# Thesis/Dissertation Sheet

| | | |
|---|---|---|
| Surname/Family Name | : | **Aya Moreno** |
| Given Name/s | : | **Carlos Enrique** |
| Abbreviation for degree as give in the University calendar | : | **PhD** |
| Faculty | : | **Faculty of Science** |
| School | : | **School of Mathematics and Statistics** |
| Thesis Title | : | **Shape-preserving wavelet-based density estimation with applications to image analysis** |

**Abstract 350 words maximum: (PLEASE TYPE)**

Wavelet estimators for a probability density enjoy many good properties; however, they are not shape-preserving in the sense that the final estimate may be negative nor integrate to unity. A solution to negativity issues may be to estimate first the square-root of the density and then square this estimate up. In this thesis, we propose and investigate such an estimation scheme, generalising to higher dimensions a previous construction of Penev and Dechevsky (1997), which is valid only in one dimension, using nearest-neighbour balls. The theoretical properties of the proposed estimator are obtained, and it is shown to reach the optimal rate of convergence uniformly over large classes of densities under mild conditions. For spatially inhomogeneous densities and in general, there is a need to threshold the empirical wavelet coefficients in order to avoid over-fitting. In the case of density estimation, the most common approach is to use cross-validation over a likelihood function. Aligned with our results, we provide a principled alternative using a cross-validation type approach over an empirical approximation to the Bhattacharyya coefficient and the associated Hellinger distance, which is suitable when the square-root of the density is estimated. The effectiveness of these data-driven algorithms is demonstrated via Monte Carlo simulations and a thorough review of their usage in the traditional Old Faithful geyser dataset. Finally, we aim to extend these tools and applications to the raising field of intrinsic statistics in Riemannian manifolds and present an example on how techniques based on k-th nearest neighbours can be applied in image analysis using the MNIST and Fashion-MNIST datasets.

Declaration relating to disposition of project thesis/dissertation

……………………………………………………………… ……………………………………

Signature Date

## COPYRIGHT STATEMENT

‘I hereby grant the University of New South Wales or its agents a non-exclusive licence to archive and to make available (including to members of the public) my thesis or dissertation in whole or part in the University libraries in all forms of media, now or here after known. I acknowledge that I retain all intellectual property rights which subsist in my thesis or dissertation, such as copyright and patent rights, subject to applicable law. I also retain the right to use all or part of my thesis or dissertation in future works (such as articles or books).’

‘For any substantial portions of copyright material used in this thesis, written permission for use has been obtained, or the copyright material is removed from the final public version of the thesis.’

Signed …………………………………………...........................

Date ……………………………………………...........................

## AUTHENTICITY STATEMENT

‘I certify that the Library deposit digital copy is a direct equivalent of the final officially approved version of my thesis.’

Signed …………………………………………...........................

Date ……………………………………………...........................

UNSW SYDNEY | Australia's Global University

# INCLUSION OF PUBLICATIONS STATEMENT

UNSW is supportive of candidates publishing their research results during their candidature as detailed in the UNSW Thesis Examination Procedure.

**Publications can be used in their thesis in lieu of a Chapter if:**

- The candidate contributed greater than 50% of the content in the publication and is the "primary author", ie. the candidate was responsible primarily for the planning, execution and preparation of the work for publication
- The candidate has approval to include the publication in their thesis in lieu of a Chapter from their supervisor and Postgraduate Coordinator.
- The publication is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in the thesis

Please indicate whether this thesis contains published material or not:

□ This thesis contains no publications, either published or submitted for publication
*(if this box is checked, you may delete all the material on page 2)*

□ Some of the work described in this thesis has been published and it has been documented in the relevant Chapters with acknowledgement
*(if this box is checked, you may delete all the material on page 2)*

☒ This thesis has publications (either published or submitted for publication) incorporated into it in lieu of a chapter and the details are presented below

**CANDIDATE'S DECLARATION**

I declare that:

- I have complied with the UNSW Thesis Examination Procedure
- where I have used a publication in lieu of a Chapter, the listed publication(s) below meet(s) the requirements to be included in the thesis.

| Candidate's Name | Signature | Date (dd/mm/yy) |
|---|---|---|
|  |  |  |

**POSTGRADUATE COORDINATOR’S DECLARATION** *To only be filled in where publications are used in lieu of Chapters*

I declare that:

- the information below is accurate
- where listed publication(s) have been used in lieu of Chapter(s), their use complies with the UNSW Thesis Examination Procedure
- the minimum requirements for the format of the thesis have been met.

| | | |
|---|---|---|
| **PGC’s Name**<br>**Ian Doust** | **PGC’s Signature**<br>X | **Date (dd/mm/yy)** |

**For each publication incorporated into the thesis in lieu of a Chapter, provide all of the requested details and signatures required**

**Details of publication #1:**

*Full title:* Shape-preserving wavelet-based multivariate density estimation

*Authors:* Carlos Aya-Moreno, Gery Geenens, Spiridon Penev

*Journal or book name:* Journal of Multivariate Analysis

*Volume/page numbers:* Volume 168, Pages 30-47

*Date accepted/ published:* November 2018

| | | | | | | |
|---|---|---|---|---|---|---|
| **Status** | *Published* | ***X*** | *Accepted and In Press* | | *In progress (submitted)* | |

**The Candidate’s Contribution to the Work**

Main author: main results, proofs and simulations.

**Location of the work in the thesis and/or how the work is incorporated in the thesis:**

Cha Rer 3.

**PRIMARY SUPERVISOR’S DECLARATION**

I declare that:

- the information above is accurate
- this has been discussed with the PGC and it is agreed that this publication can be included in this thesis in lieu of a Chapter
- All of the co-authors of the publication have reviewed the above information and have a greed to its veracit y b y si gnin g a ‘Co-Author Authorisation’ form.

| | | |
|---|---|---|
| **Primary Supervisor’s name**<br>**Spiridon Penev** | **Primary Supervisor's signature** | **Date (dd/mm/yy)** |

Add additional boxes if required

# Shape-preserving wavelet-based density estimation with applications to image analysis

**Carlos Enrique Aya Moreno**

School of Mathematics and Statistics

Faculty of Science

August, 2020

Submitted in total fulfillment of the requirements
of the degree of Doctor of Philosophy

# Originality statement

‘I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person, or substantial proportions of material which have been accepted for the award of any other degree or diploma at UNSW or any other educational institution, except where due acknowledgement is made in the thesis. Any contribution made to the research by others, with whom I have worked at UNSW or elsewhere, is explicitly acknowledged in the thesis. I also declare that the intellectual content of this thesis is the product of my own work, except to the extent that assistance from others in the project’s design and conception or in style, presentation and linguistic expression is acknowledged.’

# Abstract

Wavelet estimators for a probability density enjoy many good properties; however, they are not shape-preserving in the sense that the final estimate may be negative nor integrate to unity. A solution to negativity issues may be to estimate first the square-root of the density and then square this estimate up. In this thesis, we propose and investigate such an estimation scheme, generalising to higher dimensions a previous construction of Penev and Dechevsky (1997), which is valid only in one dimension, using nearest-neighbour balls. The theoretical properties of the proposed estimator are obtained, and it is shown to reach the optimal rate of convergence uniformly over large classes of densities under mild conditions. For spatially inhomogeneous densities and in general, there is a need to threshold the empirical wavelet coefficients in order to avoid over-fitting. In the case of density estimation, the most common approach is to use cross-validation over a likelihood function. Aligned with our results, we provide a principled alternative using a cross-validation type approach over an empirical approximation to the Bhattacharyya coefficient and the associated Hellinger distance, which is suitable when the square-root of the density is estimated. The effectiveness of these data-driven algorithms is demonstrated via Monte Carlo simulations and a thorough review of their usage in the traditional Old Faithful geyser dataset. Finally, we aim to extend these tools and applications to the raising field of intrinsic statistics in Riemannian manifolds and present an example on how techniques based on $k$-th nearest neighbours can be applied in image analysis using the MNIST and Fashion-MNIST datasets.

