's performance for the Euclidean embedding experiment described in Section 2.6.1 of Chapter 2 for KDE case. To compute the KL divergence we split the data in two, use the first half for estimation of the KDE, and the second half for estimation of the KL divergence. We have chosen $k_\ell = 3\sqrt{n_\ell}$ for each ℓ, as in the KME case, and used the same stopping criterion with $\epsilon = 10^{-9}$. The results are shown in Figure 3.11 and tables 3.5 and 3.6, the plot of $\frac{\|D-D_0\|_F}{\|D\|_F}$ for several values of ϵ is shown in Fig. 3.12.

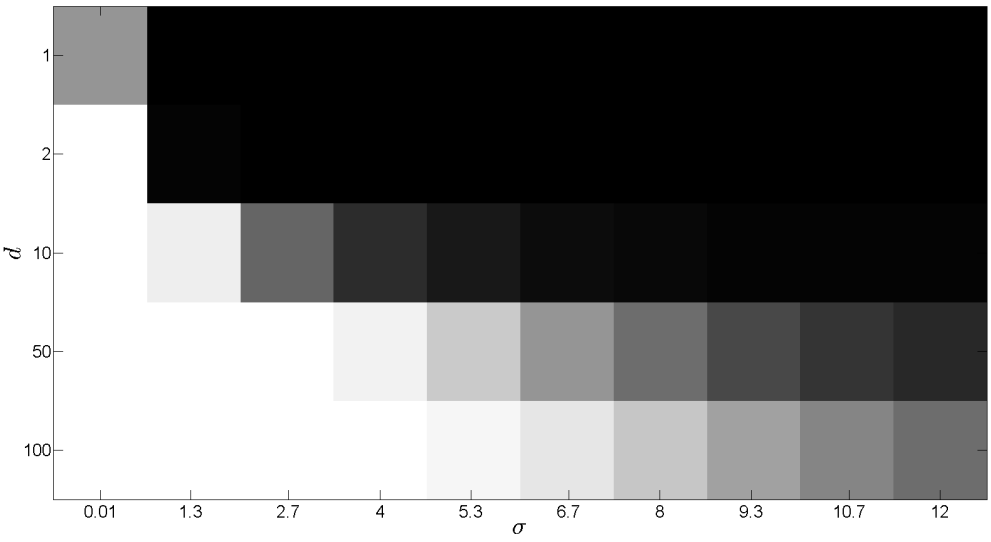

Figure 3.3: Visualization aid for Table 3.2

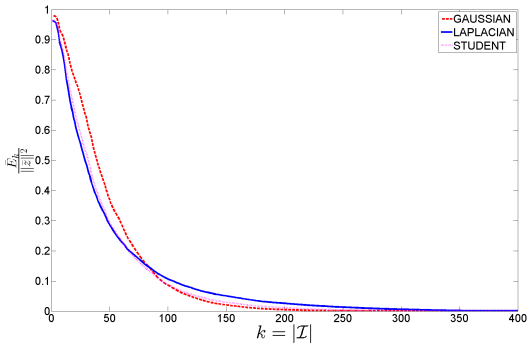

Figure 3.4: Error comparison among different kernels for the banana data set. The bandwidth selection method is the Jaakkola heuristic.

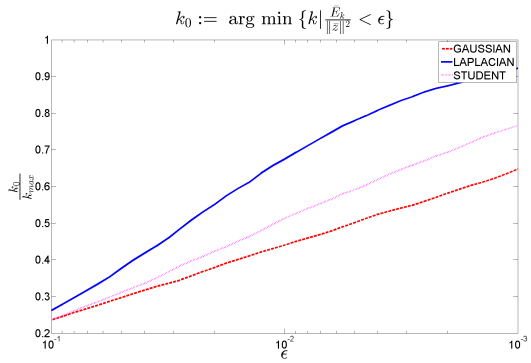

Figure 3.5: Sparsity comparison among different kernels for the banana data set. The bandwidth selection method is the Jaakkola heuristic.

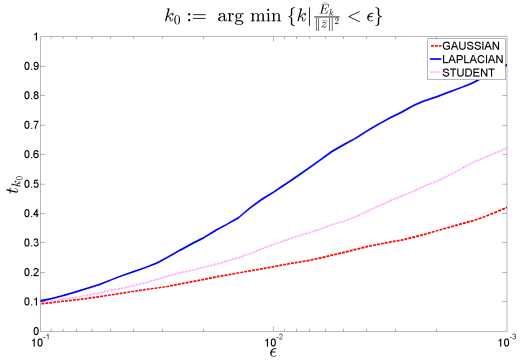

Figure 3.6: Time comparison among different kernels for the banana data set. The bandwidth selection method is the Jaakkola heuristic.

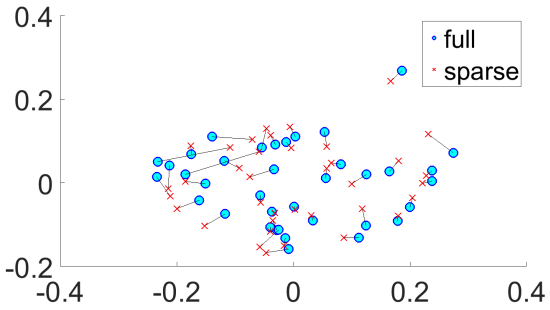

Figure 3.7: 2-dimensional representation of flow cytometry data for the Laplacian kernel and the KME case. As it is the case with the Gaussian kernel, the approximation is very accurate.

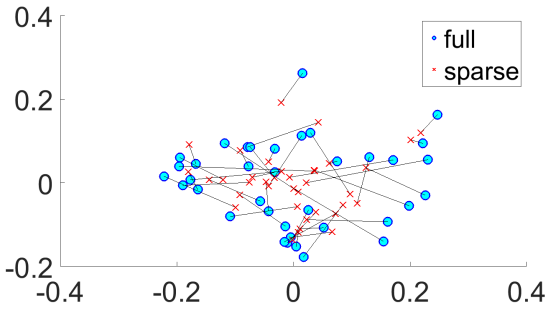

Figure 3.8: 2-dimensional representation of flow cytometry data for the Student-type kernel and the KME case. The overall shape is preserved, but it is not very accurate, probably due to the suboptimal choice of the kernel's bandwidth.

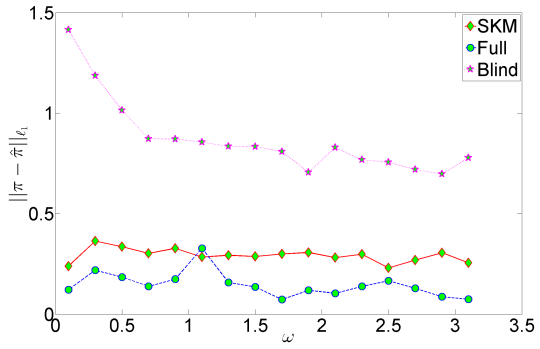

Figure 3.9: Class Proportion Estimation - Laplacian. ℓ_1 error of estimated proportions for the Laplacian kernel. The behavior shown is very similar to the case of the Gaussian kernel.

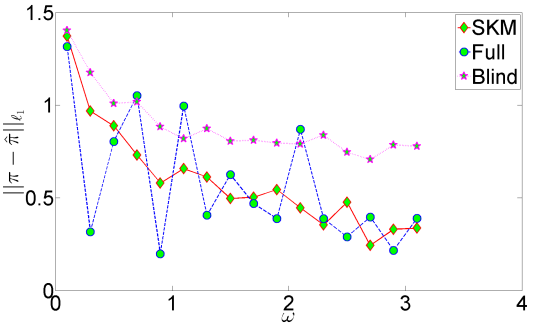

Figure 3.10: Class Proportion Estimation - Student. ℓ_1 error of estimated proportions for the Student-type kernel. Note that in this case the full method exhibits an erratic behavior, this is because the number of allowed iterations to find the bandwidth is not enough to find an appropriate bandwidth.

Table 3.3: Time and Performance Comparison for Mean Shift algorithm. Laplacian kernel case.

	Time			Performance	
	Preparation	Mean Shift	Total	$d_i(\cdot, \mathcal{B})$	$\widehat{d}_H(\cdot, \mathcal{B})$
$\mathcal{B}$	0	3.33hrs	3.33hrs	0	0
$\mathcal{A}_{SKM}$	6.7mins	1.7mins	8.4mins	0.035	0.04
$\mathcal{A}_{LSH}$	23.1s	9.94mins	10.3mins	0.082	0.03

Table 3.4: Time and Performance Comparison for Mean Shift algorithm. Cauchy kernel case.

	Time			Performance	
	Preparation	Mean Shift	Total	$d_i(\cdot, \mathcal{B})$	$\widehat{d}_H(\cdot, \mathcal{B})$
$\mathcal{B}$	0	2.84hrs	2.84hrs	0	0
$\mathcal{A}_{SKM}$	7.98mins	1.85mins	9.83mins	0.007	0.04
$\mathcal{A}_{LSH}$	25.5s	7.75mins	8.17mins	0.01	0.02

3.4 Time

In this section we supplement Section 2.6.4 of Chapter 2. Under the same setup, we present Table 3.7, which is analogous to Table 2.5, but instead of sparsity we look at the times required to obtain an accuracy of 10^{-3}. Note SKM is sometimes faster than RAND because it is able to achieve the desired accuracy with smaller k, meaning it spends less time computing α's.

3.5 Constrained Projection Bound

In this section we present a bound on the error of the projection onto $S_\Delta = \left\{\sum_{i\in\mathcal{I}} \alpha_i z_i \mid \sum \alpha_i = 1, \alpha_i \geq 0\right\}$. We conclude that, although not as tight as the bounds in Chapter 2, it also suggests the maximization of $\nu_\mathcal{I}$.

Let $\mathcal{H}$ be a Hilbert space and $\{z_1, \cdots, z_n\} \subset \mathcal{H}$ such that $C \geq \langle z_i, z_j\rangle \geq 0$ for all (i, j), let $\bar{z}$ indicate the mean of these vectors. Furthermore let $S_\Delta = \left\{\sum_{i\in\mathcal{I}} \alpha_i z_i \mid \sum \alpha_i = 1, \alpha_i \geq 0\right\}$, and $P\bar{z}$ the projection of $\bar{z}$ onto this set. The following two lemmas will be proven at the end.

Lemma 2.

$$-\left\|P\bar{z}\right\|^2 \leq -\frac{1}{C}\left(\max_{z\in S_\Delta}\langle \bar{z}, z\rangle\right)^2 + \frac{2}{C}\max\langle \bar{z}, z\rangle\langle \bar{z} - P\bar{z}, P\bar{z}\rangle - \frac{1}{C}\langle \bar{z} - P\bar{z}, P\bar{z}\rangle^2$$

Later it will be convenient to express this result as

$$-\left\|P\bar{z}\right\|^2 \leq -\frac{1}{C}M^2 + \frac{2}{C}MT - \frac{1}{C}T^2.$$

where $T = \langle \bar{z} - P\bar{z}, \bar{z}\rangle$ *and* $M = \max_{z\in S_\Delta}\langle \bar{z}, z\rangle$.

Lemma 3.

$$\max_{z\in S_\Delta}\langle \bar{z}, z\rangle \geq \frac{n-k}{n}\min_{i\notin\mathcal{I}}\max_{j\in\mathcal{I}}\langle z_i, z_j\rangle.$$

Table 3.5: Sparsity with respect to ϵ. The $\frac{k_0}{k}$ values are shown for the KDE case.

ϵ	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
KDE case	0.2228	0.3615	0.5822	0.7724	0.8964

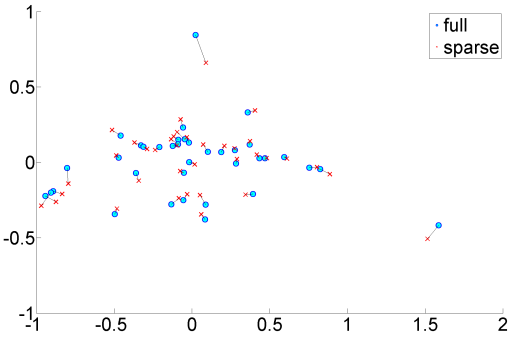

Figure 3.11: 2-dimensional representation of flow cytometry data - KDE case. Each point represents a patient's distribution. The embeddings were obtained by applying ISOMAP to distances between KDEs as measured by the KL divergence.

Theorem 2.

$$\|\bar{z} - P\bar{z}\| \leq \sqrt{2\left(C - \left(1 - \frac{|\mathcal{I}|}{n}\right)\nu_{\mathcal{I}}\right)}.$$

Proof.

$$\begin{aligned}
\|\bar{z} - P\bar{z}\|^2 &= \langle \bar{z} - P\bar{z}, \bar{z} - P\bar{z} \rangle \\
&= \langle \bar{z}, \bar{z} \rangle - 2\langle \bar{z}, P\bar{z} \rangle + \langle P\bar{z}, P\bar{z} \rangle \\
&= \langle \bar{z}, \bar{z} \rangle - 2\langle \bar{z} - P\bar{z} + P\bar{z}, P\bar{z} \rangle + \langle P\bar{z}, P\bar{z} \rangle \\
&= \langle \bar{z}, \bar{z} \rangle - 2\langle \bar{z} - P\bar{z}, P\bar{z} \rangle - \langle P\bar{z}, P\bar{z} \rangle .
\end{aligned}$$

Table 3.6: Time Comparison for the Euclidean embedding of the Flow Cytometry dataset - KDE case.

	k-center	D computation	Total
Full	0	2.18 hrs	2.18 hrs
SKM	5mins	2mins	7mins

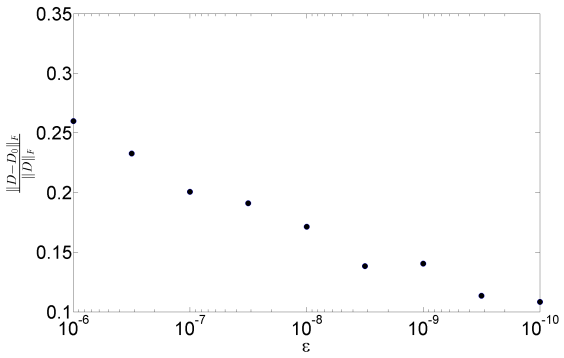

Figure 3.12: The relative error incurred by the SKM-based matrix D_0 as a function of ϵ, averaged over 10 runs - KDE case. The average k-center and D_0 computation times range from 1 to 7.25 minutes, and from 35 to 160 seconds, respectively. The average ratio k_0/k_{max} ranges from 0.22 to 0.9.

so, using the result and notation from lemma 2, we have

$$\begin{aligned}
\|\bar{z} - P\bar{z}\|^2 &= \|\bar{z}\|^2 - 2T - \|P\bar{z}\|^2 \\
&\leq \|\bar{z}\|^2 - 2T - \frac{1}{C}M^2 + \frac{2}{C}MT - \frac{1}{C}T^2 \\
&= \|\bar{z}\|^2 - \frac{1}{C}M^2 - 2T\left(1 - \frac{M}{C}\right) - \frac{T^2}{C} \\
&= \|\bar{z}\|^2 - \frac{1}{C}M^2 + \frac{1}{C}(C-M)^2 - \frac{1}{C}(C-M)^2 - \frac{2}{C}T(C-M) - \frac{T^2}{C} \\
&= \|\bar{z}\|^2 - \frac{1}{C}M^2 + \frac{1}{C}(C-M)^2 - \frac{1}{C}\left[(C-M)^2 + 2T(C-M) + T^2\right] \\
&= \|\bar{z}\|^2 - \frac{1}{C}M^2 + \frac{1}{C}(C-M)^2 - \frac{1}{C}\left((C-M) + T\right)^2 \\
&\leq \|\bar{z}\|^2 - \frac{1}{C}M^2 + \frac{1}{C}(C-M)^2 \\
&= \|\bar{z}\|^2 - \frac{1}{C}M^2 + C - 2M + \frac{1}{C}M^2 \\
&= \|\bar{z}\|^2 + C - 2M \\
&\leq 2(C-M).
\end{aligned}$$

Finally, by lemma 3 we have

$$\|\bar{z} - P\bar{z}\|^2 \leq 2\left(C - \left(1 - \frac{|\mathcal{I}|}{n}\right)\nu_{\mathcal{I}}\right),$$

which justifies the maximization of $\nu_{\mathcal{I}}$. □

Table 3.7: Time in seconds required for an accuracy of 10^{-3}. The symbol (-) indicates the accuracy was never achieved

	SKM	RAND	N_{L2}	N_{KMEANS}	HERD
banana	0.3813	0.6796	6.5807	0.6591	-
image	17.6782	12.2856	238.3666	6.8079	-
ringnorm	0.4186	0.4116	5.7902	0.3580	.2443
breast-cancer	0.0735	0.0666	0.6592	0.0769	-
heart	0.0619	0.0739	0.4521	0.0910	-
thyroid	0.0399	0.0593	0.1874	0.0685	-
diabetes	0.6195	0.8336	9.1065	0.6702	-
german	1.4882	1.4861	37.1124	1.1943	1.1147
twonorm	0.3802	0.3207	6.9767	0.3063	.2286
waveform	0.4059	0.4195	6.8953	0.3961	-
iris	0.0474	0.0575	0.2873	0.0761	-

Proofs of Lemmas

In the following lemma, "closed" is meant in the topological sense, not the algebraic sense.

Lemma 4 (Luenberger pg. 69)**.** *Let s^* be a vector in a Hilbert space $\mathcal{H}$ and let S be a closed convex subset of $\mathcal{H}$. Then there is a unique vector $s_0 \in S$ such that*

$$\|s^* - s_0\| \leq \|s^* - s\|$$

for all $s \in S$. Furthermore a necessary and sufficient condition that s_0 be the unique minimizing vector is that $\langle s^* - s_0, s - s_0 \rangle \leq 0$ for all $s \in S$.

Proof. See [**?**]. □

Proof of Lemma 2. Since $S_{\mathcal{I}}$ is (topologically) closed, then, for any $z \in S_{\Delta}$, we have:

$$\begin{aligned}
\langle \bar{z}, z \rangle &= \langle \bar{z} - P\bar{z} + P\bar{z}, z \rangle \\
&= \langle P\bar{z}, z \rangle + \langle \bar{z} - P\bar{z}, z \rangle \\
&\leq \|P\bar{z}\| \, \|z\| + \langle \bar{z} - P\bar{z}, z - P\bar{z} + P\bar{z} \rangle \\
&= \|P\bar{z}\| \, \|z\| + \langle \bar{z} - P\bar{z}, z - P\bar{z} \rangle + \langle \bar{z} - P\bar{z}, P\bar{z} \rangle \\
&\leq \|P\bar{z}\| \, \|z\| + \langle \bar{z} - P\bar{z}, P\bar{z} \rangle \\
&\leq \|P\bar{z}\| \sqrt{C} + \langle \bar{z} - P\bar{z}, P\bar{z} \rangle .
\end{aligned}$$

For second inequality see Lemma 4. Also, note

$$\begin{aligned}
\max \langle \bar{z}, z \rangle &\leq \|P\bar{z}\| \sqrt{C} + \langle \bar{z} - P\bar{z}, P\bar{z} \rangle \\
&\rightarrow \\
\max \langle \bar{z}, z \rangle - \langle \bar{z} - P\bar{z}, P\bar{z} \rangle &\leq \|P\bar{z}\| \sqrt{C} \\
&\rightarrow \\
\left(\max \langle \bar{z}, z \rangle - \langle \bar{z} - P\bar{z}, P\bar{z} \rangle \right)^2 &\leq \|P\bar{z}\|^2 C \\
&\rightarrow \\
\|P\bar{z}\|^2 C &\geq \left(\max \langle \bar{z}, z \rangle \right)^2 - 2 \max \langle \bar{z}, z \rangle \langle \bar{z} - P\bar{z}, P\bar{z} \rangle + \langle \bar{z} - P\bar{z}, P\bar{z} \rangle^2
\end{aligned}$$

where the third line follows since

$$\begin{aligned}\langle \bar{z} - P\bar{z}, P\bar{z}\rangle &= \langle \bar{z}, P\bar{z}\rangle - \langle P\bar{z}, P\bar{z}\rangle \\ &\leq \langle \bar{z}, P\bar{z}\rangle \\ &\leq \max_{z\in S_\Delta} \langle \bar{z}, z\rangle ,\end{aligned}$$

and therefore $\max \langle \bar{z}, z\rangle - \langle \bar{z} - P\bar{z}, P\bar{z}\rangle$ is nonnegative. So we have that

$$\begin{aligned}-\|P\bar{z}\|^2 &\leq -\frac{1}{C}\left(\max \langle \bar{z}, z\rangle\right)^2 + \frac{2}{C}\max \langle \bar{z}, z\rangle \langle \bar{z} - P\bar{z}, P\bar{z}\rangle - \frac{1}{C}\langle \bar{z} - P\bar{z}, P\bar{z}\rangle^2 \\ &\leq -\frac{1}{C}M^2 + \frac{2}{C}MT - \frac{1}{C}T^2.\end{aligned}$$

□

Proof of Lemma 3. Let $\Delta = \{\alpha \mid \sum \alpha_i = 1, \alpha_i \geq 0\}$, then

$$\begin{aligned}\max_{z\in S_\Delta} \langle \bar{z}, z\rangle &= \max_{\alpha\in\Delta} \left\langle \bar{z}, \sum_{i\in\mathcal{I}} \alpha_i z_i \right\rangle \\ &\geq \max_{i\in\mathcal{I}} \langle \bar{z}, z_i\rangle \\ &= \max_{i\in\mathcal{I}} \frac{1}{n}\sum_{j\in[n]} \langle z_j, z_i\rangle \\ &\geq \frac{1}{n}\max_{i\in\mathcal{I}} \left[\sum_{j\in\mathcal{I}} \langle z_i, z_j\rangle + \sum_{j\notin\mathcal{I}} \langle z_i, z_j\rangle\right] \\ &\geq \frac{1}{n}\max_{i\in\mathcal{I}} \sum_{j\notin\mathcal{I}} \langle z_i, z_j\rangle \\ &\geq \frac{1}{n}\max_{i\in\mathcal{I}} \sum_{j'\notin\mathcal{I}} \min_{j\notin\mathcal{I}} \langle z_i, z_j\rangle \\ &\geq \frac{n-k}{n}\max_{i\in\mathcal{I}}\min_{j\notin\mathcal{I}} \langle z_i, z_j\rangle .\end{aligned}$$

Note that the third inequality is true since, if $z_\ell = \arg\max_{i\in\mathcal{I}} \sum_{j\notin\mathcal{I}} \langle z_i, z_j\rangle$, then

$$\begin{aligned}\max_{i\in\mathcal{I}} \sum_{j\notin\mathcal{I}} \langle z_i, z_j\rangle &= \sum_{j\notin\mathcal{I}} \langle z_\ell, z_j\rangle \\ &\leq \sum_{j\notin\mathcal{I}} \langle z_\ell, z_j\rangle + \sum_{j\in\mathcal{I}} \langle z_\ell, z_j\rangle \\ &= n\langle \bar{z}, z_\ell\rangle \\ &\leq n\max_{i\in\mathcal{I}} \langle \bar{z}, z_i\rangle ,\end{aligned}$$

where the first inequality is true by the assumption $\langle z_i, z_j\rangle \geq 0 \ \forall\ (i,j)$. □

Chapter 4

Consistent Kernel Density Estimation with Non-Vanishing Bandwidth

Overview

The KDE is a widely used and thoroughly analyzed nonparametric estimator of a probability density function. Its convergence to the through density requires the bandwidth to approach zero as the sample size approaches infinity. In this section we present the fixed bandwidth kernel density estimator (fbKDE), which is a weighted KDE. We prove the fbKDE, with a fixed bandwidth, consistently estimates square integrable densities and provide convergence rates for symmetric positive definite radial kernels. We conduct experiments and demonstrate that the fbKDE is superior to the KDE and a previously proposed weighted KDE in the uniform norm, and that it compares favorably in square norm.

4.1 Introduction

First recall the definition from Chapter 1. Given an iid sample $X_1, \ldots, X_n \in \mathbb{R}^d$ drawn according to a probability density f, the kernel density estimator is

$$f_{KDE} = \frac{1}{n} \sum_{i=1}^{n} k(\cdot, X_i),$$

where k is a kernel function with parameter σ.

A strength of the KDE is that it makes few assumptions about f and that it is consistent, meaning that it converges to f as $n \to \infty$ [18]. Analysis of the KDE stems from the following application of the triangle inequality in some norm of interest, where $*$ denotes convolution:

$$\|f_{KDE} - f\| \leq \|f_{KDE} - f * k\| + \|f * k - f\|.$$

Critical to the analysis of the KDE is the dependence of the *bandwidth* parameter σ on n. The first term tends to zero provided $n\sigma^d \to \infty$, i.e, the number of data points per unit volume tends to infinity. This is shown using properties of convolutions (since f_{KDE} and

$f * k$ may be viewed as convolutions of the kernel with the empirical and true distributions, respectively) and concentration of measure. For the latter term to tend to zero, $\sigma \to 0$ is necessary, so that the kernel behaves like a Dirac measure. With additional assumptions on the smoothness of f, the optimal growth of σ as a function of n can be determined.

The choice of bandwidth determines the performance of the KDE, and automatically selecting the optimal kernel bandwidth remains a difficult problem. Thus, researchers have developed numerous plug-in rules and cross-validation strategies, all of which are successful in some situations. A recent survey counts no fewer than 30 methods in the literature and cites 6 earlier review papers on the topic [15].

As an alternative to the standard KDE, some authors have investigated weighted KDEs, which have the form

$$f_\alpha = \sum_{i=1}^{n} \alpha_i k(\cdot, X_i).$$

The weight vector $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{R}^n$ is learned according to some criterion, and such weighted KDEs have been shown to yield improved performance over the standard KDE in certain situations [73, 74, 75, 76]. Consistency of such estimators has been investigated, but still under the assumption that $\sigma \to 0$ with n [77, 78].

In this work we consider the question of whether it is possible to learn the weights of a weighted KDE such that the resulting density estimator is consistent, for a broad class of f, where the bandwidth σ remains fixed as $n \to \infty$. This question is of theoretical interest, given that all prior work establishing consistency of a KDE, to our knowledge, requires that the bandwidth shrinks to zero. The question is also of practical interest, since such a density estimator could potentially be less sensitive to the choice of bandwidth than the standard KDE, which, as mentioned above, is the main factor limiting the successful application of KDEs in practice.

In Section 4.2 below, we introduce a weighted KDE that we refer to as the fixed-bandwidth KDE (fbKDE). Its connection to related work is given in Section 4.3. The theoretical properties of this estimator, including consistency and rates of convergence with a fixed bandwidth, are established in Section 4.4. Our analysis relies on the RKHS, a common machine learning structure seldom used to understand KDE's. In Section 4.5, a simulation study is conducted to compare the fbKDE against the standard KDE and another weighted KDE from the literature. Our results indicate that the fbKDE is a promising alternative to these other methods of density estimation.

4.2 Fixed bandwidth KDE

We start by assuming access to iid data $X_1, \ldots, X_n$ sampled from an unknown distribution with density f, having support $\mathcal{S} = \operatorname{supp}\{f\}$ contained in the known domain $\mathcal{X} \subset \mathbb{R}^d$, and with dominating measure μ. The set $\mathcal{X}$ is either compact, in which case μ is taken to be the Lebesgue measure, or $\mathcal{X} = \mathbb{R}^d$, in which case μ is a known finite measure. We study a weighted kernel density estimator of the form

$$f_\alpha = \sum_{i=1}^{n} \alpha_i k(\cdot, Z_i) \tag{4.1}$$

where $\alpha := (\alpha_1, \ldots, \alpha_n)^T \in A_n \subset \mathbb{R}^n$, $Z_i = X_i + \Gamma_i$, and Γ_i is sampled iid from a known distribution with density f_Γ. Here, f_Γ is chosen to ensure $Z_i \in \mathcal{X}$, but not necessarily in

$\mathcal{S}$. Note that f_α is defined on $\mathcal{X}$ and k on $\mathcal{X} \times \mathcal{X}$. Throughout, A_n is taken to be an ℓ_1 ball in $\mathbb{R}^n$, that is

$$A_n = \{\alpha \in \mathbb{R}^n \mid \|\alpha\|_1 \leq R_n\},$$

for $R_n \in \mathbb{R}$. The reason for centering the kernels at $Z_i = Z_i + \Gamma_i$ instead of X_i is that to accurately approximate f with a fixed bandwidth, we might need centers outside the support of f.

To measure the error between f_α to f we may consider the $\mathcal{L}^2_\mu(\mathcal{S})$ difference, where $\mathcal{L}^2_\mu(\mathcal{S})$ is the space of equivalence classes of square integrable functions, and where (using both f for the function and its equivalence class) $\|f\|^2_{\mathcal{L}^2_\mu(\mathcal{S})} := \int_{\mathcal{S}} f^2 d\mu$. However, we do not know the set $\mathcal{S}$ and cannot compute said difference. Hence, we consider the $\mathcal{L}^2_\mu(\mathcal{X})$ difference from f_α to f.

To determine the scaling coefficients α we consider minimizing the $\mathcal{L}^2_\mu(\mathcal{X})$ difference between f and f_α. We have the following minimization problem:

$$\min_{\alpha \in A_n} \|f - f_\alpha\|^2_{\mathcal{L}^2_\mu(\mathcal{X})}.$$

Since $\|f - f_\alpha\|^2_{\mathcal{L}^2_\mu(\mathcal{X})} = \int f_\alpha^2 - 2\int f f_\alpha + \int f^2$, and the term $\int f^2$ is independent of α, the above problem is equivalent to minimizing

$$\begin{aligned} J(\alpha) &:= \int_{\mathcal{X}} f_\alpha(x)^2 d\mu(x) - 2\int_{\mathcal{X}} f_\alpha(x) f(x) d\mu(x) \\ &= \int_{\mathcal{X}} f_\alpha(x)^2 d\mu(x) - 2\int_{\mathcal{S}} f_\alpha(x) f(x) d\mu(x) \\ &= \int_{\mathcal{X}} f_\alpha(x)^2 d\mu(x) - 2H(\alpha), \end{aligned}$$

where

$$\begin{aligned} H(\alpha) &:= \int_{\mathcal{S}} f_\alpha(x) f(x) d\mu(x) \\ &= \sum_{i=1}^n \alpha_i \int_{\mathcal{S}} k(x, Z_i) f(x) d\mu(x) \\ &=: \sum_{i=1}^n \alpha_i h_i. \end{aligned}$$

Since we don't know f, we don't know the true form of $H(\alpha)$ and $J(\alpha)$. However, the terms h_i are expectations with respect to f so we can estimate the term $H(\alpha)$ using the available data $\{X_i\}_{i=1}^n$. We use the leave-one-out estimator

$$H_n(\alpha) := \sum_{i=1}^n \alpha_i \frac{1}{n-1} \sum_{j \neq i} k(X_j, Z_i) =: \sum_{i=1}^n \alpha_i \widehat{h}_i.$$

With the aid of $H_n(\alpha)$, we define the function

$$J_n(\alpha) := \int_{\mathcal{X}} f_\alpha(x)^2 d\mu(x) - 2H_n(\alpha) \tag{4.2}$$

to which we have access. Let $J^* := \inf_{\alpha \in A_n} J(\alpha)$ and

$$\alpha^{(n)} := \arg\min_{\alpha \in A_n} J_n(\alpha). \tag{4.3}$$

The estimator

$$f_{\alpha^{(n)}} = \sum_{i=1}^{n} \alpha^{(n)}{}_i k(\cdot, Z_i) \tag{4.4}$$

is referred to as the fixed bandwidth kernel density estimator (fbKDE), and is our object of study. In the following sections. this estimator is shown to be consistent for a fixed kernel bandwidth σ under certain conditions on f, k, R_n, and f_Γ.