# Contents

# Acknowledgements

First, I would like to thank my supervisor Professor Spiridon Penev for giving me the opportunity to work under his supervision and for the patience with which he guided me through out these years pursuing a research degree in statistics as a part-time student. I am also very grateful to my cosupervisor, Dr. Gery Geenens, for his insights, help, support and, in particular, for his support during the production of the paper that we submitted. This research has been quite a journey for me and I have no doubt that if this work leaves a mark in the annals of statistics it is because I have learnt a great deal of insight into the discipline from both of them. Prof. Penev, Dr. Geenens, thank you!

I also want to thank the School of Mathematics and Statistics, particularly the postgraduate coordinators Prof. David Warton and A/Prof. Ian Doust. To Prof. Warton for the initial push to get my research started few years back and to A/Prof. Doust for the final push to get my research finished. Also, within the Research Technology Services unit, the invaluable support from Duncan Smith and Martin Thompson in regards to the use of the HPC facilites and Katana in particular.

To my wife, Nancy Martin La Rotta, thank you for the endless encouragement and for enduring what appeared to be countless nights of work. This research would not have been possible without you by my side. And to my children, Sarah, Daniel and Ariane, my daily source of motivation, for their much needed smile and encouragement during all these years.

"*Every good giving and every perfect gift is from above, coming down from the Father of light.*" James 1:17a.

# Acronyms

**$k$-NN** $k$th Nearest Neighbour. 30, 36, 77, 80, 82, 86, 97

**BC** Bhattacharyya Coefficient. 46, 49, 51, 55, 57, 68–71

**CV** Cross-Validation. 39, 44, 45, 60, 66, 86

**DSP** Digital Signal Processing. 24, 94, 95

**Fashion-MNIST** Fashion MNIST. 4, 75, 76, 82, 87–89, 97

**HD** Hellinger Distance. 24, 33, 36, 39, 43, 45, 46, 48, 50, 54, 55, 59–61, 64, 66, 93, 108

**i.i.d** independent and identically distributed. 6, 20, 29, 47, 53

**ISE** Integrated Squared Error. 35

**KDE** Kernel density estimation. 6, 39, 59–62, 66, 77, 95, 96

**LOO-CV** Leave-one-out Cross-Validation. 43, 46, 48, 50, 92

**LS** Least Squares. 47

**MDL** Minimum description length. 24, 93

**MISE** Mean Integrated Squared Error. 19, 35–37, 39, 45, 107

**MNIST** Modified National Institute of Standards and Technology. 4, 74–76, 78, 82, 85, 97

**MRA** Multiresolution Analysis. 8–12, 14–16

**MSHD** Mean Squared Hellinger Distance. 35, 36, 38, 39

**OCR** Optical Character Recognition. 74

**PDF** Probability density function. 1, 6, 18, 20, 22, 35, 48, 76, 92

**SHD** Squared Hellinger Distance. 35

**SPWDE** Shape-Preserving Wavelet-based Density Estimation. 38, 39, 112, 128

# List of Figures

# List of Tables

# Chapter 1

# Introduction

Probability density functions (PDFs) are a central concept in probability and statistics. An absolutely continuous random variable $X$ with values in $\mathbb{R}$ is completely described by its PDF as it provides a way for probabilities and other properties about $X$ to be calculated, for instance

$$P(a \leq X \leq b) = \int_a^b f(d)dx.$$

The problem of density estimation is about the construction of an estimator $\hat{f}$ of $f$ from a given set of observations $X_i$, $i = 1, 2, ..., n$ of $X$. Although it shares some methods and concepts with regression, it differs from it and is an important problem on its own. If the functional form of the density is known or assumed in advance, density estimation is then the procedure to determine the underlying parameters of the distribution. For instance, one of such methods is maximum likelihood estimation, that can be traced back to Gauss and was developed in its current form and popularised by R. Fisher between 1912 and 1922 (Hald, 1999). On the other hand, when there is no formal parametric structure the methods are called nonparametric. In this case, $f$ is taken to belong to a large family of densities so that it cannot be represented by a finite number of parameters (Izenman, 1991; Simonoff, 2012; Sprent and Smeeton, 2016). Nonparametric density estimation has a rich history of methods with different assumptions and mathematical properties. The so called naïve estimator was introduced in Fix and Hodges (1951), followed by Rosenblatt (1956) and Parzen (1962) who cemented the work on kernel-based density estimation - a method widely in use today.

Towards the end of the twentieth century, the recently developed theory of wavelets seemed like a next step in the development of such estimators. The mathematical theory of wavelets offers a powerful tool for approximating possibly irregular functions or surfaces and has been successfully applied in many different fields of physics, mathematics and engineering. Classical references on the topic are Daubechies (1992); Meyer (1992), or see Strang (1989, 1993); Mohlenkamp and Pereyra (2008) for shorter reviews. In statistics, it provides a convenient framework for nonparametric density estimation and regression. Indeed, some variants of wavelet estimators are (near-) optimal in some sense over large classes of functions (Donoho and Johnstone, 1994, 1995, 1996, 1998; Fan et al., 1996; Kerkyacharian and Picard, 1993). Comprehensive reviews of wavelet methods applied to statistics can be found in Härdle et al (1998); Vidakovic (2009); Nason (2010).

One major drawback, though, of such wavelet-based estimators is that they are in general not 'shape-preserving'. When estimating a probability density $f$, the resulting estimator $\hat{f}$ may neither be non-negative, nor integrate to 1; see Dechevsky and Penev (1997, 1998). Usually, simple numerical rescaling solves the integrability issue, but overcoming the non-negativity issue requires caution. One way to address it is to first construct a wavelet estimator of $g = \sqrt{f}$ which, when squared up, would obviously produce an estimator of $f$ satisfying the non-negativity constraint. Another is to produce an estimator of $g = \log f$ (O'Sullivan (1988); Kooperberg and Stone (1991); Hazelton and Cox (2016)) and then obtain $f$ by $e^{\hat{g}}$. The former is convenient in its simplicity and, as explained below, it is the one we will pursue in this work.

When performing estimation using wavelets, a few choices have to be made to calculate the resulting estimator. Similarly to kernel-based density estimators and other nonparametric methods, wavelet-based estimators are subject to over- or under-smoothing. Thus, for a wavelet-based estimator, one needs to chose an appropriate resolution that somewhat is an optimal approximation between these two pathological extremes. Also, in kernel-based methods, it is known that the Epanechnikov kernel is optimum among a broad family of kernel functions

(Epanechnikov, 1969) although, it is worth adding, there is no much loss in *efficiency* using other popular kernels. On the other hand, the wavelet case is more complicated as one should, in principle, choose a wavelet that better reflects the underlying local structure of the derivatives of the function being approximated. The combination of this local structure and the wavelet-basis decomposition leads to a sparse representation that uses fewer terms but that is optimal in a similar sense as for the resolution. This is often done by exclusion or shrinkage of coefficients based on a thresholding rule. Again, in order to properly apply the method, one needs to be able to discover where this threshold lies. In this thesis, we shall present a data-driven method to select these meta-parameters, exploiting the structure of the estimator and the fact that we target the square root $\sqrt{f}$ of the density.