4.3 Related work

The use of the $\mathcal{L}^2$ norm as an objective function for kernel density estimation is not new, and in fact, choosing σ to minimize J_n, with $\alpha_i = 1/n$, is the so-called least-squares leave-one-out cross-validation technique for bandwidth selection. Minimizing J_n subject to the constraints $\alpha_i \geq 0$ and $\sum_i \alpha_i = 1$ was proposed by [73], and later rediscovered by [74], who also proposed an efficient procedure for solving the resulting quadratic program, and compared the estimator to the KDE experimentally. This same estimator was later analyzed by [77], who established an oracle inequality and consistency under the usual conditions for consistency of the KDE.

Weighted KDEs have also been developed as a means to enhance the standard KDE in various ways. For example, a weighted KDE was proposed in [75] as a form of multiple kernel learning, where, for every data point, multiple kernels of various bandwidths were assigned, and the weights optimized using J_n. A robust kernel density estimator was proposed in [76], by viewing the standard KDE as a mean in a function space, and estimating the mean robustly. To improve the computational efficiency of evaluating a KDE, several authors have investigated sparse KDEs, as we did in Chapter 2 learned by various criteria [79, 32, 45, 48, 29].

The one-class SVM has been shown to converge to a truncated version of f in the $\mathcal{L}^2$ norm [80]. If the truncation level (determined by the SVM regularization parameter) is high enough, and the density is bounded, then f is consistently estimated. An ensemble of kernel density estimators is studied by [78], who introduce aggregation procedures such that a weighted combination of standard KDEs of various bandwidths performs as well as the KDE whose bandwidth is chosen by an oracle.

In the above-cited work on weighted KDEs, whenever consistency is shown, it assumes a bandwidth tending to zero. Furthermore, the weights are constrained to be nonnegative. In contrast, we allow the weights on individual kernels to be negative, and this enables our theoretical analysis below. Finally, we remark that the terms "fixed" or "constant" bandwidth have been used previously in the literature to refer to a KDE where each data point is assigned the same bandwidth, as opposed to a "variable bandwidth" KDE where each data point might receive a different bandwidth, we instead use "fixed bandwidth" to mean the bandwidth remains fixed as the sample size grows.

4.4 Theoretical Properties

Notation The space $L^p_\nu(\mathcal{X})$ is the set of functions on $\mathcal{X}$ for which the p^{th} power of their absolute value is ν-integrable over $\mathcal{X}$. $\mathcal{L}^p_\nu(\mathcal{X})$ is the set of equivalence classes of $L^p_\nu(\mathcal{X})$, where two functions g_1 and g_2 are equivalent if $\int_\mathcal{X} |g_1 - g_2|^p \, d\nu = 0$. The symbol $\|\cdot\|_2$ will denote both the Euclidean norm in $\mathbb{R}^d$ as well as the $\mathcal{L}^2_\nu$ norm; which is used will be clear from the context, as the elements of $\mathbb{R}^d$ will be denoted by letters towards the end of the alphabet (x, y, and z). The set $C(\mathcal{X})$ denotes the space of continuous functions on $\mathcal{X}$. Finally, the support of any function g is denoted by $\operatorname{supp}\{g\}$. Also, recall we refer to symmetric positive definite radial kernels as SPD radial kernels.

If k is a positive definite radial kernel then $\sup_{x,x' \in \mathcal{X}} k(x,x) \leq C_k$ for some $C_k > 0$. This holds because

$$\begin{aligned} k(x,x') &= \langle k(\cdot,x), k(\cdot,x') \rangle \leq \|k(\cdot,x)\| \, \|k(\cdot,x')\| \\ &= \sqrt{k(x,x)}\sqrt{k(x',x')} = \sqrt{g(0)}\sqrt{g(0)} \\ &= g(0). \end{aligned}$$

We will make constant use of the following assumptions:

A0 The set $\mathcal{X}$ is either a compact subset of $\mathbb{R}^d$, in which case μ is taken to be the Lebesgue measure, or $\mathcal{X} = \mathbb{R}^d$, in which case μ is a known finite measure.

This assumption is always held throughout the paper. It will not be explicitly stated in the statements of results below, but will be remembered in the proofs when needed.

A1 $X_1, \ldots, X_n$ are sampled independently and identically distributed (iid) according to f. $\Gamma_1, \ldots, \Gamma_n$ are sampled iid according to f_Γ. Furthermore $X_1, \ldots, X_n, \Gamma_i, \ldots, \Gamma_n$ are independent.

Given $\{(X_i, \Gamma_i)\}_{i=1}^n$ as in A1, we define $Z_i := X_i + \Gamma_i$ for $1 \leq i \leq n$. This notation will be kept throughout the paper.

A2 The kernel k is PSD radial, with $k(x,x) = C_k$ for all $x \in \mathcal{X}$. Furthermore, k is Lipschitz, that is, there exists $L_k > 0$ such that the inequality $\|k(\cdot,x) - k(\cdot,y)\|_2 \leq L_k \|x - y\|_2$ holds for all x, $y \in \mathcal{X}$.

Recall from equation (4.3) that $\alpha^{(n)}$ is the minimizer of J_n over A_n. To show the consistency of f_α the overall approach of the following sections will be to show that $J(\alpha^{(n)})$ is close to $J^* = \inf_{\alpha \in A_n} \|f - f_\alpha\|_2^2$ with high probability, and then show that $J(\alpha)$ (and therefore J^*) can be made arbitrarily small for optimal choice of α. We start by stating an oracle inequality relating $J(\alpha^{(n)})$ and J^*.

Lemma 5. *Let $\epsilon > 0$. Let $\{(X_i, \Gamma_i)\}_{i=1}^n$ satisfy assumption A1. Let k satisfy $\sup_{x,x' \in \mathcal{X}} k(x,x') \leq C_k$ and f_α be as in Equation (4.1). Let $\delta = 2n \exp\left(-\frac{(n-1)\epsilon^2}{8C_k^2 R_n^2}\right)$. With probability $\geq 1 - \delta$,*

$$\|f - f_{\alpha^{(n)}}\|_2^2 \leq \epsilon + \inf_{\alpha \in A_n} \|f - f_\alpha\|_2^2 .$$

□

This result allows us to concentrate on the term $J^* \inf_{\alpha \in A_n} \|f - f_\alpha\|_2^2$, which we proceed to do in the following sections.

4.4.1 Consistency of $f_{\alpha^{(n)}}$

We state the consistency of $f_{\alpha^{(n)}}$ for positive definite radial kernels:

Theorem 3. *Let $\epsilon > 0$. Let $\{(X_i, \Gamma_i)\}_{i=1}^n$ satisfy A1, where $f \in \mathcal{F}(\mathcal{X}) := L^2_\mu(\mathcal{X}) \cap C(\mathcal{X})$ and $\operatorname{supp}\{f_Z\} \supseteq \mathcal{X}$. Let k satisfy A2 and $f_{\alpha^{(n)}}$ be as in Equation (4.4). If A_n is such that $R_n \to \infty$ but $R_n^2 \log n/n \to 0$ as $n \to \infty$, then*

$$P_{X,\Gamma}\left\{\|f - f_{\alpha^{(n)}}\|_2^2 > \epsilon\right\} \to 0$$

as $n \to \infty$. □

The sketch of the proof is as follows. To analyze the term $J^* = \inf_{\alpha \in A_n} \|f - f_\alpha\|_2^2$ from Lemma 5, we will use the fact that for positive definite radial kernels $\mathcal{H}$ is dense in $\mathcal{F}$ [17] in the sup-norm sense and that $\mathcal{H}^0 := \left\{\sum_j^N c_j k(\cdot, y_j) | y_j \in \mathcal{X}, c_i \in \mathbb{R}, \text{and } N \in \mathbb{N}\right\}$ is dense in $\mathcal{H}$ in the $\mathcal{H}-$norm sense [16]. That is, there exists a function $f_\mathcal{H} \in \mathcal{H}$ arbitrarily close to f and a function $f_\beta \in \mathcal{H}^0$ arbitrarily close to $f_\mathcal{H}$. The function f_β has the form

$$f_\beta := \sum_{j=1}^m \beta_j k(\cdot, y_j), \tag{4.5}$$

where $\beta = (\beta_1, \ldots, \beta_m)^T$. Note this is an abuse of notation since the functions f_α and f_β do not have the same centers nor necessarily the same number of components. By the triangle inequality we have for any α in A_n:

$$\|f - f_\alpha\|_2 \leq \|f - f_\mathcal{H}\|_2 + \|f_\mathcal{H} - f_\beta\|_2 + \|f_\beta - f_\alpha\|_2 . \tag{4.6}$$

By the above denseness results, the first two terms are small. To make the third term small we need two things: that A_n is large enough so that there is an $\alpha \in A_n$ matching β, and that the centers $\{Z_i\}_{i=1}^n$ of f_α are close to the centers $\{y_j\}_{j=1}^m$ of f_β, which will be true with certain probability. In Section 4.7 we will prove all these approximations and show that the relevant probability is indeed large and approaches one.

4.4.2 Convergence rates for $f_{\alpha^{(n)}}$

The convergence rates for radial SPD kernels may be slow, since the kernels are "universal" in that they can approximate arbitrary functions in $L^2_\mu(\mathcal{X}) \cap C(\mathcal{X})$. To get better rates we can make stronger assumptions on f. Thus, let $\mathcal{F}_k = \left\{f \mid f \geq 0 \text{ a.e.}, \int f(x)d\mu(x) = 1, \ f = \int_\mathcal{X} k(\cdot, x)\lambda(x)dx, \ \lambda \in L^1(\mathcal{X})\right\}$, that is, the space of densities expressible as k-smoothed L_1 functions. Then we obtain the following convergence rates.

Theorem 4. *Let $\delta \in (0, 1)$. Let $\mathcal{S} = \mathcal{X}$, k satisfy A2, $\{(X_i, \Gamma_i)\}_{i=1}^n$ satisfy A1 with $f \in \mathcal{F}_k$, and $\min_{z \in \mathcal{X}}\{f_Z(z)\} > 0$. Let $f_{\alpha^{(n)}}$ be as in Equation (4.4). If $d > 4$ and $R_n \sim n^{1/2-d/2}$, then with probability $\geq 1 - \delta$*

$$\|f - f_{\alpha^{(n)}}\|_2^2 \lesssim \left(\frac{1}{n}\right)^{2/d} \log^{1/2}(n/\delta) .$$

If $d \leq 4$ and $R_n \sim n^{(1-C)/2}$ for C an arbitrary constant in $(0,1)$, then with probability $\geq 1-\delta$

$$\|f - f_{\alpha^{(n)}}\|_2^2 \lesssim \left(\frac{1}{n}\right)^{C/2} \log^{2/d}(n/\delta).$$

□

The symbol $\lesssim$ indicates the first term is bounded by a positive constant (independent of n and d) times the second term, and $\sim$ means they grow at the same rate. Note the condition $\min_{z\in\mathcal{X}}\{f_Z(z)\} > 0$ is satisfied, for example, if $\mathcal{X}$ is compact and f_Γ is Gaussian. The first step in proving Theorem 4 is just a reformulation of the oracle inequality from Lemma 5:

Lemma 6. *Let $\delta_1 \in (0,1)$. Let $\{(X_i,\Gamma_i)\}_{i=1}^n$ satisfy assumption A1. Let k satisfy assumption A2 and f_α be as in Equation (4.1). Then with probability $\geq 1-\delta_1$*

$$J(\alpha^{(n)}) \leq \epsilon_1(n) + J^*$$

where $\epsilon_1(n) := \sqrt{8}C_k R_n \sqrt{\frac{\log(4n/\delta_1)}{n-1}}$. □

Now, for the J^* term in Lemma 6 we introduce the function f_β as in Equation (4.6) and make the following decomposition, valid for any $\alpha \in A_n$:

$$\|f - f_\alpha\|_2 \leq \|f - f_\beta\|_2 + \|f_\beta - f_\alpha\|_2.$$

The following lemma concerns the term $\|f - f_\beta\|_2$, and is taken from [81]:

Lemma 7. *Let $f \in \mathcal{F}_k$. For any $m \in \mathbb{N}$ there are m points $\{y_j\}_{j=1}^m \subset \mathcal{X}$ and m coefficients $\{c_j\}_{j=1}^m \subset \mathbb{R}$ such that*

$$\left\| f - \sum_{i=1}^m \frac{c_j \|\lambda\|_1}{m} k(\cdot, y_j) \right\|_\infty \leq \epsilon_2(m)$$

where $\epsilon_2(m) := C\|\lambda\|_1 \sqrt{\frac{V_k}{m}}$ for some absolute constant C and where V_k is the VC-dimension of the set $\{k(\cdot,x) \mid x \in \mathcal{X}\}$. □

The VC dimension of a set $\{g_i\}$ is the maximum number of points that can be separated arbitrarily by functions of the form $g_i - r$, $r \in \mathbb{R}$. For radial kernels, $V_k = d+1$ (see [82], [83]).

Now let $\beta_j = \frac{c_j\|\lambda\|_1}{m}$ and $f_\beta = \sum_{j=1}^m \beta_j k(\cdot, y_j)$. For the remaining term $\|f_\beta - f_\alpha\|_2$ we will proceed as in the proof of Theorem 3. That is, we need that for all y_j there is a point Z_{i_j} close to it, and then we can approximate f_β with $f_\alpha = \sum_{i=1}^n \alpha_i k(\cdot, Z_i)$.

Lemma 8. *Let $\delta_2 \in (0,1)$, let f, m, and f_β be as above. Let $\{(X_i,\Gamma_i)\}_{i=1}^n$ satisfy assumption A1. With probability $\geq 1-\delta_2$*

$$\inf_{\alpha\in A_n} \|f_\beta - f_\alpha\|_2 \leq \epsilon_3(n,m)$$

where $\epsilon_3(n,m) := \frac{C}{n^{1/d}} \log^{1/d}(m/\delta_2)$, for C a constant independent of n and m. □

Putting Lemmas 6, 7, and 8 together and choosing the correct $m(n)$ we obtain the proof of Theorem 4.

4.4.3 Convergence rates for the Box kernel

While the previous theorem considered SPD radial kernels, the oracle inequality applies more generally, and in this section we present rates for a nonradial kernel, the box kernel. In this section we assume $\mathcal{X} = [0,1]^d$ and that the kernel centers are predetermined according to a uniform grid. precise details are given in the proof of Theorem 5. Thus the fbKDE centered at $\{y_i\}_{i=1}^M$ is $\tilde{f}_\alpha := \sum_{i=1}^M \alpha_i k(\cdot, y_i)$, where the α weights are learned in the same way as before, the only change being the kernel centers. Let $\mathcal{F} = L_1(\mathcal{X}) \cap L_2(\mathcal{X}) \cap \mathrm{Lip}(\mathcal{X})$, where $\mathrm{Lip}(\mathcal{X})$ are the Lipschitz functions on $\mathcal{X}$, and let L_f be the Lipschitz constant of f. Also, let k be the box kernel $k(x,y) = \frac{1}{(2\sigma)^d}\mathbf{1}_{\{\|x-y\|_\infty \le \sigma\}}$ defined on $\mathcal{X} \pm \sigma \times \mathcal{X} \pm \sigma$, and for simplicity assume $\sigma = \frac{1}{2q}$ for q a positive integer. The following theorem is proved in the supplementary material.

Theorem 5. *Let $f \in \mathcal{F}$, $\mathrm{supp}\{f_Z\} \supset \mathcal{X}$, $\{(X_i, \Gamma_i)\}_{i=1}^n$ satisfy A1, and $R_n \sim n^{(d-1)/(2d+2)}$. Let $\delta \in (0,1)$. With probability at least $1-\delta$*

$$\left\| f - \tilde{f}_{\alpha^{(n)}} \right\|_2^2 \lesssim \frac{\log^{1/2}(n/\delta)}{n^{1/(d+1)}}.$$

□

As with the previous results, the stochastic error is controlled via an oracle inequality. Whereas the preceding results leveraged known approximation properties of SPD radial kernels, in the case of the box kernel we give a novel argument for approximating Lipschitz functions with box kernels having a non-vanishing bandwidth.

4.5 Experimental Results

We now explore the performance of fbKDE in a few different ways. First, we will observe the influence of the γ variables and the set $\mathcal{X}$ on the approximation behavior of fbKDE. Second, we will compare the performances of fbKDE to KDE and vKDE (defined below) when the bandwidth is chosen too large as the sample size increases. Finally, we will compare performance for many different datasets and for favorable choices of bandwidth.

A note on implementation

When implementing the fbKDE, there are a few considerations. First, when computing $\int_{\mathcal{X}} f_\alpha^2(x) d\mu(x)$, the first term of $J_n(\alpha)$, we must compute the integral $\int_{\mathcal{X}} k(x,z_i)k(x,z_j)d\mu(x)$. For computational considerations we assume $\mathcal{X} = \mathbb{R}^d$ and μ is the Lebesgue measure. This deviates from our theory, which requires finite μ for $\mathcal{X} = \mathbb{R}^d$. Thus, for the Gaussian kernel, which we use in our experiments, this leads to $\int_{\mathcal{X}} k(x,z_i)k(x,z_j)d\mu(x) = k_{\sqrt{2}\sigma}(z_i,z_j)$. To obtain the α weights we have to solve a quadratic program. We used the ADMM algorithm described in [84], with the aid of the ℓ_1 projection algorithm from [55].

4.5.1 The role of σ_γ and $\mathcal{X}$

We first study the behavior of fbKDE as the γ variables change and with respect to the domain of error. For this let us consider $n = 1000$ samples from a triangular density on $[0,1]$. Besides being compact, the density has a discontinuity at the origin. The γ

variables were drawn from a Gaussian distribution with parameter σ_γ. We look at two possible values: $\sigma_\gamma = .05$ and $\sigma_\gamma = .25$, to contrast between small and large σ_γ. We consider both the cases $\mathcal{X} = \mathbb{R}^d$ and $\mathcal{X}$ compact, in which case we set it to be the interval $[\min(x_i) - \sigma_\gamma, \max(x_i) + \sigma_\gamma]$. We have used values of $R = 5$ and kernel bandwidth $\sigma = .1$.

Figure 4.1 shows the resulting fbKDE for the above setup. The errors J_n for large and small σ_γ, in the case $\mathcal{X} = \mathbb{R}^d$, are, respectively, -1.2706 and -1.2702. The errors for compact $\mathcal{X}$ are -1.2711 for large σ_γ and -1.2757 for small σ_γ. When the error is taken over all of $\mathbb{R}^d$ the choice of σ_γ does not seem to affect much the behavior of fbKDE, even for very different values of this parameter. When the error is taken over the compact interval defined by σ_γ the fbKDE overshoots at the discontinuity. We also studied other densities with compact support and discontinuities, like the uniform density and truncated Gaussians, as well as other values of the kernel parameter σ, and the exhibited behavior is similar.

The first behavior to observe is that outside the support of the density the fbKDE does take negative values. By placing negative α weights in one side of the discontinuity and positive weights in the other, portions of the positively weighted kernels are subtracted off allowing for a sharp change, and a finer approximation. This behavior will be studied with more care, compared to the standard KDE, in Section 4.5.2. Now, when the error is taken over $\mathbb{R}^d$ the fbKDE, although it overshoots, stays close to zero since it is penalized by differing much from the density even outside its support. When the error is taken over a small compact set (small σ_γ), the fbKDE overshoots greatly outside said support. This is because it tries to minimize the error inside the compact set without regard for its consequences outside of it, hence it achieves the smallest error inside the compact set, to the expense of larger deviations outside of it. When σ_γ is large, the behavior is similar to that of the error taken over $\mathbb{R}^d$ because the set $\mathcal{X}$ is large and well outside the support of the density, so the approximation also considers minimizing the error over the zero portion of $\mathcal{X}$.

4.5.2 Inappropriate σ and sample size

We now study the behavior of fbKDE when the bandwidth σ is too large, and we compare to that of the standard KDE. We have chosen Gaussian γ's but the results are similar for the uniform case. We also compare to the behavior of the variable bandwidth Kernel Density Estimator, which we refer as vKDE, see [85]. The vKDE has the form

$$f_{vKDE} = \frac{1}{n} \sum_{i=1} n k_{\sigma_i}(x, X_i)$$

where each data point has an individual bandwidth σ_i. [85] proposes to use

$$\sigma_i = \sigma \left(\frac{\lambda}{f_{KDE}(X_i)} \right)^{1/2},$$

where σ is a global bandwidth parameter and λ is the geometric mean of $\{f_{KDE}(X_i)\}_{i=1}^n$.

We consider three cases in which the kernel parameter σ has been kept large and fixed at a value of .25. In Figure 4.2 we observe again the triangle density, as well as the different estimates for two largely different sample sizes $n = 40$ and $n = 2000$. The resulting J_n error terms for the fbKDE, KDE, and vKDE are, respectively, $-.87$, $-.99$, and $-.98$ for the small sample case, and -1.25, -1.1, and -1.12 for the large sample case.

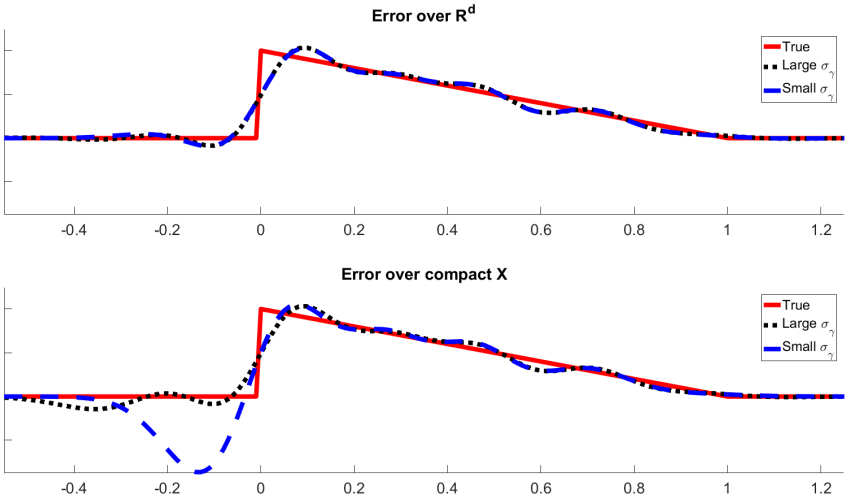

Figure 4.1: True density (solid) along with fbKDE for both a large $\sigma_\gamma = .25$ (dotted) and a small $\sigma_\gamma = .05$ (dashed). The top figure shows the case $\mathcal{X} = \mathbb{R}^d$. In this case the behavior is similar for both small and large parameter: fbKDE overshoots at the discontinuity but tries to keep a value close to zero. The bottom shows the case for compact $\mathcal{X}$, taken to be $[\min(x_i) - \sigma_\gamma, \max(x_i) + \sigma_\gamma]$. For large σ_γ the behavior is similar to the top case, since $\mathcal{X}$ contains a substantial region where the density is zero and for which fbKDE has to approximate. For small σ_γ, the approximation outside the support is not important and it highly overshoots but achieves the best error over $\mathcal{X}$.

Figure 4.3 shows a mixture of Gaussian with equal mixture weight centered at -1 and 1 with shared standard deviation of $2/9$. In the small sample case the resulting errors $-.171$, $-.275$, and $-.28$. In the large sample case the respective errors $-.31$, $-.278$, and $-.287$. Finally, Figure 4.4 shows a similar case for a mixture with standard deviations .25 and .1 and equal weights. The resulting errors are $-.12$, $-.41$ and $-.32$ for the small sample case, and $-.92$, $-.74$, and $-.79$ for the large sample case. In all cases and to better understand the fbKDE we have plotted the α values (stem plots) and their respective weighted components (light dashed curves). We have set $R = 5$ and $\sigma_\gamma = .001$.

Notice first, in the triangle case of Figure 4.2, that the fbKDE is able to skew more towards the concentrated region of the triangle in spite of the sudden drop and the large kernel bandwidth. This is because the allowance of negative mass outside but close to the original support is able to erase some of the excess mass obtained by putting positive weight close to the edge.This behavior is capable of approximating fine properties of densities. Figure 4.3 illustrates a similar principle for continuous densities. In this case the low density region between the two modes is hard to approximate with the large inappropriate bandwidth, and in fact both the KDE and the vKDE assign excess mass to this region even for large sample size. Furthermore the sample size does not seem to aid their approximation when the bandwidth is kept fixed and large, as is expected from their analytic properties. The fbKDE, however, does gain advantage as the sample size grows, and even for such a large kernel bandwidth it is able to match the true density. It does so by assigning slightly negative values to the α weights close to the center. These negative values, as in the discontinuity case for the triangular density, are able to subtract excess mass and refine the approximation. Thanks to the larger sample size it is able to control better the trade-off between positive and negative weights. Finally Figure 4.4

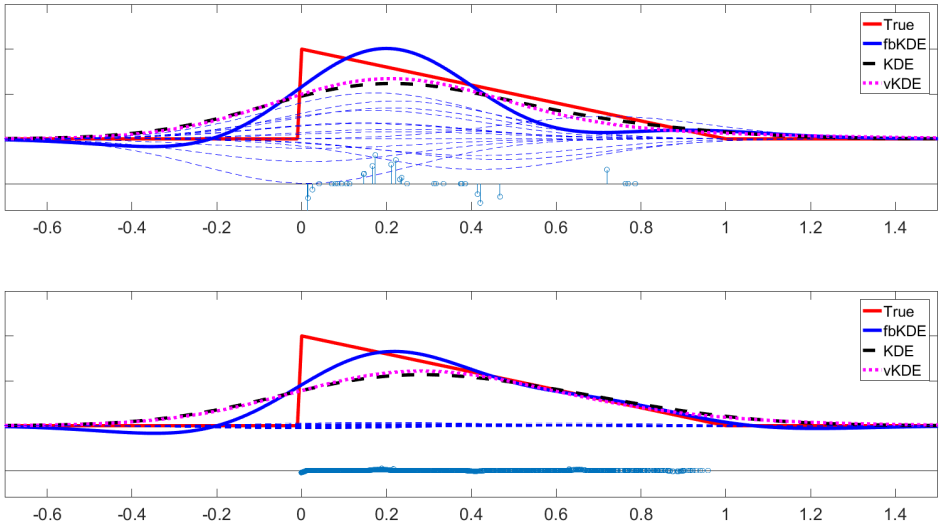

Figure 4.2: True density along with fbKDE, KDE and vKDE. All estimates use a kernel with large bandwidth $\sigma = .25$. The top figure shows the estimates for small sample size $n = 40$. Even in this low sample regime the fbKDE is able to skew more towards the origin thanks to the negative mass outside the original support. As the sample size grows the approximation is refined for the fbKDE while it remains about the same for the other estimates.

shows a similar case in which besides a mass valley there are two regions with different approximation scales. The vKDE was indeed originally proposed for the purpose of approximating densities with this behavior. Similar to the previous case, the fbKDE improves better as the sample size grows, while the KDE and vKDE seem to stagnate.

4.5.3 Performance for favorable σ

We examine a few benchmark real datasets as well as synthetic data from 1-dimensional densities. The synthetic densities are the triangular density as well as three Gaussian mixtures, a bimodal, a trimodal and a kurtotic, as shown in Figure 4.5. We computed the parameters σ, R_n, and σ_γ in two different ways. First we used rules of thumb. For σ, we used Silverman's rule of thumb ([11]). For R_n we used, based on the convergence rates, $(n/\log(n))^{1/3}$ for $d \leq 4$ and $(n/\log(n))^{(1/2-2/d)}$ for $d > 4$. For σ_γ we used, inspired by the Jaakkola heuristic [71], the median distance to the 5^{th} nearest neighbor. Second we used a V-fold CV procedure over 100 parameter combinations drawn randomly from Θ_d, with $V = 2$ for $n > 1000$ and $V = 3$ otherwise (see [86]). $\Theta_d := [.1, .5]_\ell \times [1.1, 2\sqrt{(n)}]_\ell \times [.001, .1]_\ell$ for $d \leq 4$, and $\Theta_d := [.1, 1]_\ell \times [1.1, 2n^{(1/2-2/d)}]_\ell \times [.001, .1]_\ell$ for $d > 4$, where the subscript ℓ indicates logarithmic spacing and where the range is chosen thus since the data is standardized and, for the R_n range, informed by the convergence rates. Finally, we used n to be 4/5 of the original data size for training and the rest for testing. We compute the value J_n^T as $\int {f_{\alpha^{(n)}}}^2 - 2\sum_{i=1}^n \alpha^{(n)}{}_i \frac{1}{n_T} \sum_{\ell=1}^{n_T} k(x_\ell^{(T)}, z_i)$, where $\left\{x_\ell^{(T)}\right\}_{\ell=1}^{n_T}$ is the test set. Figure 4.6 shows the bimodal density, the fbKDE, KDE, and vKDE along with the α values for the fbKDE. Table 4.1 shows the $J_n(\alpha^{(n)})^T$ error as well as the $\|\cdot\|_\infty$ error, where for some function g, $\|g\|_\infty := \max_{x \in \mathcal{X}} |g(x)|$.

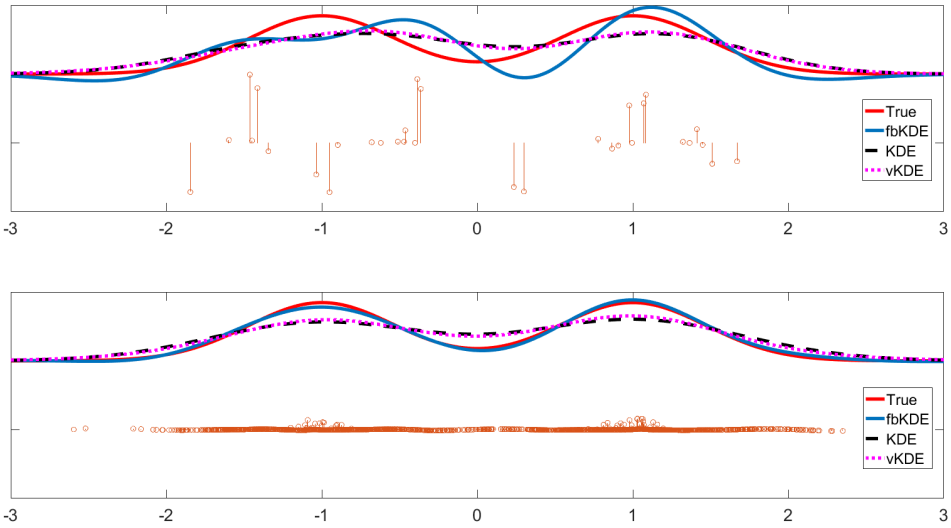

Figure 4.3: True density along with fbKDE, KDE and vKDE. All estimates use a kernel with large bandwidth $\sigma = .25$. The valey between the modes poses a challenge for KDE and vKDE for such a large kernel bandwidth, and a larger sample size does not aid their approximation. The fbKDE, to the contrary, can control better the balance between negative and positive mass and produces a very accurate approximation for the larger sample size. Note that in this case the fbKDE is everywhere nonnegative, a desired quality.