Our construction will be based on properties of nearest neighbours, extending a construction of Penev and Dechevsky (1997) made to approximate $\sqrt{f}$ in the univariate case to the multivariate setting. In his International Congress of Mathematicians (ICM) address (Donoho, 2002), D. Donoho talks about the *unreasonable effectiveness of harmonic analysis* and argues that this is due in part to the fact that "information has its own architecture", an "inner architecture" that "we should attempt to discover and exploit". Interestingly enough, our algorithm based on nearest neighbours brings to the fore the *geometry of the data*, thus taking one step further Donoho's argument about the role of harmonic analysis (and wavelet-based methods in particular) in providing the scaffold upon which those estimators are built.

This thesis is organised as follows. Chapter 2, "Foundations", has an overview of nonparametric density estimation, the theory of wavelets and the link between these two. It concludes with a short review of existing literature in shape-preserving density estimation. In Chapter 3, "A shape-preserving multivariate density estimator using wavelets", we present our estimator accompanied by several asymptotic results thus providing further insights into its properties. Simulation results compare our estimator against traditional methods. This chapter appeared in Aya-Moreno et al. (2018). Next, Chapter 4 "The non-linear shape-preserving wavelet-based density estimator", fully develops data-driven methods to determine resolution, threshold and other required parameters in the construction of the estimator. Again, asymptotic results give the reader confidence that the algorithms used are sound under some general assumptions. This chapter finishes with a practical example on how these methods are applied with real-life data. As a coda to the theoretical work of the previous chapters and to further emphasise the geometric nature of the tools, we present in Chapter 5 "Image analysis application" a short application of some of the above techniques in a practical problem in image classification, specifically using the Modified National Institute of Standards and Technology (MNIST) and Fashion MNIST (Fashion-MNIST) datasets. Finally, Chapter 6, "Discussion" summarises our results and discusses a number of potential avenues for future research.

# Chapter 2

# Foundations

## 2.1 Nonparametric density estimation

Density estimation, in the continuous setting, is the problem of finding the PDF which may have generated a given sample. The problem is usually stated as follows: given a sample of $n$ i.i.d observations $X_1, X_2, ..., X_n$, $X_i \in \Omega \subset \mathbb{R}^d$, find an underlying $f : \mathbb{R}^d \to \mathbb{R}$ PDF that "best" describes or explains the data, i.e. such that $X_i \sim f$ can be justified in some sense.

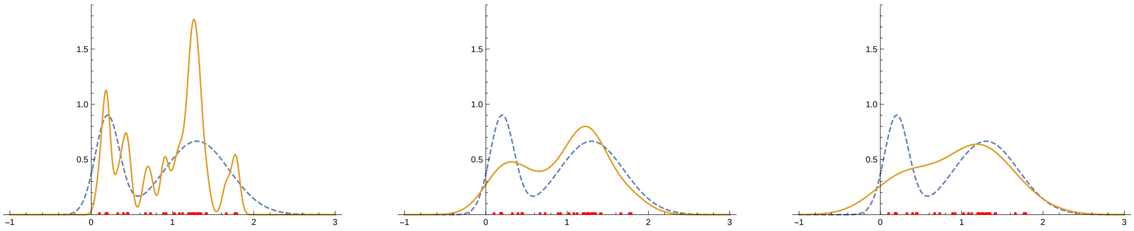
For instance, the statistician can assume as a starting point that the PDF belongs to a certain family of distributions, $\mathcal{F} = \{f_\theta | \theta \in \Theta\}$, where $\Theta$ is the parameter space. The problem above is then reduced to find a certain parameter $\hat{\theta} \in \Theta$ such that $X_i \sim f_{\hat{\theta}}$. This could be accomplished by a maximum likelihood estimation, i.e. $\hat{\theta} = \arg\max_\theta \prod_i f_\theta(X_i)$.

Alternatively, when a suitable family cannot be proposed or potential parametric families do not explain the data well, nonparametric or distribution free methods exist that allow the scientist to assume as little structure as possible. Among those nonparametric methods for density estimation are histograms, Kernel density estimation (KDE) and wavelet-based density estimation. A common characteristic of those is that the number of parameters grows with the number of data points; in the worst case keeping all observations as parameters. For comparison, we briefly explain KDE.

Kernel density estimation was first proposed in Rosenblatt (1956) and Parzen (1962). In the univariate case, the kernel density estimator is defined as

$$\widehat{f}_h(x) \doteq \frac{1}{n}\sum_{i=1}^{n} K_h\left(x - x_i\right) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \tag{2.1}$$

where $K$ is usually a symmetric, nonnegative kernel function which satisfies $\int K = 1$, i.e. a probability function itself. The hyperparameter $h$ is the bandwidth and plays a crucial role in controlling overfitting: a too wide bandwidth will oversmooth the estimator whereas a too small value will overfit the sample at hand. See Figure 2.1. Kernel density estimation is probably the most popular method due to its simplicity, it has been well studied, produces smooth densities (when $K$ is smooth) and can easily be generalised to multiple dimensions. Although variations to this exist, note that the basic definition uses the whole sample as parameters. For further details see e.g. Silverman (1986).



(a) Overfitted, $h = 0.05$

(b) Rule Of Thumb bandwidth (Silverman, 1986), $h = 0.233$

(c) Oversmoothed, $h = 0.35$

Figure 2.1. Example KDE estimators of a mixture distribution (dashed blue line), based on a sample of 40 points (red dots) with three different bandwidths and a Gaussian kernel function.

In this work, we use wavelet-based density estimation, which will be introduced shortly.

## 2.2 Wavelet theory

The origins of wavelet theory lie at the intersection of various problems in mathematics, engineering and physics (Jorgensen (2006)). In the early 20th century, Haar (1909) introduced a multi level approach to function approximation using his "system $\chi$", now called Haar wavelet, in Hilbert spaces. Some argue that it took nearly seventy years for a general construction of his system $\chi$ to be rediscovered and formalised, but the truth is that a chain of developments and specific "wavelet examples" were proposed throughout the twentieth century that finally took shape in the wavelet corpus we have today (Meyer (1992); Daubechies (1992)). The Shannon "wavelet" of around 1940 or wavelet "prototypes" in the work of Lusin and Calerón are often cited examples of such intermediate developments (Labate et al. (2013); Meyer (1992)).

However, a turning point occurred around 1976, when Jean Morlet, a French geophysicist, developed wavelets (in French *ondelettes*) to solve the problem of detailed analysis of seismic signals (Morlet (1976)). His work started with the Gabor transform and through connections with some methods in quantum mechanics became what is known today as the continuous wavelet transform (Grossman and Morlet (1984))). It is worth pointing out that they rediscovered the admissibility condition and the reproducing formula of Calderón (1965), a mathematical result on integro-differential operators (Saeki, 1995; Rzeszotnik, 2001). Soon after that, in the late 1980s and early 1990s, Mallat (1989); Meyer (1992); Daubechies (1992), among others, developed the theory of Multiresolution Analysis (MRA), a discretisation of Morlet and Grossman's work that is recursive in the same way as the Haar's system was but more general and suitable to be implemented by computer. Today, the literature on wavelets is quite extensive and their applications are growing (Strang (1989); Mohlenkamp and Pereyra (2008)). For instance, the JPEG 2000 format used in most digital cameras today is based on wavelets (Taubman and Marcellin (2012)) and there are also numerous applications in statistics (e.g. Härdle et al (1998); Jorgensen (2006); Antoniadis (2007)). Below we present the essential wavelet theory concepts and properties that we will use throughout.

### 2.2.1 Multiresolution Approximation

The construction of MRA in $\mathbb{R}^d$ can be introduced in several ways. Here we chose to follow more or less the presentation by Meyer (1992). Other slightly different perspectives are offered in Mallat (1989); Daubechies (1992); Härdle et al (1998); Jorgensen (2006) among many others.

*Definition* 2.2.1. A multiresolution approximation of $L^2(\mathbb{R}^d)$ is an increasing sequence $V_j$, $j \in \mathbb{Z}$, of closed linear subspaces of $L^2(\mathbb{R}^d)$ satisfying[1]:

$$V_{j+1} = \{2^{j\,d/2} f(2^j.) \,:\, f \in V_j\} \text{ for all } j \in \mathbb{Z}, \tag{2.2}$$

$$\bigcap_{j \in \mathbb{Z}} V_j = \{0\}, \tag{2.3}$$

$$\bigcup_{j \in \mathbb{Z}} V_j \text{ is dense in } L^2(\mathbb{R}^d), \text{and} \tag{2.4}$$

$$\text{There is } \varphi \in V_0 \text{ such that } \left\{\varphi(x - .) : z \in \mathbb{Z}^d\right\} \text{ is a orthonormal basis of } V_0. \tag{2.5}$$

The function $\varphi$ is called the scaling function or father wavelet. Let $W_j$ be the orthogonal complement of $V_j$ within $V_{j+1}$, that is $V_{j+1} = V_j \oplus W_j$. The so called mother wavelets are found in $W_j$, but before their introduction, an important concept is in order. We extend MRA with the following:

*Definition* 2.2.2. Let $D^\alpha$ be the $\alpha$-th (multi-index notation) partial weak derivative operator ($D^\alpha = (\partial/\partial x_1)^{\alpha_1} \ldots (\partial/\partial x_d)^{\alpha_d}$). A multiresolution approximation is called $r$-regular for $r \in \mathbb{N}$, if the function $\varphi$ in (2.2.1) can be chosen such that for each $m \in \mathbb{N}$ there is a positive positive constant $C_m$ such that

$$|D^\alpha \varphi(x)| \leq C_m \left(1 + |x|\right)^{-m}, \tag{2.6}$$

for any multi-index $|\alpha| = \alpha_1 + \cdots + \alpha_d \leq r$.

We can now present the structure of the space $W_j$ (Meyer, 1992)

**Theorem 1.** *Let $V_j$ be an $r$-regular MRA of $L^2(\mathbb{R}^d)$ and for $x \in \mathbb{R}^d$ and multi-index $\alpha$, let $x^\alpha = (x_i^{\alpha_i})$ for $i = 1, 2, \ldots, d$. Then there exist $Q = 2^d - 1$ functions $\psi^{(q)}$ such that for every multi-index $\alpha \in N^d$ with $|\alpha| \leq r$, $1 \leq q \leq Q$ and $m \in N$, $m \geq 1$*

$$\text{(a)}\quad \left|D^\alpha \psi^{(q)}(x)\right| \leq C_m \left(1 + |x|\right)^{-m}, \tag{2.7}$$

$$\text{(b)}\quad \left\{\psi^{(q)}(x - z), 1 \leq q \leq Q, z \in \mathbb{Z}^d\right\} \textit{ is an orthonormal basis of } W_0\textit{; and} \tag{2.8}$$

$$\text{(c)}\quad \int x^\alpha \psi^{(q)}(x) dx = 0. \tag{2.9}$$

[1] The reader may note that in Meyer (1992), orthogonality is not part of the MRA but constructed from a Riesz basis. As our main results are developed in the orthonormal setting, we incorporate that in the definition although a more general setting seems possible (see Subsection 2.3.2, Subsection 4.4.3 and Chapter 6).

As these $\psi^{(q)}$ satisfy (c) above, this implies that they have positive and negative parts. Also, by (a), their derivatives vanish as $|x|$ increases. They must have some sort of wave-like shape, hence their name "wavelets". Condition (c) above is sometimes used instead of $r$-regularity and defines the number of *vanishing moments* of the wavelets - essentially they are equivalent in this context. For definiteness, we formalise the concept below.

*Definition* 2.2.3. A wavelet $\psi \in L^2(\mathbb{R}^d)$ is said to have $N$ vanishing moments, if

$$\int x^{\alpha}\psi(x)\,\mathrm{d}x = 0, \text{ for } |\alpha| \leq N. \tag{2.10}$$

Now, define $\varphi_{j,z}(x) = 2^{j\,d/2}\varphi(2^jx - z)$ and $\psi^{(q)}_{j,z}(x) = 2^{j\,d/2}\psi^{(q)}(2^jx - z)$. As $\overline{\bigcup_{j\in\mathbb{Z}} V_j} = L^2(\mathbb{R}^d)$, the direct sum $\bigoplus_{j\in\mathbb{Z}} W_j$ $\underline{\text{is}}$ $L^2(\mathbb{R}^d)$; this means, in principle, that the wavelets $\psi^{(q)}_{j,z}$ can be used to represent any $f \in L^2(\mathbb{R}^d)$ across all levels of refinement. However, this $W_j$-only representation does not work as it implies an unattainable full coverage towards zero in the frequency domain, requiring a "cork", a low pass filter to cover that region (Valens, 1999)[2]. This low pass filter is provided by a starting $V_0$ representation, leading to $L^2(\mathbb{R}^d) = \overline{V_0 \oplus \left(\bigoplus_{j\in N} W_j\right)}$. Actually, the initial scale $j = 0$ is arbitrary, so one can start at any scale $J_0 \in \mathbb{Z}$. This can be formalised in the following.

Let the coordinate projections of $f$ into $V_j$ and $W_j$ be

$$\alpha_{j,z} \doteq < f, \varphi_{j,z} > \tag{2.11}$$

$$\beta^{(q)}_{j,z} \doteq < f, \psi^{(q)}_{j,z} > . \tag{2.12}$$

Then, the above direct sum for $L^2(\mathbb{R}^d)$ means that for any $f \in L^2(\mathbb{R}^d)$ we have $(Q_d = \{1, 2, \ldots, 2^d - 1\})$

$$f(x) = \sum_{z\in\mathbb{Z}^d} \alpha_{J_0,z}\varphi_{J_0,z}(x) + \sum_{j=J_0}^{\infty}\sum_{z\in\mathbb{Z}^d}\sum_{q\in Q_d} \beta^{(q)}_{j,z}\psi^{(q)}_{j,z}(x), \tag{2.13}$$

[2] An alternative argument in Meyer (1992), pg. 67, is that the extension of MRA to more general $L^p\left(\mathbb{R}^d\right)$ spaces will trigger the need for an initial set of basis functions that are not "waves", i.e. that do not integrate to 0 but to 1.