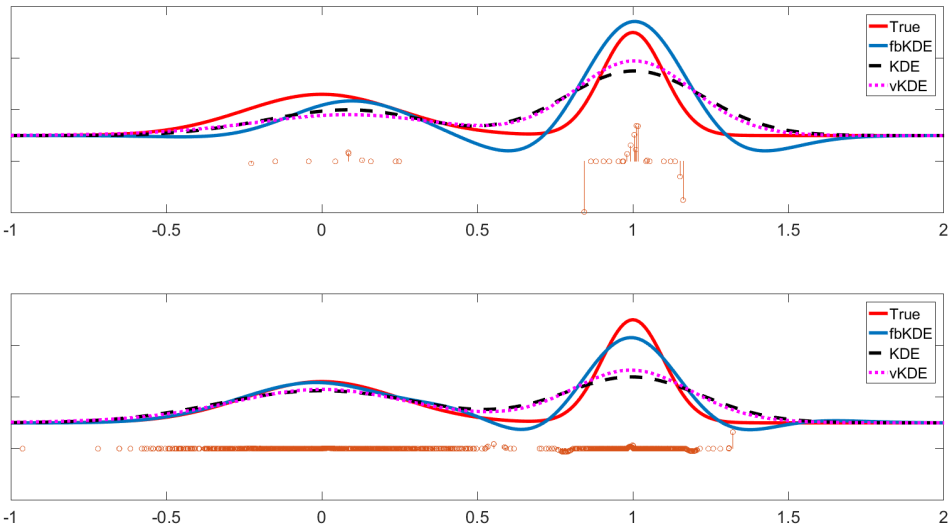

Figure 4.4: True density along with fbKDE, KDE and vKDE. All estimates use a kernel with large bandwidth $\sigma = .25$. In this case the estimates have to approximate the function both under the challenge of a valley region and the challenge of different approximation scales. The KDE and vKDE turn out too smooth, assigning extra mass to the valley and less mass to the thinner component. The fbKDE, again by properly using negative weights, can both approximate the valley and change approximation scales in spite of the large kernel bandwidth.

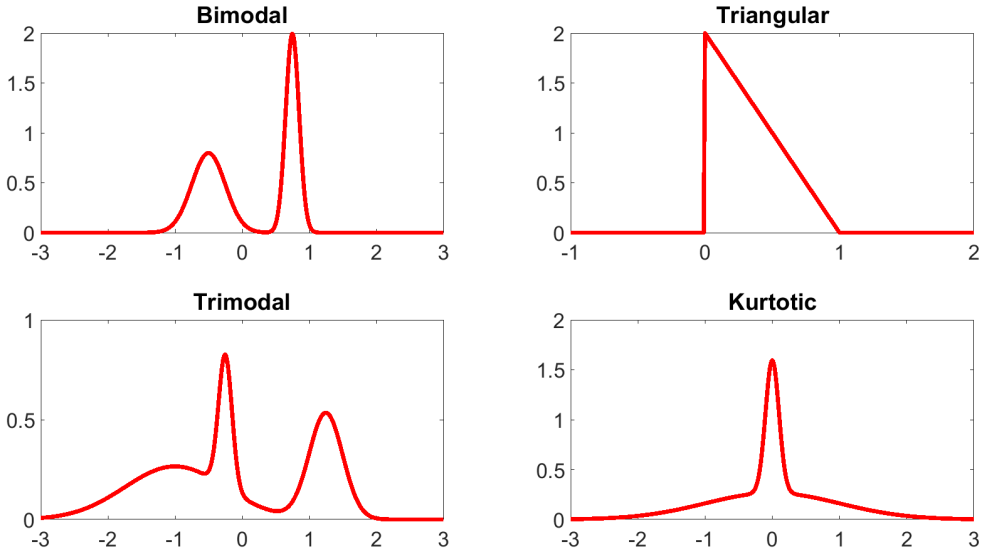

Figure 4.5: Bimodal, triangular, trimodal and kurtotic densities used to evaluate the fbKDE performance.

In Figure 4.6 the density has two Gaussian elements with different widths. It is difficult for KDE and even for the vKDE to approximate such a density. The fbKDE, however, is able to approximate components of different smoothness by making some weights negative. These weights subtract excess mass in regions of refinement. Note that by doing so the fbKDE may itself overshoot and become negative. A similar behavior is exhibited for other densities, in which smoothness varies across regions. In Table 4.1 we report the performance of the three estimators for both CV and rule of thumb. Note the fbKDE often performs better in terms of the $\|\cdot\|_\infty$, and when the bandwidth is chosen according to a rule of thumb. The fbKDE outperforms both the KDE and vKDE in about half of the cases.

Finally we show, for the bimodal density, a comparison of the performance as the sample size grows. We have chosen the parameters according to the rules of thumb discussed above. Table 4.2 presents the errors. Note that as the sample size grows the KDE and vKDE do not significantly improve, even though the bandwidth is being updated according to Silverman's rule. The fbKDE leverages new observations and refines its approximation, and this effect is more obvious for the $\|\cdot\|_\infty$ case. Indeed, the $\|\cdot\|_\infty$ error for fbKDE is smaller at $n = 50$ than at for KDE and vKDE at $n = 2050$. Similar results hold for the other synthetic datasets. This highlights a notable property of the fbKDE, that it can handle densities with differing degrees of smoothness.

4.6 Conclusion

We have presented a new member of the family of kernel estimators, the *fixed bandwidth kernel density estimator*, with desirable statistical properties. In particular, we showed the fbKDE is consistent for fixed kernel parameter σ, and we provided convergence rates. The fbKDE is a good alternative to the KDE in cases where computing an optimal bandwidth is difficult and for densities that are hard to approximate with inappropriate bandwidths. In these cases and as is shown in the experimental section, the fbKDE can greatly improve

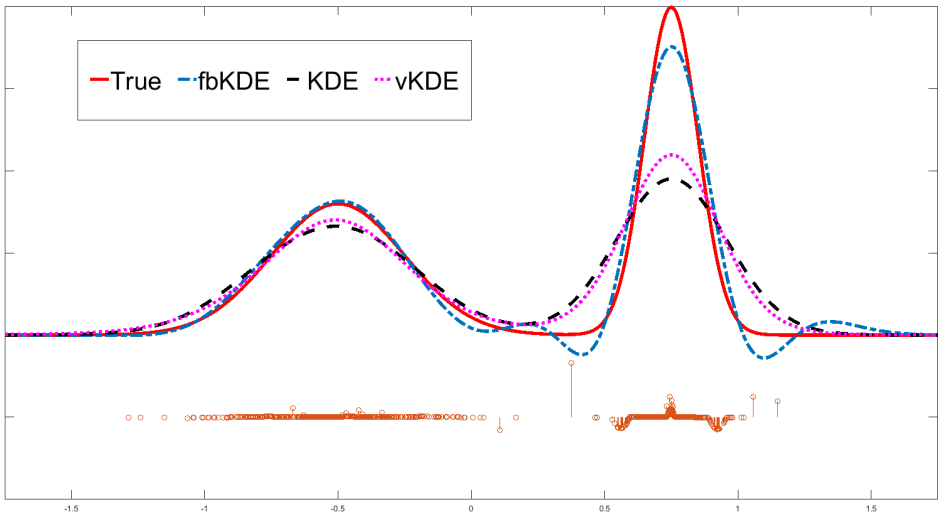

Figure 4.6: Bimodal density and kernel estimators with training size 800. The stem subplot indicates the values of α (centered offset for visualization). Note that some of the α weights are negative.

on the KDE. The way in which fbKDE achieves a more refined approximation is by balancing properly placed positive and negative weights, sometimes outside of the original support, which is facilitated by the Γ variables, and which is not possible with the standard KDE. A few problems of interest remain open. We have illustrated two possible rate of convergence results, but expect many other such results are possible, depending on the choice of kernel and smoothness assumptions. It also remains an open problem to extend our results to more general domains $\mathcal{X}$ and to dependent data.

4.7 Proofs

Oracle Inequality

First recall the oracle inequality lemma:

Lemma 5 Let $\epsilon > 0$. Let $\{(X_i, \Gamma_i)\}_{i=1}^n$ satisfy assumption A1. Let k satisfy assumption A2 and f_α be as in Equation (4.1). Let $\delta = 2n \exp\left(-\frac{(n-1)\epsilon^2}{8C_k^2 R_n^2}\right)$. With probability $\geq 1 - \delta$,

$$\|f - f_{\alpha^{(n)}}\|_2^2 \leq \epsilon + \inf_{\alpha \in A_n} \|f - f_\alpha\|_2^2 .$$

Proof of Lemma 5. For clarity in this section we will use capital letter to denote random variables. Recall the following definitions from Section 4.2. For a random variable $\Gamma_i \sim f_\Gamma$ and for $\{X_i\}_{i=1}^n \sim f^n$ we define

$$h_i = h_i(X_i, \Gamma_i) := \int_{\mathcal{S}} k(x, X_i + \Gamma_i) f(x) d\mu(x),$$

Table 4.1: Performance comparison for different datasets and bandwidth selection methods. For the synthetic datasets we drew 1000 samples, $n = 800$ of which were used for training.

		Rule of Thumb			Cross-Validation		
		fbKDE	KDE	vKDE	fbKDE	KDE	vKDE
Bimodal	J_n^T	**-1.0180**	-0.8110	-0.8765	**-1.0660**	-0.9785	-1.0413
	$\|\cdot\|_\infty$	**0.2287**	1.0141	0.8468	**0.1865**	0.5534	0.2782
Triangular	J_n^T	-1.2889	-1.2897	**-1.2958**	**-1.2332**	-1.2095	-1.2073
	$\|\cdot\|_\infty$	**1.0121**	1.0200	1.0923	1.1599	**1.1437**	1.2242
Trimodal	J_n^T	**-0.3317**	-0.2919	-0.3025	-0.3457	**-0.3379**	-0.3456
	$\|\cdot\|_\infty$	**0.2335**	0.4571	0.4156	0.1212	0.1879	**0.1032**
Kurtotic	J_n^T	**-0.5444**	-0.4735	-0.5271	-0.5831	-0.5647	**-0.5894**
	$\|\cdot\|_\infty$	**0.2800**	0.8122	0.5540	0.1379	0.3902	**0.1181**
Banana	J_n^T	-0.0838	-0.0821	**-0.0839**	-0.0821	-0.0837	**-0.0853**
Ringnorm	J_n^T	2.4E-09	-2.3E-10	**-2.7E-10**	-1.7E-10	-3.2E-10	**-3.5E-10**
Thyroid	J_n^T	-0.0932	-0.0448	**-0.1415**	**-0.2765**	-0.2514	-0.2083
Diabetes	J_n^T	-1.4E-05	-0.0004	**-9.8E-04**	**-0.0010**	-0.0007	**-0.0010**
Waveform	J_n^T	1.5E-09	**-1.2E-11**	1.25E-11	-2.1E-12	-1.2E-11	**-1.25E-11**
Iris	J_n^T	0.0166	**-0.0204**	0.0058	**-0.0102**	0.0027	0.0777

Table 4.2: Performance comparison with respect to the sample size for the bimodal density.

		Sample size n						
		50	250	450	1050	1650	1850	2050
$\|\cdot\|_\infty$ error	fbKDE	**0.7046**	**0.5847**	**0.4836**	**0.3549**	**0.1807**	**0.1642**	**0.1761**
	KDE	1.1341	1.0567	1.1451	1.0833	0.9684	0.9420	0.9160
	vKDE	1.0287	0.8811	1.0459	0.9670	0.8106	0.7562	0.7300
J_n^T error	fbKDE	**-0.8985**	**-0.9623**	**-0.7487**	**-1.0788**	**-0.9787**	**-1.0493**	**-0.9722**
	KDE	-0.7099	-0.7639	-0.6859	-0.8220	-0.8284	-0.8728	-0.8277
	vKDE	-0.7763	-0.8284	-0.7091	-0.8793	-0.8782	-0.9372	-0.8839

and

$$\widehat{h}_i = \widehat{h}_i(X, \Gamma_i) := \frac{1}{n-1} \sum_{j \neq i} k(X_j, X_i + \Gamma_i),$$

were we have used the simplified notation (X, Γ_i) to represent $(X_1, \ldots, X_n, \Gamma_i)$. To simplify notation further, we let $X_{/i}$ represent $(X_1, \ldots, X_{i-1}, X_{i+1}, \ldots, X_n)$ and use $\mathrm{P}_{X,\Gamma}\{\cdot\}$ for $\mathrm{P}_{X_1,\ldots,X_n,\Gamma_1,\ldots,\Gamma_n}\{\cdot\}$, and the same goes for $\mathbb{E}_{X,\Gamma}(\cdot)$. We now look at the probability that

$H_n(\alpha)$ is close to $H(\alpha)$. We have

$$\begin{aligned}
\mathrm{P}_{X,\Gamma}\left\{\sup_{\alpha\in A_n}|H_n(\alpha)-H(\alpha)|>\epsilon\right\} &= \mathrm{P}_{X,\Gamma}\left\{\sup_{\alpha\in A_n}\left|\sum_{i=1}^{n}\alpha_i\widehat{h}_i(X,\Gamma_i)-\sum_{i=1}^{n}\alpha_i h_i(X_i,\Gamma_i)\right|>\epsilon\right\}\\
&= \mathrm{P}_{X,\Gamma}\left\{\sup_{\alpha\in A_n}\left|\sum_{i=1}^{n}\alpha_i(\widehat{h}_i(X,\Gamma_i)-h_i(X_i,\Gamma_i))\right|>\epsilon\right\}\\
&\leq \mathrm{P}_{X,\Gamma}\left\{\sup_{\alpha\in A_n}\sum_{i=1}^{n}|\alpha_i|\left|\widehat{h}_i(X,\Gamma_i)-h_i(X_i,\Gamma_i)\right|>\epsilon\right\}\\
&\leq \mathrm{P}_{X,\Gamma}\left\{R_n\max_{1\leq i\leq n}\left|\widehat{h}_i(X,\Gamma_i)-h_i(X_i,\Gamma_i)\right|>\epsilon\right\}\\
&\leq \sum_{i=1}^{n}\mathrm{P}_{X,\Gamma}\left\{\left|\widehat{h}_i(X,\Gamma_i)-h_i(X_i,\Gamma_i)\right|>\frac{\epsilon}{R_n}\right\}.
\end{aligned}$$

Now let $\mathcal{A}_i := \left\{(x_1,\ldots,x_n,\gamma_i)\in\mathcal{S}^n\times\operatorname{supp}\{f_\Gamma\}\mid\left|\widehat{h}_i(x,\gamma_i)-h_i(x,\gamma_i)\right|>\frac{\epsilon}{R_n}\right\}$ and note that $\mathrm{P}_{X,\Gamma}\{\mathcal{A}_i\}=\mathrm{P}_{X,\Gamma_i}\{\mathcal{A}_i\}$. We have

$$\begin{aligned}
\mathrm{P}_{X,\Gamma_i}\{(X_1,\ldots,X_n,\Gamma_i)\in\mathcal{A}_i\} &= \int_{\operatorname{supp}\{f_\Gamma\}}\mathrm{P}_{X|\Gamma_i}\{(X_1,\ldots,X_n,\gamma_i)\in\mathcal{A}_i\mid\Gamma_i=\gamma_i\}\,f_\Gamma(\gamma_i)d\gamma_i\\
&= \int_{\operatorname{supp}\{f_\Gamma\}}\mathrm{P}_X\{(X_1,\ldots,X_n,\gamma_i)\in\mathcal{A}_i\}\,f_\Gamma(\gamma_i)d\gamma_i,
\end{aligned}$$

by independence. Furthermore,

$$\begin{aligned}
\mathrm{P}_X\{(X_1,\ldots,X_n,\gamma_i)\in\mathcal{A}_i\} &= \int_{\mathcal{S}}\mathrm{P}_{X_{/i}|X_i}\{(X_1,\ldots,x_i,\ldots,X_n,\gamma_i)\in\mathcal{A}_i\mid X_i=x_i\}\,f(x_i)d\mu(x_i)\\
&= \int_{\mathcal{S}}\mathrm{P}_{X_{/i}}\{(X_1,\ldots,x_i,\ldots,X_n,\gamma_i)\in\mathcal{A}_i\}\,f(x_i)d\mu(x_i).
\end{aligned}$$