and the projection of $f$ in $V_J$ is

$$f_J(x) = \sum_{z \in \mathbb{Z}^d} \alpha_{J,z} \varphi_{J,z}(x). \tag{2.14}$$

Now, as $\varphi$ and $\psi^{(q)}$ are in $V_1$ ($V_1 = V_0 \oplus W_0$), they can be expressed in its basis as

$$\varphi(x) = \sum_{z \in \mathbb{Z}^d} H_z \varphi(2x - z) \tag{2.15}$$

$$\psi^{(q)}(x) = \sum_{z \in \mathbb{Z}^d} G_z^{(q)} \varphi(2x - z), \tag{2.16}$$

where $H_z$ and $G_z^{(q)}$ are the coordinates of $\varphi$ and $\psi^{(q)}$ respectively. If $\varphi$ has compact support, so do the $\psi^{(q)}$ and there is only a finite number of terms in the sums (2.15, 2.16). In the one-dimensional case, $H_z$ and $G_z$ are known in the signal processing community as filter banks and are central to numerical calculations on $l^2(\mathbb{R})$ sequences. In fact, Daubechies (1988) elaborates in great detail how conditions imposed on a MRA construction are equivalent to certain conditions on $H_z$ and $G_z$, finalising with her celebrated breakthrough on the construction of compactly supported orthogonal wavelets in $L^2(\mathbb{R})$ of arbitrary number of vanishing moments which we will summarise in Subsection 2.2.3.

The term $\sum_{z \in \mathbb{Z}} \alpha_{J_0,z} \varphi_{J_0,z}(x)$ is called the 'trend' at level $J_0$, while, for each level $j \geq J_0$, $\sum_{z \in \mathbb{Z}} \sum_{q \in Q_d} \beta_{j,z}^{(q)} \psi_{j,z}^{(q)}(x)$ is the 'detail' at level $j$. A key feature of a multiresolution representation such as (2.13) is that, for any $j \geq J_0$, the trend at level $j+1$ coincides with the trend at level $j$ supplemented with the detail at level $j$. Specifically,

$$\sum_{z \in \mathbb{Z}^d} \alpha_{j+1,z} \varphi_{j+1,z}(x) = \sum_{z \in \mathbb{Z}^d} \alpha_{j,z} \varphi_{j,z}(x) + \sum_{z \in \mathbb{Z}^d} \sum_{q \in Q_d} \beta_{j,z}^{(q)} \psi_{j,z}^{(q)}(x). \tag{2.17}$$

This implies that the projection onto $V_{J_1+1}$ as in (2.14) is equivalent to this truncated wavelet expansion

$$f_{J_0,J_1}(x) = \sum_{z \in \mathbb{Z}^d} \alpha_{J_0,z} \varphi_{0,z}(x) + \sum_{j=J_0}^{J_1} \sum_{z \in \mathbb{Z}^d} \sum_{q \in Q_d} \beta_{j,z}^{(q)} \psi_{j,z}^{(q)}(x), \tag{2.18}$$

which is what one uses in practice. Of course, truncation will make sense only if this projection can be made close to $f$ in some objective sense. This is presented next.

### 2.2.2 Wavelet approximations in Sobolev spaces

From the father wavelet $\varphi$, let the *approximating kernel* $K : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ be

$$K(x,y) = \sum_{z \in \mathbb{Z}^d} \varphi(x-z)\varphi(y-z) \tag{2.19}$$

and its *refinement* at resolution $j \in \mathbb{N}$ be

$$K_j(x,y) = \sum_{z \in \mathbb{Z}^d} 2^{dj}\varphi(2^j x - z)\varphi(2^j y - z) = \sum_{z \in \mathbb{Z}^d} \varphi_{j,z}(x)\varphi_{j,z}(y). \tag{2.20}$$

Define the two associated operators:

$$Kf(x) = \int_{\mathbb{R}^d} K(x,y)f(y)\,\mathrm{d}y$$

and

$$K_j f(x) = \int_{\mathbb{R}^d} K_j(x,y)f(y)\,\mathrm{d}y,$$

for all functions $f \in L_2(\mathbb{R}^d)$. We will see that $K_j$ is an approximate identity but before that a remarkable result

**Theorem 2.** *(Meyer (1992), Theorem 4 (Ch. 2), Corollary p. 38) Let $V_j$ be an $r$-regular MRA of $L^2(\mathbb{R}^d)$ and let $K_j : L^2(\mathbb{R}^d) \to V_j$ be the orthogonal projection defined above. Then for any polynomial $P$ of degree less than or equal to $r$, $K_j(P) = P$.*

Thus, in a $r$-regular MRA, polynomials of degree less than or equal to $r$ are kept unchanged by the projection operator $K_j$, which sheds light on the nature of the spaces $V_j$ associated with such MRA.

Finally, to define the approximation power of wavelets and MRA, we use Sobolev spaces, which are defined as follows.

*Definition* 2.2.4. A space of functions defined on $\Omega \subset \mathbb{R}^d$ for which all mixed partial derivatives up to order $m \geq 0$ exist (in the weak sense) and that belong to $L_p(\Omega)$,

$1 \leq p \leq \infty$ is called a Sobolev space. Formally,

$$W^{m,p}(\Omega) = \left\{\phi \in L^p(\Omega) : D^\alpha \phi \in L^p(\Omega) \ \forall \alpha \in \mathbb{N}^d : |\alpha| \leqslant m\right\},$$

where $D^\alpha$ is the $\alpha$th (multi-index notation) partial weak derivative operator, and $|\alpha| = \alpha_1 + \cdots + \alpha_d$. A norm on $W^{m,p}(\Omega)$ is classically defined as $\|\phi\|_{m,p} = \sum_{|\alpha| \leq m} \|D^\alpha \phi\|_p$ (Triebel, 1992).

Now, suppose that the father wavelet $\varphi$ introduced in Assumption 3.2.2 is such that the induced kernel (2.19) satisfies the following assumption.

**Assumption 2.2.1.** The kernel $K$ (2.19) is such that $|K(x,y)| \leq F(x-y)$, for some square integrable function $F : \mathbb{R}^d \to \mathbb{R}$ with $\int_{\mathbb{R}^d} |x|^\nu F(x)\,\mathrm{d}x < \infty$ for all $\nu \in \mathbb{N}^d$ such that $|\nu| = m$. Moreover, for all $x \in \mathbb{R}^d$, $\int_{\mathbb{R}^d} (y-x)^{\nu'} K(x,y)\,\mathrm{d}y = \delta_{0,\nu'}$, for all $\nu' \in \mathbb{N}^d$ such that $|\nu'| \leq m-1$.

**Theorem 3.** *(Härdle et al (1998), Theorem 8.1(ii)) Under assumptions above and if $f \in W^{m,p}$, then $\|K_j f - f\|_p \leq C 2^{-j}$.*

For a thorough presentation of above see Daubechies (1992); Meyer (1992); Härdle et al (1998).

Now, we focus our attention to one important aspect we haven't addressed in detail. As mentioned above, the existence and construction of wavelets of compact support will enable the implementation of algorithms that do not need to approximate or compute, somehow, infinite sums. We cover this next.

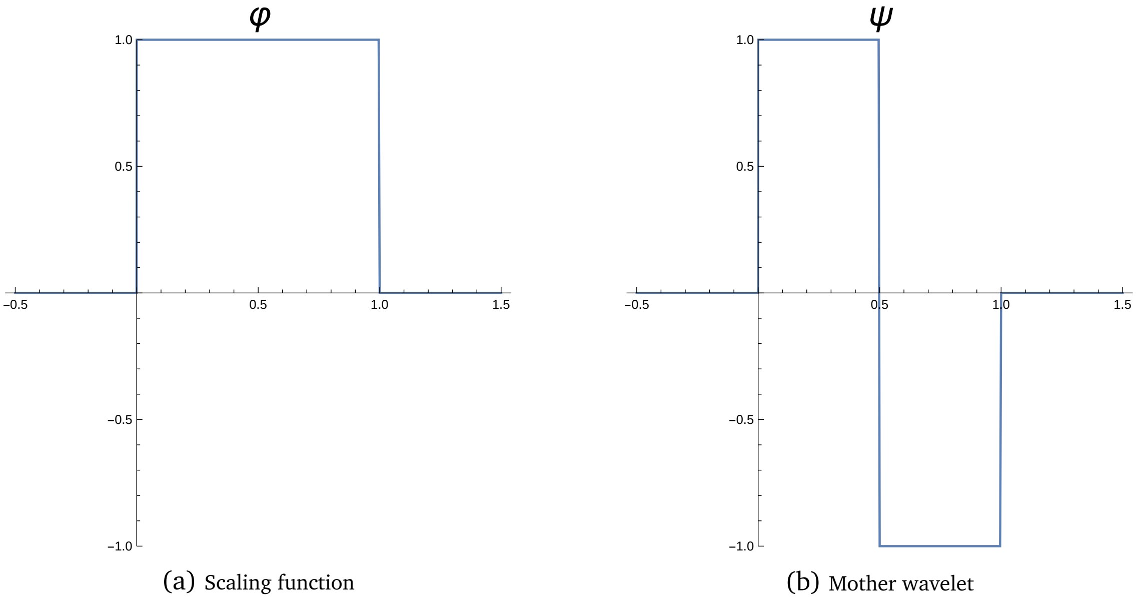
### 2.2.3 Wavelets of compact support

The Haar system is usually cited as the basic example of a MRA in $L^2(\mathbb{R})$. The scaling and wavelet functions are defined by

$$\varphi(x) = \begin{cases} 1, & 0 \leqq x < 1 \\ 0, & \textit{otherwise;} \end{cases} \tag{2.21}$$

$$\psi(x) = \begin{cases} 1, & 0 \leqq x < 1/2 \\ -1, & 1/2 \leqq x < 1 \\ 0, & \textit{otherwise.} \end{cases} \tag{2.22}$$

The corresponding plots are as depicted in Figure 2.2.



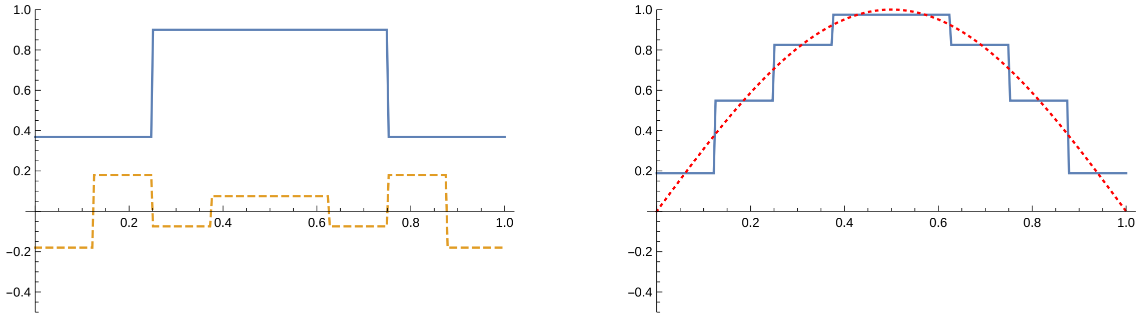
(a) Scaling function

(b) Mother wavelet

Figure 2.2. Haar system

Here an example in the smaller space $L^2([0,1])$. The projections of $f(x) = \sin(\pi x)$ into $V_2$ and $W_2$ are plotted in Figure 2.3 (a). The sum between these two functions lets to a new refinement in $V_3 = V_2 \oplus W_2$ plotted in (b). For comparison, the true $f$ is also plotted in (b) as a dotted red line.

The Haar system has compact support but its regularity is $r = 0$, i.e. it is able to represent perfectly only functions that are constant on the dyadic intervals (see Theorem 2), and a valid question is whether or not it is possible to construct wavelet basis of regularity $r$ for any $r \geq 1$ that has compact support. The answer



(a) Projection into $V_2$ and $W_2$

(b) Projection into $V_3$ and true $f$ (dotted red line)

Figure 2.3. Haar MRA in $L^2([0,1])$ for $sin(\pi x)$

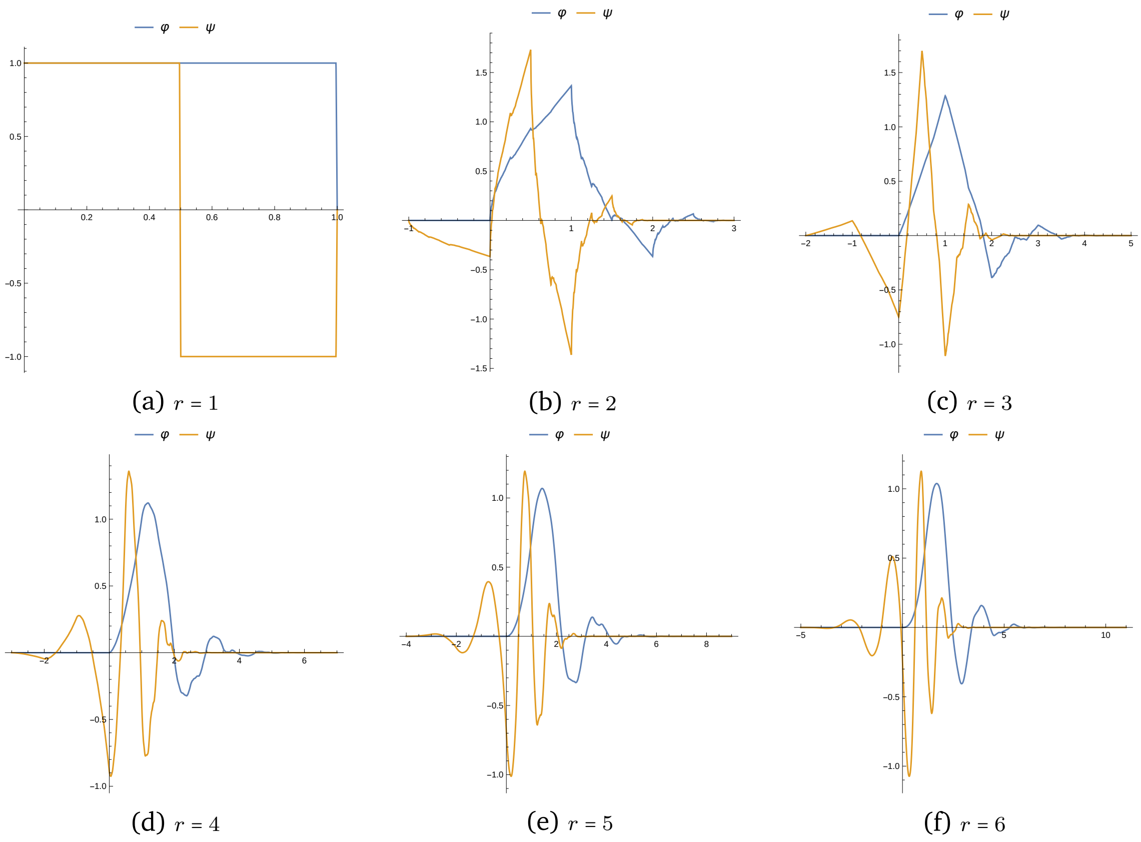
to this was provided in the following result by I. Daubechies