We now bound the term inside the integral, abbreviated as $\mathrm{P}_{X_{/i}}\left\{(X_{/i},z_i)\in\mathcal{A}_i\right\}$. First, note that $h_i(x_i,\gamma_i)=\mathbb{E}_{X_j}\left(k(X_j,z_i)\right)$ for any $j\neq i$, therefore

$$\begin{aligned}
\mathbb{E}_{X_{/i}|X_i,\Gamma_i}\left(\widehat{h}_i(X,\Gamma_i)\mid X_i=x_i,\Gamma_i=\gamma_i\right) &= \mathbb{E}_{X_{/i}}\left(\widehat{h}_i(X_{/i},x_i,\gamma_i)\right)\\
&= h_i(x_i,\gamma_i).
\end{aligned}$$

Now, by assumption k is a positive definite radial kernel so there is a C_k such that $k(x,x')\leq C_k$ for all x,x', hence:

$$
\begin{aligned}
\mathrm{P}_{X_{/i}}\left\{(X_{/i}, z_i) \in \mathcal{A}_i\right\} &= \mathrm{P}_{X_{/i}}\left\{\left|\frac{1}{n-1}\sum_{\substack{j=1\\ j\neq i}}^{n} k(X_j, z_i) - \mathbb{E}_{X_{/i}}\left(\frac{1}{n-1}\sum_{\substack{j=1\\ j\neq i}}^{n} k(X_j, z_i)\right)\right| > \epsilon\right\} \\
&= \mathrm{P}_{X_{/i}}\left\{\left|\frac{1}{n-1}\sum_{\substack{j=1\\ j\neq i}}^{n} k(X_j, z_i) - \frac{1}{n-1}\mathbb{E}_{X_{/i}}\left(\sum_{\substack{j=1\\ j\neq i}}^{n} k(X_j, z_i)\right)\right| > \epsilon\right\} \\
&= \mathrm{P}_{X_{/i}}\left\{\left|\sum_{\substack{j=1\\ j\neq i}}^{n} k(X_j, z_i) - \mathbb{E}_{X_{/i}}\left(\sum_{\substack{j=1\\ j\neq i}}^{n} k(X_j, z_i)\right)\right| > (n-1)\epsilon\right\} \\
&\leq 2\exp\left\{-\frac{2(n-1)\epsilon^2}{C_k^2}\right\},
\end{aligned}
$$

where we have used Hoeffding's inequality. So we obtain

$$
\begin{aligned}
\mathrm{P}_{X,\Gamma}\left\{\left|\widehat{h}_i - h_i\right| > \frac{\epsilon}{R_n}\right\} &= \int_{\mathrm{supp}\{f_\Gamma\}}\int_{\mathcal{S}} \mathrm{P}_{X_{/i}}\left\{(X_{/i}, z_i) \in \mathcal{A}_i\right\} f(x_i) f_\Gamma(\gamma_i) d\mu(x_i) d\gamma_i \\
&\leq 2\exp\left\{-\frac{2(n-1)\epsilon^2}{C_k^2 R_n^2}\right\} \cdot \int_{\mathrm{supp}\{f_\Gamma\}} f_\Gamma(\gamma_i) d\gamma_i \cdot \int_{\mathcal{S}} f(x_i) d\mu(x_i) \\
&= 2\exp\left\{-\frac{2(n-1)\epsilon^2}{C_k^2 R_n^2}\right\},
\end{aligned}
$$

and

$$
\mathrm{P}_{X,\Gamma}\left\{\sup_{\alpha\in A_n} |H_n(\alpha) - H(\alpha)| > \epsilon\right\} \leq 2n\exp\left\{-\frac{2(n-1)\epsilon^2}{C_k^2 R_n^2}\right\}.
$$

Therefore, letting $\delta = 2n\exp\left\{-\frac{2(n-1)\epsilon^2}{C_k^2 R_n^2}\right\}$, for any $\alpha \in A_n$

$$
\begin{aligned}
\mathrm{P}_{X,\Gamma}\left\{|J_n(\alpha) - J(\alpha)| \leq 2\epsilon\right\} &= \mathrm{P}_{X,\Gamma}\left\{|H_n(\alpha) - H(\alpha)| \leq \epsilon\right\} \\
&= 1 - \mathrm{P}_{X,\Gamma}\left\{|H_n(\alpha) - H(\alpha)| > \epsilon\right\} \\
&\geq 1 - \delta.
\end{aligned}
$$

So, with probability $\geq 1-\delta$, $J_n(\alpha) \leq J(\alpha) + 2\epsilon$, and with probability $\geq 1-\delta$, $J(\alpha) \leq J_n(\alpha) + 2\epsilon$. Recall that $J_n(\alpha^{(n)}) \leq J_n(\alpha)$ for all $\alpha \in A_n$. Then with probability $\geq 1 - 2\delta$,

$$
J(\alpha^{(n)}) \leq \inf_{\alpha\in A_n} J(\alpha) + 4\epsilon.
$$

If we substitute $\epsilon' = 4\epsilon$ we obtain the desired result. □

Consistency of $f_{\alpha^{(n)}}$

In the following we will make use of the fact that, for continuous positive definite radial kernels, the RKHS norm dominates the sup-norm which in turn dominates the $\mathcal{L}^2_\mu(\mathcal{X})$ norm. Let's state this as a lemma.

Lemma 9. *Let k be a kernel satisfying Assumption A2 with RKHS $\mathcal{H}$, then for any $h \in \mathcal{H}$ we have*

$$\|h\|_2 \leq \|h\|_\infty \leq \|h\|_{\mathcal{H}}.$$

Proof. By assumption k is bounded and continuous so by Lemma 4.28 of [16] so is every element of $\mathcal{H}$. Hence, for $h \in \mathcal{H}$ and for $\mathcal{X}$ either compact or all of $\mathbb{R}^d$ the essential supremum equals the supremum, so we obtain

$$\begin{aligned}
\|h\|_2 &= \left(\int_{\mathcal{X}} |h(x)|^2 \, d\mu(x) \right)^{1/2} \\
&\leq \mu^{1/2}(\mathcal{X}) \sup_{x \in \mathcal{X}} \{|h(x)|\} \\
&= \mu^{1/2}(\mathcal{X}) \, \|h\|_\infty \\
&\leq \mu^{1/2}(\mathcal{X}) \sup_{x \in \mathcal{X}} \{|\langle h, k(\cdot, x)\rangle_{\mathcal{H}}|\} \\
&\leq \mu^{1/2}(\mathcal{X}) C_k^{1/2} \, \|h\|_{\mathcal{H}}
\end{aligned}$$

where the penultimate inequality follows from the reproducing property and the last inequality is just Cauchy-Swartz. □

Now to prove Theorem 3 we need a couple intermediate lemmas.

Lemma 10. *Let k satisfy assumption A2 and $f \in L^2_\mu(\mathcal{X}) \cap C(\mathcal{X})$, then*

$$\|f_\alpha - f\|_2 \leq \|f_\alpha - f_\beta\|_2 + 2\mu(\mathcal{X})^{1/2}\epsilon.$$

Proof. If $\mathcal{X}$ is compact, then $\mathcal{H}$ is dense in $C(\mathcal{X})$ (see [17]). Therefore, for fixed ϵ, there is an $f_{\mathcal{H}} \in \mathcal{H}$ such that

$$\|f_{\mathcal{H}} - f\|_\infty \leq \epsilon,$$

and by lemma 9

$$\|f_{\mathcal{H}} - f\|_2 \leq \mu(\mathcal{X})^{1/2}\epsilon.$$

If $\mathcal{X} = \mathbb{R}^d$, [17] tells us $\mathcal{H}$ is dense in $\mathcal{L}^2_\mu(\mathcal{X})$, so it directly follows that there is an $f_{\mathcal{H}}$ satisfying, for any $\epsilon > 0$,

$$\|f_{\mathcal{H}} - f\|_2 \leq \mu(\mathcal{X})^{1/2}\epsilon.$$

Similarly, since $\mathcal{H}^0$ is dense in $\mathcal{H}$ [16], for any fixed ϵ there is an $f_\beta \in \mathcal{H}^0$ such that

$$\|f_\beta - f_{\mathcal{H}}\|_{\mathcal{H}} \leq C^{-1/2}\epsilon,$$

hence, by lemma 9

$$\|f_\beta - f_{\mathcal{H}}\|_2 \leq \mu(\mathcal{X})^{1/2}\epsilon.$$

Therefore:

$$\begin{aligned}
\|f_\alpha - f\|_2 &\leq \|f_\alpha - f_\beta\|_2 + \|f_\beta - f_{\mathcal{H}}\|_2 + \|f_{\mathcal{H}} - f\|_2 \\
&\leq \|f_\alpha - f_\beta\|_2 + 2\mu(\mathcal{X})^{1/2}\epsilon.
\end{aligned}$$

□

Note that $f_\beta \in \mathcal{H}^0$ implies $f_\beta = \sum_{j=1}^m \beta_j k(\cdot, y_j)$ for some $m \in \mathbb{N}$ and where $(\beta_j, y_j) \in \mathbb{R} \times \mathcal{X}$ for all $1 \leq j \leq m$. To make the first term small, we first quantify the continuity of the kernel k. Let $\epsilon' = \epsilon / \|\beta\|_1$ and define

$$\eta_\epsilon := \mu^{1/2}(\mathcal{X}) \frac{\epsilon'}{L(k)},$$

where $L(k)$ is the Lipschitz constant of k. Then for every x and y in $\mathcal{X}$ we have that $\|x - y\|_2 \leq \eta_\epsilon$ implies $\|k(\cdot, x) - k(\cdot, y)\|_2 \leq \mu^{1/2}(\mathcal{X})\epsilon'$.

Recall $f_\alpha = \sum_{i=1}^n \alpha_i k(\cdot, Z_i)$, with the above result in hand we now have to make sure that at least a subset of the centers $\{Z_i\}_{i=1}^n$ of f_α are close to the centers $\{y_j\}_{j=1}^m$ of f_β with high probability. First, define $B_j = \left\{x \in \mathcal{X} \mid \|x - y_j\|_2 \leq \eta_\epsilon, \right\}$ and define $P_Z := P_{X+\Gamma}$. Then we obtain the following lemma:

Lemma 11. *Let $\epsilon > 0$ and f_β, f_α and B_j as above. Then*

$$P_Z \left\{ \forall B_j \; \exists Z_{i_j} \in \{Z_i\}_{i=1}^n \;\ni Z_{i_j} \in B_j \right\} \to 1$$

as $n \to \infty$.

Proof of Lemma 11. Let the event $A_j^C = \{\forall z \in \{Z_i\}_{i=1}^n, \;\; z \notin B_j\}$. Then

$$\begin{aligned}
\mathrm{P}_Z \left\{ A_j^C \right\} &= \mathrm{P}_Z \left\{ Z_1 \notin B_j, \ldots, Z_n \notin B_j \right\} \\
&= \prod_{i=1}^n \mathrm{P}_Z \left\{ Z_i \notin B_j \right\} \\
&= \prod_{i=1}^n (1 - \mathrm{P}_Z \left\{ Z_i \in B_j \right\}) \\
&= \prod_{i=1}^n (1 - p_j) = (1 - p_j)^n,
\end{aligned}$$

where $p_j = \int_{B_j} f_Z(z) dz$. Let $p_0 := \min_j \{p_j\}$ and recall that for $p < 1$ we have $1 - p \leq e^{-p}$. Hence $(1 - p_j)^n \leq e^{-np_j} \leq e^{-np_0}$, and

$$\begin{aligned}
\mathrm{P}_Z \left\{ \cap^m A_j \right\} &= 1 - \mathrm{P}_Z \left\{ \cup A_j^C \right\} \\
&\geq 1 - \sum_{j=1}^m \mathrm{P}_Z \left\{ A_j^C \right\} \\
&= 1 - \sum_{j=1}^m (1 - p_j)^n \\
&\geq 1 - \sum_{j=1}^m e^{-np_0} \\
&= 1 - m e^{-np_0}.
\end{aligned}$$

For this term to approach zero we need $p_0 := \min_j \int_{B_j} f_Z(z) dz$ to be strictly positive. This follows from the assumption that $\mathrm{supp}\,(f_Z) \supseteq \mathcal{X}$. Since m and p_0 only depend on ϵ and other constants, we get

$$\mathrm{P}_Z \left\{ \cap^m A_j \right\} \geq 1 - m e^{-np_0} \to 1$$

as $n \to \infty$. Finally, note that throughout the proof P_Z is the same as $P_{X,\Gamma}$ □

Lemma 12. *Let $\mathcal{X}$, μ satisfy A0, $\{(X_i, \Gamma_i)\}_{i=1}^n$ satisfy A1, and k satisfy A2. Then, $\forall$ $\epsilon > 0$ $\exists C$ such that*

$$P_{X,\Gamma}\left\{\inf_{\alpha \in A_n} \|f - f_\alpha\|_2 > \epsilon\right\} \leq Ce^{-np_0}$$

for sufficiently large n.