**Theorem 4.** *For each integer $r \geq 1$, there exists a MRA $V_j$ of $L^2(\mathbb{R})$ which is $r$-regular and such that the associated functions $\varphi$ and $\psi$ have compact support.*

The presentation above is from Meyer (1992), as Daubechies (1988) has a more detailed theorem based on the "graphical" algorithm of such systems. The end result in the later is the introduction of a family of wavelets now known as Daubechies wavelets. After elaborating all the constraints associated with the coefficients $H_z$ in (2.15) in order for them to become a MRA[3], the solutions with "minimal phase" (and smallest support) for a given number of vanishing moments are chosen - these are the Daubechies' wavelets. Some of these Daubechies' wavelets are plotted in Figure 2.4. Note that the plots are not at the same scale: the support of Daubechies' $\varphi$ of order $r$ is $[0, 2r - 1]$ and the support of $\psi$ is $[-r + 1, r]$. With the increase in regularity, one pays the price of a longer support.

An interesting result by Daubechies, (Daubechies, 1992, Th. 8.1.4), demonstrates that there are no symmetric, real-valued, orthogonal wavelets of minimal compact support for a given number of vanishing moments. However, by relaxing restrictions on the size of the support, it is possible to construct orthonormal wavelets with other properties. For instance, a higher degree of symmetry may be desired in certain signal processing applications, like image analysis. Or for compression (sparsity), having vanishing moments in the scaling function in addition to those of the mother wavelet, i.e. having $\int x^l \varphi(x)\,\mathrm{d}x = 0$ for $l = 1, \ldots, L - 1$, leads to better compression ratios. I. Daubechies also built wavelets with these characteristics

[3] $G_z^{(q)}$ in 2.16 are derived directly from $H_z$ in the univariate case



(a) $r = 1$ (b) $r = 2$ (c) $r = 3$

(d) $r = 4$ (e) $r = 5$ (f) $r = 6$

Figure 2.4. The Daubechies wavelet family for different regularities. Plots not drawn at the same scale. The plots for $r = 1$ are Haar's wavelets, better seen in Figure 2.2.

and they are known as *symlets* and *coiflets* respectively. The reader is encouraged to visit Daubechies (1992) for more details.

## 2.2.4 Wavelet extensions

### 2.2.4.1 Riesz frames

If one is willing to sacrifice orthogonality, it is possible to extend MRA beyond the orthonormal case by using frames (Daubechies, 1992). In fact, frames have provided the foundation for the variety of wavelet extensions that came after the breakthroughs in the field in the late 80's. These include so called second generation wavelets (Sweldens, 1996) and many multivariate anisotropic extensions in current research (Grohs et al., 2013). Precursors to frames were known already in the signal processing community as quadrature mirror filters. In analysis, frames, introduced by Duffin and Schaeffer (1952), can be regarded as the most natural generalization of the notion of an orthonormal basis (Casazza and Kutyniok (2012)).

*Definition* 2.2.5. A family of functions $(\varphi_\lambda)_{\lambda\in\Lambda}$ in a Hilbert space $\mathcal{H}$ is a frame if there exist $0 < A, B < \infty$ so that for all $f \in \mathcal{H}$,

$$A\left\|f\right\|^2 \leq \sum_{\lambda\in\Lambda} |< f, \varphi_\lambda >| \leq B\left\|f\right\|^2.$$

If $A = B$, it is called a tight frame, with the case $A = B = 1$ corresponding to an orthogonal basis. In essence, the upper bound establishes boundedness of the $(< f, \varphi_\lambda >)_{\lambda\in\Lambda}$ sequence whereas lower bound ensures $0 \in \mathcal{H}$ can only be represented by the zero sequence $(0)_{\lambda\in\Lambda}$. In fact, it can be shown that if $(\varphi_\lambda)_{\lambda\in\lambda}$ is a frame, there is another family $(\tilde{\varphi}_\lambda)_{\lambda\in\Lambda}$ which is also a frame and that the two lead to the following reconstruction formula, an extension to (2.11), (2.12) and (2.13)[4],

$$\alpha_\lambda \doteq < f, \tilde{\varphi}_\lambda >, \tag{2.23}$$

$$f(x) = \sum_{\lambda\in\Lambda} \alpha_\lambda \varphi_\lambda(x). \tag{2.24}$$

Because of this pair of dual frames, the construction is usually called biorthogonal (Daubechies, 1992). In signal processing, calculating the sequence $\alpha_\lambda$ corresponds to the *analysis* phase and computing the sum for the reconstruction the *synthesis* phase, with the coefficients themselves for $\varphi$ and $\tilde{\varphi}$ called quadrature mirror filters.

Biorthogonal wavelets thus offer more flexibility and include as particular cases the spline wavelets. In our work, we will focus on orthogonal wavelets but some results can be extended easily to the biorthogonal case and we will offer some numerical examples of this.

#### 2.2.4.2 Further reading

Despite being a relatively young area of mathematics, the literature of wavelets is quite vast with nearly half a million entries reported by a Google's scholar search as of 2019. The reader interested in deepening their knowledge can, in addition to the classical books by Daubechies (1992), Meyer (1992) and Mallat (1999), look at introductory sources, reference and application books like Strang (1989);

[4] Let $\mathbb{Z}_j = \{i \in \mathbb{Z} : i \geq j\}$, then make $\Lambda = \mathbb{Z}^d \oplus \left(\mathbb{Z}_{J_0} \times Q \times \mathbb{Z}^d\right)$ to map indexes for alphas and betas.

Härdle et al (1998); Valens (1999); Weiss and Wilson (2001); Jorgensen (2006); Vidakovic (2009); Starck et al. (2010); Labate et al. (2013). For literature references regarding general construction of wavelets and Riesz frames see Cohen and Daubechies (1992); Sweldens (1996); Casazza and Kutyniok (2012); Grohs et al. (2013).

## 2.3 Density estimation with wavelets

Density estimation is a central problem in statistics and within nonparametric theory it is perhaps one of the most investigated topics (Silverman, 1986). The mathematical theory of wavelets presented above offers a powerful tool for approximating possibly irregular functions or surfaces and, in statistics, it provides a convenient framework for some nonparametric problems, in particular density estimation.

> Before we even heard of wavelets, we had realized that if only there existed some very special functions, for which we had drawn up a wish list, then we could tackle a whole variety of statistical problems that had not been successfully tamed before. And then wavelets came along, and they did all we had hoped for.
>
> (D. Donoho, as quoted by (Daubechies, 1993))

Indeed, some variants of wavelet estimators are (near-) optimal in some sense over large classes of functions (Kerkyacharian and Picard, 1993; Donoho et al., 1996; Donoho et al, 1995; Donoho and Johnstone, 1994, 1995, 1996, 1998; Fan et al., 1996). Härdle et al (1998); Vidakovic (2009) and Nason (2010) give comprehensive reviews of wavelet methods applied to statistics. Below, we present their application to density estimation.