Proof. Let $\delta_2 = m\exp(-np_0)$. With probability $\geq 1 - \delta_2$ we have that for every j there is an i_j such that $\left\|k(\cdot, z_{i_j}) - k(\cdot, y_j)\right\|_2 \leq \frac{\epsilon}{\|\beta\|_1}$. Then, for α^* defined as

$$\alpha_i^* = \begin{cases} \beta_j & : i = i_j \\ 0 & : i \neq i_j \end{cases}$$

we have

$$\begin{aligned} \|f_{\alpha^*} - f_\beta\|_2 &= \left\|\sum_{i=1}^n \alpha_i^* k(\cdot, z_i) - \sum_{j=1}^m \beta_j k(\cdot, y_j)\right\|_2 \\ &\leq \sum_{j=1}^m |\beta_j| \left\|k(\cdot, z_{i_j}) - k(\cdot, y_j)\right\|_2 \\ &\leq \sum_{j=1}^m |\beta_j| \, \mu^{1/2}(\mathcal{X}) \frac{\epsilon}{\|\beta\|_1} \\ &= \mu^{1/2}(\mathcal{X})\epsilon. \end{aligned}$$

Note that if for two sequences $\{s_n\}, \{s_n'\}$ we have $s_n \leq s_n'$ for $n \geq N_0$, then $\lim_{n\to\infty} s_n \leq \lim_{n\to\infty} s_n'$, granted such limits exist. Let $s_n := \mathrm{P}_{X,\Gamma}\left\{\inf_{\alpha\in A_n} \|f - f_\alpha\|_2 > 3\mu^{1/2}(\mathcal{X})\epsilon\right\}$ and $s_n' := \delta_2$, note that for n large enough, say $n = N_0$, $R_n \geq \|\beta\|_1$ and therefore $\alpha^* \in A_n$. So we can see that for $n \geq N_0$, the inequality

$$\begin{aligned} \inf_{\alpha \in A_n} \|f - f_\alpha\|_2 &\leq \|f - f_{\alpha^*}\|_2 \\ &\leq \|f - f_{\mathcal{H}}\|_2 + \|f_{\mathcal{H}} - f_\beta\|_2 + \|f_\beta - f_{\alpha^*}\|_2 \\ &\leq 3\mu^{1/2}(\mathcal{X})\epsilon \end{aligned}$$

holds with probability $\geq 1 - \delta_2$. That is, for $n \geq N_0$, $s_n \leq s_n'$, hence since $s_n' = \delta_2 \to 0$, we get $s_n \to 0$. □

Proof of Theorem 3.

$$\begin{aligned} &\mathrm{P}_{X,\Gamma}\left\{J(\alpha^{(n)}) \leq 4\epsilon + \left(3\mu^{1/2}(\mathcal{X})\epsilon\right)^2\right\} \qquad (4.7) \\ &\geq \mathrm{P}_{X,\Gamma}\left\{\left\{J(\alpha^{(n)}) \leq J^* + 4\epsilon\right\} \cap \left\{J^* \leq \left(3\mu^{1/2}(\mathcal{X})\epsilon\right)^2\right\}\right\} \\ &= 1 - \mathrm{P}_{X,\Gamma}\left\{\left\{J(\alpha^{(n)}) \leq J^* + 4\epsilon\right\}^C \cup \left\{J^* \leq \left(3\mu^{1/2}(\mathcal{X})\epsilon\right)^2\right\}^C\right\} \\ &\geq 1 - \left(\mathrm{P}_{X,\Gamma}\left\{J(\alpha^{(n)}) > J^* + 4\epsilon\right\} + \mathrm{P}_{X,\Gamma}\left\{J^* > \left(3\mu^{1/2}(\mathcal{X})\epsilon\right)^2\right\}\right). \qquad (4.8) \end{aligned}$$

By Lemma 5 the middle term approaches zero and by Lemma 12 the last term does, so

$$\lim_{n\to\infty} \mathrm{P}_{X,\Gamma}\left\{J(\alpha^{(n)}) \leq \epsilon'\right\} = 1,$$

where $\epsilon' = 4\epsilon + \left(3\mu^{1/2}(\mathcal{X})\epsilon\right)^2$. □

Convergence Rates of $f_{\alpha^{(n)}}$

The proof of Lemma 7 is found in [81]. Recall Lemma 8:

Lemma 8 Let $\delta_2 \in (0,1)$, let $f \in \mathcal{F}_k$ and let f_β and m be as in (4.5). Let $\{(X_i, \Gamma_i)\}_{i=1}^n$ satisfy assumption A1, then with probability $\geq 1 - \delta_2$

$$\inf_{\alpha \in A_n} \|f_\beta - f_\alpha\|_2 \leq \epsilon_3(n,m)$$

where $\epsilon_3(n,m) := \frac{C}{n^{1/d}} \log^{1/d}(m/\delta_2)$, for C a constant independent of n and m. □

Proof of Lemma 8. Throughout this proof define f_β as in Lemma 7. Let $\eta := \frac{\epsilon_3}{L\|\beta\|_1}$. Following an argument similar to that of Lemmas 11 and 12 we know that with probability $\geq 1 - me^{-np_0}$ the event that for all y_j there is a data point Z_{i_j} such that $\left\|y_j - Z_{i_j}\right\|_2 \leq \eta$ will hold. Recall $p_0 = \min_j \int_{B_j} f_Z(z)dz$, hence

$$\begin{aligned} p_0 &= \min_j \int_{B_j} f_Z(z)dz \\ &\geq \min_{x \in \mathcal{X}} \{f_Z(z)\} \min_j \int_{B_j} dz \\ &\geq \min_{x \in \mathcal{X}} \{f_Z(z)\} \min_j vol(B_j) \\ &= \min_{x \in \mathcal{X}} \{f_Z(z)\} c' \eta^d, \end{aligned}$$

where c' is the volume of the d-dimensional unit ball.

Now pick the α coefficients as in Lemma 11, then with probability $\geq 1 - me^{-nc\eta^d}$ we have:

$$\begin{aligned} \inf_{\alpha \in A_n} \|f_\beta - f_\alpha\|_2 &\leq \left\| \sum_{j=1}^m \beta_j \left(k(\cdot, y_j) - k(\cdot, z_{i_j}) \right) \right\|_2 \\ &\leq L\mu^{1/2}(\mathcal{X}) \|\beta\|_1 \eta \\ &= \mu^{1/2}(\mathcal{X}) \epsilon_3. \end{aligned}$$

Fixing $\delta_2 \in (0,1)$ and setting $me^{-nc\eta^d(\epsilon)} = \delta_2$, where $c = c' \min_{x \in \mathcal{X}} \{f^*(x)\}$, we obtain

$$\epsilon_3 = \frac{L\|\beta\|_1}{(nc)^{1/d}} \log^{1/d}(m/\delta_2).$$

Finally, noting that $\|\beta\|_1 = \|\lambda\|_1$, where λ is as in Lemma 7, yields the desired result. □

We now prove Theorem 4.

Proof of Theorem 4. We will use the notation and results of Lemmas 6, 7 and 8. First, note that from Lemma 7 we have $\|f - f_\beta\|_2 \leq \mu^{1/2}(\mathcal{X}) \|f - f_\beta\|_\infty \leq \mu^{1/2}(\mathcal{X}) \epsilon_2(m)$. So putting the three Lemmas together we have that with probability $\geq 1 - (\delta_1 + \delta_2)$

$$\|f - f_{\alpha^{(n)}}\|_2^2 \leq \epsilon_1(n) + \mu(\mathcal{X})(\epsilon_2(m) + \epsilon_3(n,m))^2,$$

or, taking into account only the dependence on n, m, δ_1 and δ_2 for the different ϵ_i's, we have that $\|f - f_{\alpha^{(n)}}\|_2^2$ is of the order of

$$\|f - f_{\alpha^{(n)}}\|_2^2 \lesssim \log^{1/2}(n/\delta_1)\frac{R_n}{n^{1/2}} + \frac{1}{m} + \log^{2/d}(m/\delta_2)\frac{1}{n^{2/d}}.$$

Now, we want the number m of centers in f_β close to but no larger than the number n of data points, so we set $m = n^\theta$ for some θ such that $0 < \theta < 1$. Furthermore, we need R_n to grow accordingly, so that $R_n = n^{c\theta}$, for $c > 0$ a constant possibly dependent on d. This yields, ignoring the log terms for now:

$$\|f - f_{\alpha^{(n)}}\|_2^2 \lesssim \frac{1}{n^{1/2-c\theta}} + \frac{1}{n^\theta} + \frac{1}{n^{2/d}}.$$

Setting the first two rates equal we obtain $\theta = 1/2(1+c)$. Note that if $d > 4$ we can match the third term by setting $c = d/4 - 1$ to obtain an overall rate of $2/d$. Otherwise we can set c to any small number to obtain a rate $1/2(1+c)$ slightly slower than $1/2$.

Finally, for the log terms, just let $\delta_1 = \delta_2 = \delta/2$ and note that if $d > 4$ then $\log^{1/2}$ dominates, and if $d \leq 4$ then $\log^{2/d}$ dominates. □

Box rates

In this section we use the assumptions and notation from Section 4.4.3 To sketch the proof we begin by reformulating Lemma 5 for the box kernel case:

Lemma 13. *Let $\delta \in (0,1)$. Let $\{(X_i)\}_{i=1}^n$ satisfy A1 and k satisfy $\sup_{x,x' \in \mathcal{X}} k(x,x') \leq C_k$. If $\tilde{f}_{\alpha^{(n)}}$ is as above, then with probability $\geq 1-\delta$*

$$J(\alpha^{(n)}) \leq \epsilon_1(n) + \inf_{\alpha \in A_n} \left\| f - \tilde{f}_\alpha \right\|_2$$

where $\epsilon_1(n) := \sqrt{8} C_k R_M \sqrt{\frac{\log(4M/\delta_1)}{n-1}}$. □

We now take care of the term $\inf_{\alpha \in A_n} \left\| f - \tilde{f}_\alpha \right\|_2$.

Lemma 14. *Let $f \in \mathcal{F}$. For any $m \in \mathbb{N}$ there is a function f_β of the form $f_\beta = \sum_{i=1}^{(mq)^d} \beta_i k(\cdot, y_i)$, where $\{y_i\}_{i=1}^{(mq)^d} \subset \mathcal{X} \pm \sigma$ and $\|\beta\|_1 \leq (mq)^{d-1} R_{f,\sigma}$ satisfying*

$$\|f - f_\beta\|_2 \leq \epsilon_2(m)$$

where $\epsilon_2(m) := \frac{L_f \sqrt{d}}{mq}$. □

Proof of Lemma 14. Let $\iota := (\iota_1, \ldots, \iota_d)$ be a multi-index with positive elements and associated index i related by the function h:

$$i = h(\iota) = 1 + \sum_{\ell=1}^{d} (\iota_\ell - 1)(mq)^{\ell-1}$$

and its inverse

$$h^{-1}(i) = \left(\left\lceil \frac{i \mod (mq)^1}{(mq)^0} \right\rceil, \left\lceil \frac{i \mod (mq)^2}{(mq)^1} \right\rceil, \ldots, \left\lceil \frac{i \mod (mq)^d}{(mq)^{d-1}} \right\rceil \right).$$

Divide $\mathcal{X} = [0,1]^d$ into $(mq)^d$ hypercube regions of equal volume to form the partition $\left\{\Pi_{\ell=1}^d \left[\frac{\iota_\ell - 1}{mq}, \frac{\iota_\ell}{mq}\right]\right\}_{\iota_1,\ldots,\iota_d=1}^{(mq)} = \{T_i\}_{i=1}^{(mq)^d}$, where $T_i = \Pi_{\ell=1}^d \left[\frac{\iota_\ell - 1}{mq}, \frac{\iota_\ell}{mq}\right]$. Now, let

$$f_m = \sum_{i=1}^{(mq)^d} f(\overline{x}_i)\mathbf{1}_{\{T_i\}}$$

where $\overline{x}_i \in T_i$. Any choice of $\overline{x}_i$ works but for clarity we choose $\overline{x}_i = (\iota_1/mq, \cdots, \iota_d/mq)$. Note that f_m is close to f:

$$\begin{aligned}
\|f - f_m\|_2^2 &\leq \int_{\mathcal{X}} (f(x) - f_m(x))^2 dx \\
&= \sum_{i=1}^{(mq)^d} \int_{T_i} \left(f(x) - f_m(x)\right)^2 dx \\
&= \sum_{i=1}^{(mq)^d} \int_{T_i} \left(f(x) - f(\overline{x}_i)\right)^2 dx \\
&\leq \sum_{i=1}^{(mq)^d} \int_{T_i} \left(L_f \|x - \overline{x}_i\|_2\right)^2 dx \\
&\leq \sum_{i=1}^{(mq)^d} \int_{T_i} \left(L_f \frac{\sqrt{d}}{mq}\right)^2 dx \\
&\leq \left(\frac{L_f\sqrt{d}}{mq}\right)^2.
\end{aligned}$$

Hence

$$\|f - f_m\|_2 \leq \frac{L_f\sqrt{d}}{mq}.$$

Now we note that f_m can also be expressed as a sum of fixed bandwidth kernels:

$$f_m = \sum_{i=1}^{(mq)^d} \beta_i (2\sigma)^d k(\cdot, y_i),$$

where

$$y_i = \left[\frac{\iota_1 - 1}{mq} + \sigma, \cdots, \frac{\iota_d - 1}{mq} + \sigma\right]^T$$

and β is as follows. Let $\beta_1 = f(\overline{x}_1)$ and

$$\beta_i = f(\overline{x}_i) - \sum_{\kappa=s_i}^{S_i} \beta_{h(\kappa)} - f(0)\mathbf{1}_{\{\iota_\ell = 1 \ \forall \ell\}}$$

for $i \leq 2 \leq (mq)^d$, where $s_i = (\max\{1, \iota_1 - (m-1)\}, \cdots, \max\{1, \iota_d - (m-1)\})$ and $S_i = (\max\{1, \iota_1 - 1\}, \cdots, \max 1, \iota_d - 1)$ are multi-indices. Note that the β_i's sequentially capture the residual of the function f_m as we travel along the T_i regions.

To find $\|\beta\|_1$ note that since $\|\overline{x}_1 - 0\|_2 \leq L_f\sqrt{d}/(mq)$, we have $|\beta_1| \leq f(0) + L_f\sqrt{d}/(mq)$. Also note $\beta_i = f(\overline{x}_i) - f(\overline{x}_{i-1})$ for $2 \leq \iota_1 \leq m$, hence for $2 \leq i \leq m$

$$\begin{aligned}|\beta_i| &= |f(\overline{x}_i) - f(\overline{x}_{i-1})| \\ &\leq L_f \|\overline{x}_i - \overline{x}_{i-1}\|_2 \\ &\leq L_f \frac{\sqrt{d}}{mq}\end{aligned}$$

For larger ι_1 we have $\beta_i = f(\overline{x}_i) - f(\overline{x}_{i-1}) + \beta_{i-m} - f(0)\mathbf{1}_{\{\iota_\ell = 1\ \forall \ell\}}$. Note that when $\iota_1 = m+1$ we have lost influence of β_1, so

$$|\beta_{m+1}| \leq |f(\overline{x}_i) - f(\overline{x}_{i-1})| + |\beta_1| + f(0) \leq 2L_f \frac{\sqrt{d}}{mq} + 2f(0)$$

and, similarly, $|\beta_i| \leq 2L_f \frac{\sqrt{d}}{mq}$ for $m+2 \leq i \leq 2m$. The process continues such that, in general

$$|\beta_i| \leq \left\lceil \frac{\iota_1}{m} \right\rceil \left(L_f \frac{\sqrt{d}}{mq} + f(0)\mathbf{1}_{\{\iota_\ell = 1\ \forall \ell\}} \right)$$

for $i \leq mq$. Adding these we obtain $\frac{q+1}{2}(L_f\sqrt{d} + qf_0)$. Denote this quantity by e'. Then this process is repeated over every dimension, having q chunks of multiples of e', the first multiple is $1m$, the second $2m$, and so on. The final sum is then

$$\|\beta\|_1 \leq (mq)^{d-1} \left(\frac{q+1}{2} \right)^2 \left(qf(0) + L_f\sqrt{d} \right).$$

Therefore

$$(2\sigma)^d \|\beta\|_1 = \frac{(m)^{d-1}}{q} \left(\frac{q+1}{2} \right)^2 \left(qf(0) + L_f\sqrt{d} \right).$$

□

Proof of Theorem 5. This proof is similar to the proof of Theorem P 4. Combining Lemmas 13, 14, setting $R_{n'} \sim m^{(d-1)}$, ignoring the *log* terms for now and considering only the dependence on n and δ we obtain

$$\|f - f_{\alpha^{(n)}}\|_2^2 \lesssim \frac{m^{d-1}}{n^{1/2}} + \frac{1}{m^2}.$$

Setting these terms equal we obtain $m = n^{1/(2d+2)}$, with an overall rate of $n^{-1/(d+1)}$. Adding the log term we obtain

$$\|f - f_{\alpha^{(n)}}\|_2^2 \lesssim \frac{\log^{1/2}(n^{d/(2d+2)}/\delta)}{n^{1/(d+1)}} \lesssim \frac{\log^{1/2}(n/\delta)}{n^{1/(d+1)}}.$$

□

Chapter 5

Concluding Remarks

In this work we have studied modification of kernel nonparametric estimators. In particular, we addressed two issues of importance for kernel estimators: scalability and bandwidth choice.

To address the first issue we presented a scalable sparse representation of a kernel mean, which we call sparse kernel mean (SKM) and prove approximation guarantees. To show its performance with respect to other competing methods, and to the full estimators, we compared the relative error for simple benchmark datasets. We also deviced three experiments, eulidean embedding of distributions, class proportion estimation, and mean shift clustering, in all of which the fbKDE is more accurate and faster than its competitors, some times by an order of magnitude, and it is about two of magnitude faster than the full procedure. Future research involves a similar framework in which SKM is modified so that the bandwidth of the selected components is larger than the original bandwidth. Also, due to research done on the k-center algorithm, it is possible to extend SKM so as to be robust or accommodate easily for online streaming of data, and yet remain computationally feasible.

To address the second issue, and in the specific case of kernel density estimation, we present a novel estimator, the fixed bandwidth kernel density estimator (fbKDE), which exhibits statistical consistency without the need for the kernel bandwidth parameter to approach zero. We showed that for suboptimal fixed bandwidth the fbKDE outperforms the KDE and vKDE as the sample size grows. We also showed that even for optimal choices of bandwidth the fbKDE performs better, especially in terms of the uniform norm. Future work points to studying non-universal kernels with possibly weaker approximating properties. The fbKDE based on these kernels might only be consistent for smaller function families but have possibly faster rates. Also, we conjecture our results can be expanded to dependent data and more general structured domains, not just euclidean. Finally, a combination of SKME and fbKDE remains to be done, if so, it would provide a powerful tool for nonparametric estimation.

Bibliography

[1] E. Fix and J. L. Hodges Jr, "Discriminatory analysis-nonparametric discrimination: consistency properties," Tech. Rep. 21-49-004, USAF School of Aviation Medicine, Report No. 4, 1951.

[2] M. Rosenblatt, "Remarks on some nonparametric estimates of a density function," *The Annals of Mathematical Statistics*, vol. 27, no. 3, pp. 832–837, 1956.

[3] E. Parzen, "On estimation of a probability density function and mode," *The Annals of Mathematical Statistics*, vol. 33, no. 3, pp. 1065–1076, 1962.

[4] D. M. Titterington, G. D. Murray, L. S. Murray, D. J. Spiegelhalter, A. M. Skene, J. D. F. Habbema, and G. J. Gelpke, "Comparison of discrimination techniques applied to a complex data set of head injured patients," *Journal of the Royal Statistical Society. Series A (General)*, pp. 145–175, 1981.

[5] D. J. Hand, "A comparison of two methods of discriminant analysis applied to binary data," *Biometrics*, pp. 683–694, 1983.

[6] M. J. Desforges, P. J. Jacob, and J. E. Cooper, "Applications of probability density estimation to the detection of abnormal conditions in engineering," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 212, no. 8, pp. 687–703, 1998.

[7] D. Yeung and C. Chow, "Parzen-window network intrusion detectors," in *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, vol. 4, pp. 385–388, IEEE, 2002.

[8] M. Markou and S. Singh, "Novelty detection: a review—part 1: statistical approaches," *Signal processing*, vol. 83, no. 12, pp. 2481–2497, 2003.

[9] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: a survey," *ACM Computing Surveys (CSUR)*, vol. 41, no. 3, p. 15, 2009.

[10] Y. Cheng, "Mean shift, mode seeking, and clustering," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 17, no. 8, pp. 790–799, 1995.

[11] B. W. Silverman, *Density Estimation for Statistics and Data Analysis*. London: Chapman and Hall, 1986.

[12] D. W. Scott, *Multivariate Density Estimation*. New York: Wiley, 1992.

[13] M. P. Wand and M. C. Jones, *Kernel Smoothing*. Chapman & Hall/CRC, 1994.

[14] L. Devroye and G. Lugosi, *Combinatorial Methods in Density Estimation*. New York: Springer, 2001.

[15] N.-B. Heidenreich, A. Schindler, and S. Sperlich, "Bandwidth selection for kernel density estimation: a review of fully automatic selectors," *AStA Advances in Statistical Analysis*, vol. 97, no. 4, pp. 403–433, 2013.

[16] I. Steinwart and A. Christmann, *Support Vector Machines*. Springer, 2008.

[17] C. Scovel, D. Hush, I. Steinwart, and J. Theiler, "Radial kernels and their reproducing kernel hilbert spaces," *Journal of Complexity*, vol. 26, no. 6, pp. 641–660, 2010.

[18] D. Wied and R. Weißbach, "Consistency of the kernel density estimator: a survey," *Statistical Papers*, vol. 53, no. 1, pp. 1–21, 2012.

[19] V. I. Paulsen and M. Raghupathi, *An introduction to the theory of reproducing kernel Hilbert spaces*, vol. 152. Cambridge University Press, 2016.

[20] D. Alpay, ed., *Operator Theory*. Springer Basel, 2015.

[21] E. Cruz Cortés and C. Scott, "Scalable sparse approximation of a sample mean," in *Proc. 2014 IEEE Int. Conf. on Acoustic, Speech and Signal Processing (ICASSP)*, pp. 5274–5278, 2014.

[22] E. Cruz Cortés and C. Scott, "SKM Matlab code." `http://web.eecs.umich.edu/~cscott/code.html#skm`, 2015. Last accessed February-2015.

[23] A. Smola, A. Gretton, L. Song, and B. Schölkopf, "A Hilbert space embedding for distributions," in *Algorithmic Learning Theory*, pp. 13–31, Springer, 2007.

[24] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *The Journal of Machine Learning Research*, vol. 13, no. 1, pp. 723–773, 2012.

[25] K. Fukumizu, L. Song, and A. Gretton, "Kernel Bayes' rule," in *Advances in neural information processing systems*, pp. 1737–1745, 2011.

[26] P. Gurram and H. Kwon, "Contextual SVM for hyperspectral classification using Hilbert space embedding," in *Geoscience and Remote Sensing Symposium (IGARSS), 2012 IEEE International*, pp. 5470–5473, IEEE, 2012.

[27] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *Information Theory, IEEE Transactions on*, vol. 50, no. 10, pp. 2231–2242, 2004.

[28] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *Signal Processing, IEEE Transactions on*, vol. 41, no. 12, pp. 3397–3415, 1993.

[29] Y. Chen, M. Welling, and A. Smola, "Super-samples from kernel herding," in *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2010.

[30] F. Bach, S. Lacoste-Julien, and G. Obozinski, "On the equivalence between herding and conditional gradient algorithms," in *Proceedings of the 29th International Conference on Machine Learning (ICML)*, 2012.

[31] B. Jeon and D. A. Landgrebe, "Fast parzen density estimation using clustering-based branch and bound," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 16, no. 9, pp. 950–954, 1994.

[32] M. Girolami and C. He, "Probability density estimation from optimally condensed data samples," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 25, no. 10, pp. 1253–1264, 2003.

[33] S. Chen, X. Hong, and C. J. Harris, "An orthogonal forward regression technique for sparse kernel density estimation," *Neurocomputing*, vol. 71, no. 4, pp. 931–943, 2008.

[34] M. Schafföner, E. Andelic, M. Katz, S. E. Krüger, and A. Wendemuth, "Memory-effcient orthogonal least squares kernel density estimation using enhanced empirical cumulative distribution functions," in *International Conference on Artificial Intelligence and Statistics*, pp. 428–435, 2007.

[35] D. W. Scott and W. F. Szewczyk, "From kernels to mixtures," *Technometrics*, vol. 43, no. 3, pp. 323–335, 2001.

[36] A. R. Runnalls, "Kullback-Leibler approach to Gaussian mixture reduction," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 43, no. 3, pp. 989–999, 2007.

[37] D. Schieferdecker and M. F. Huber, "Gaussian mixture reduction via clustering," in *Information Fusion, 2009. FUSION'09. 12th International Conference on*, pp. 1536–1543, IEEE, 2009.

[38] M. A. T. Figueiredo and A. K. Jain, "Unsupervised learning of finite mixture models," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 3, pp. 381–396, 2002.

[39] P. Bruneau, M. Gelgon, and F. Picarougne, "Parsimonious reduction of Gaussian mixture models with a variational-Bayes approach," *Pattern Recognition*, vol. 43, no. 3, pp. 850–858, 2010.

[40] L. Greengard and V. Rokhlin, "A fast algorithm for particle simulations," *Journal of computational physics*, vol. 73, no. 2, pp. 325–348, 1987.

[41] A. G. Gray and A. W. Moore, "N-body problems in statistical learning," in *NIPS*, vol. 4, pp. 521–527, 2000.

[42] C. Yang, R. Duraiswami, N. A. Gumerov, and L. Davis, "Improved fast Gauss transform and efficient kernel density estimation," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pp. 664–671, IEEE, 2003.

[43] D. Lee, A. Gray, and A. W. Moore, "Dual-tree fast Gauss transforms," in *Advances in Neural Information Processing Systems 18 (Dec 2005)* (Y. Weiss, B. Scholkopf, and J. Platt, eds.), MIT Press, 2006.

[44] Y. Zheng, J. Jestes, J. M. Phillips, and F. Li, "Quality and efficiency for kernel density estimates in large data," in *Proceedings of the 2013 international conference on Management of data*, pp. 433–444, ACM, 2013.

[45] J. M. Phillips, "ε-samples for kernels," in *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1622–1632, SIAM, 2013.

[46] S. Joshi, R. V. Kommaraji, J. M. Phillips, and S. Venkatasubramanian, "Comparing distributions and shapes using the kernel distance," in *Proceedings of the twenty-seventh annual symposium on Computational geometry*, pp. 47–56, ACM, 2011.

[47] J. Zhao and D. Meng, "FastMMD: Ensemble of circular discrepancy for efficient two-sample test," *NIPS Workshop on Randomized Methods for Machine Learning*, 2013.

[48] P. Drineas and M. W. Mahoney, "On the Nyström method for approximating a gram matrix for improved kernel-based learning," *The Journal of Machine Learning Research*, vol. 6, pp. 2153–2175, 2005.

[49] S. Kumar, M. Mohri, and A. Talwalkar, "Sampling methods for the Nyström method," *The Journal of Machine Learning Research*, vol. 13, no. 1, pp. 981–1006, 2012.

[50] W. B. March and G. Biros, "Far-field compression for fast kernel summation methods in high dimensions," *arXiv preprint arXiv:1409.2802*, 2014.

[51] Z. Noumir, P. Honeine, and C. R., "One-class machines based on the coherence criterion," in *Statistical Signal Processing Workshop (SSP), 2012 IEEE*, pp. 600–603, IEEE, 2012.

[52] V. V. Vazirani, *Approximation algorithms*. Springer, 2001.

[53] T. F. Gonzalez, "Clustering to minimize the maximum intercluster distance," *Theoretical Computer Science*, vol. 38, pp. 293–306, 1985.

[54] D. S. Hochbaum, *Approximation algorithms for NP-hard problems*. PWS Publishing Co., 1996.

[55] J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra, "Efficient projections onto the ℓ_1-ball for learning in high dimensions," in *Proceedings of the 25th international conference on Machine learning*, pp. 272–279, ACM, 2008.

[56] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, pp. 2319–2323, 2000.

[57] W. G. Finn, K. M. Carter, R. Raich, L. M. Stoolman, and A. O. Hero, "Analysis of clinical flow cytometric immunophenotyping data by clustering on statistical manifolds: Treating flow cytometry data as high-dimensional objects," *Cytometry Part B: Clinical Cytometry*, vol. 76, no. 1, pp. 1–7, 2009.

[58] P. Hall, "On the non-parametric estimation of mixture proportions," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 147–156, 1981.

[59] D. M. Titterington, "Minimum distance non-parametric estimation of mixture proportions," *Journal of the Royal Statistical Society*, vol. 45, no. 1, pp. 37–46, 1983.

[60] T. Sanderson and C. Scott, "Class proportion estimation with application to multiclass anomaly rejection," in *Proceedings of the 17th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2014.

[61] Y. LeCun, "The mnist database." `http://yann.lecun.com/exdb/mnist`, 2014. Last accessed 24-September-2014.

[62] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[63] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 5, pp. 603–619, 2002.

[64] B. Finkston, "Mean shift clustering." `http://www.mathworks.com/matlabcentral/fileexchange/10161-mean-shift-clustering`, 2014. Last accessed 24-September-2014.

[65] Y. Chen, C. R. Genovese, and L. Wasserman, "Enhanced mode clustering," *arXiv preprint arXiv:1406.1780*, 2014.

[66] A. Gionis, P. Indyk, and R. Motwani, "Similarity search in high dimensions via hashing," in *VLDB*, vol. 99, pp. 518–529, 1999.

[67] A. Andoni and P. Indyk, "Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions," in *Foundations of Computer Science, 2006. FOCS'06. 47th Annual IEEE Symposium on*, pp. 459–468, IEEE, 2006.

[68] G. Shakhnarovich, "Locality Sensitive Hashing." `http://ttic.uchicago.edu/~gregory`, 2014. Last accessed 24-September-2014.

[69] A. Andoni, "LSH algorithm and implementation." `http://www.mit.edu/~andoni/LSH`, 2014. Last accessed 24-September-2014.

[70] J. E. Chacón, "A population background for nonparametric density-based clustering," *arXiv preprint arXiv:1408.1381*, 2014.

[71] T. Jaakkola, M. Diekhans, and D. Haussler, "Using the Fisher kernel method to detect remote protein homologies.," in *ISMB*, vol. 99, pp. 149–158, 1999.

[72] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics bulletin*, vol. 1, no. 6, pp. 80–83, 1945.

[73] D. Kim, *Least Squares Mixture Decomposition Estimation*. Phd thesis, Dept. of Statistics, Virginia Polytechnic Inst. and State Univ., 1995.

[74] M. Girolami and C. He, "Probability density estimation from optimally condensed data samples," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, pp. 1253–1264, OCT 2003.

[75] R. Ganti and A. G. Gray, "Cake: Convex adaptive kernel density estimation," in *International Conference on Artificial Intelligence and Statistics*, pp. 498–506, 2011.

[76] J. Kim and C. D. Scott, "Robust kernel density estimation," *The Journal of Machine Learning Research*, vol. 13, no. 1, pp. 2529–2565, 2012.

[77] J. Kim and C. D. Scott, "L2 kernel classification," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 10, pp. 1822–1831, 2010.

[78] P. Rigollet and A. B. Tsybakov, "Linear and convex aggregation of density estimators," *Mathematical Methods of Statistics*, vol. 16, no. 3, pp. 260–280, 2007.

[79] E. Cruz Cortés and C. Scott, "Sparse approximation of a kernel mean," *IEEE Transactions n Signal Processing*, vol. 65, pp. 1310–1323.

[80] R. Vert and J.-P. Vert, "Consistency and convergence rates of one-class SVM and related algorithms," *J. Machine Learning Research*, pp. 817–854, 2006.

[81] G. Gnecco and M. Sanguineti, "Approximation error bounds via Rademacher's complexity," *Applied Mathematical Sciences*, vol. 2, no. 4, pp. 153–176, 2008.

[82] F. Girosi, "Approximation error bounds that use vc-bounds," in *Proc. International Conference on Artificial Neural Networks, F. Fogelman-Soulie and P. Gallinari, editors*, vol. 1, pp. 295–302, 1995.

[83] M. A. Kon and L. A. Raphael, "Approximating functions in reproducing kernel Hilbert spaces via statistical learning theory," *Wavelets and Splines: Athens 2005*, pp. 271–286, 2006.

[84] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[85] D. Comaniciu, V. Ramesh, and P. Meer, "The variable bandwidth mean shift and data-driven scale selection," in *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, vol. 1, pp. 438–445, IEEE, 2001.

[86] S. Arlot and M. Lerasle, "Choice of v for v-fold cross-validation in least-squares density estimation," *Journal of Machine Learning Research*, vol. 17, no. 208, pp. 1–50, 2016.