### 2.3.1 Linear wavelet-based density estimation

Let $f$ in (2.13) be a PDF. Noting that $\alpha_{j_0,z} = \mathbb{E}\{\varphi_{j_0,z}(X)\}$ and $\beta_{j,z}^{(q)} = \mathbb{E}\{\psi_{j,z}^{(q)}(X)\}$ paves the way for their estimation, upon observing a sample from $f$, by empirical averages, say $\hat{\alpha}_{j_0,z}$ and $\hat{\beta}_{j,z}^{(q)}$. In addition, for any practical purpose the infinite expansion (2.13) needs to be truncated after a finite number of terms, say $J \geq j_0$ – in the wavelet jargon, one says that $f$ is approximated to the resolution level $J$. So, a wavelet estimator for $f$ writes

$$\hat{f}_J(x) = \sum_{z \in \mathbb{Z}^d} \hat{\alpha}_{j_0,z}\varphi_{j_0,z}(x) + \sum_{j=j_0}^{J} \sum_{z \in \mathbb{Z}^d} \sum_{q \in Q_d} \hat{\beta}_{j,z}^{(q)} \psi_{j,z}^{(q)}(x), \tag{2.25}$$

which may ultimately include some thresholding of the estimated coefficients. Note that the sums over $z$ are finite if the wavelets have compact support, as it is usually assumed.

Putting together the work on approximating kernels, wavelets (Theorem 3) and the above estimator (2.25), we have

**Theorem 5** (Härdle et al (1998), Th. 10.1)**.** *Let $K$ be a kernel such that assumption 2.2.1 holds, and if $f$ belongs to $W^{m,2}$ then the Mean Integrated Squared Error (MISE) is uniformly bounded in balls $B(L) = \{f : \|f\|_{W^{m,2}} < L$ and $f$ a probability density$\}$,*

$$\sup_{f \in B(L)} \mathbb{E}\left[\left\|\hat{f}_J - f\right\|^2\right] \leq C_1 \frac{2^J}{n} + C_2 2^{-2Jm}, \tag{2.26}$$

*where $C_1$ and $C_2$ as positive constants.*

*When the two antagonistic quantities on the RHS are balanced, this is $2^{J(n)} \propto n^{\frac{1}{2m+1}}$, we obtain*

$$\sup_{f \in B^{m,2}(L)} \mathbb{E}\left[\left\|\hat{f}_J - f\right\|^2\right] \leq C n^{-\frac{2m}{2m+1}}, \tag{2.27}$$

*for some $C > 0$.*

### 2.3.2 Wavelet thresholding and shrinkage

The advantage of wavelet-based density estimators lies in their ability to capture local phenomena by "even simple non-linearities" involving coordinate thresholding (Donoho et al. (1996)). Thresholding involves the suppression and/or shrinkage of the estimated beta coefficients by some sort of rules. Three main approaches have been proposed: hard and soft thresholding (Donoho et al. (1996); Delyon and Juditsky (1996)), where coefficients are selected based on their magnitude being greater than a given threshold and then they are kept as-is (hard) or shrunk towards zero (soft) linearly; and block thresholding, where coefficients are selected as a block of neighbouring translations (Hall et al. (1997)) and kept as-is.

The term linear estimator refers to the fact that the estimator (2.18), via the estimated coefficients (2.11) and (2.12), is a linear function of the empirical measure

$$\nu_n = \sum_{i=1}^{n} \delta_{X_i}, \tag{2.28}$$

$\delta_x$ the Dirac mass at point $x$ (Härdle et al (1998)). This is, if $f, g$ are two PDFs, $\alpha \in [0,1]$ and $\hat{f}_L$ is a linear estimator, then $E_{\alpha f+(1-\alpha)g}\hat{f}_L = \alpha E_f \hat{f}_L + (1-\alpha)E_g \hat{f}_L$ (Donoho et al. (1996)). Of course, thresholding in its various forms breaks this identity.

We will see later that the estimator presented in this work is not, strictly speaking, linear in the above sense. However, in keeping with standard terminology, we will refer to our full estimator as the linear version and the estimator with selected coefficients by a criterion similar to above as the non-linear case.

For future reference, we present the traditional wavelet estimator and the thresholding method applied to (2.18). First, define empirical coefficients $\hat{\alpha}_{j,z}$ and $\hat{\beta}^{(q)}_{j,z}$ for (2.11) and (2.12) respectively as empirical averages[5]

$$\hat{\alpha}_{j,z} = \frac{1}{n}\sum_{i=1}^{n} \varphi_{j,z}(X_i) \tag{2.29}$$

$$\hat{\beta}^{(q)}_{j,z} = \frac{1}{n}\sum_{i=1}^{n} \psi^{(q)}_{j,z}(X_i). \tag{2.30}$$

Hard thresholding of betas is defined by

$$\tilde{\beta}^{(q)}_{jz} \doteq \begin{cases} \hat{\beta}^{(q)}_{jz}, & \text{if } \left|\hat{\beta}^{(q)}_{jz}\right| > K\,C(j)n^{-1/2} \\ 0, & \textit{otherwise} \end{cases} \tag{2.31}$$

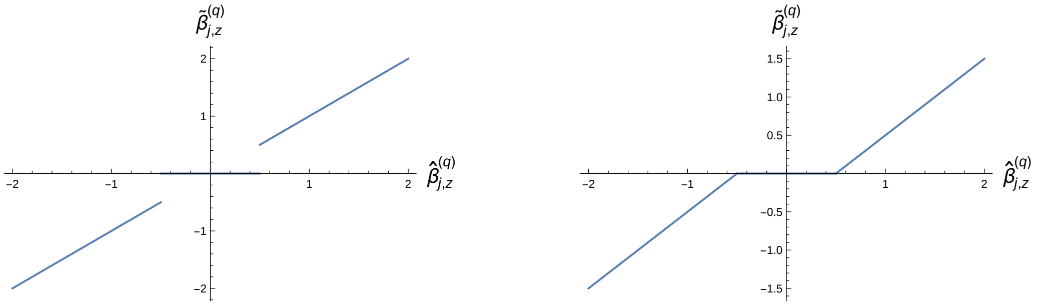
where $C(j)$ is a resolution-varying function and $K$ is a positive constant to be determined. In Donoho et al. (1996), $C(j) = \sqrt{j}$ whereas in Delyon and Juditsky (1996) $C(j) = \sqrt{j - J_0}$[6]. Finally, the hard thresholded estimator is then defined by

[5] Naturally $\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\varphi_{j,z}(X_i)\right] = < f, \varphi_{j,z} > = \alpha_{j,z}$ for i.i.d $X_i$, and similarly for $\beta^{(q)}_{j,z}$.

[6] Note Donoho et al. (1996) and Delyon and Juditsky (1996) use different meaning for $j$, $J_0$

$$\hat{f}_{J_0,J_1}(x) = \sum_{z\in\mathbb{Z}^d} \hat{\alpha}_{J_0,z}\varphi_{J_0,z}(x) + \sum_{j=J_0}^{J_1}\sum_{z\in\mathbb{Z}^d}\sum_{q\in Q_d}\tilde{\beta}^{(q)}_{j,z}\psi^{(q)}_{j,z}(x).. \qquad (2.32)$$

The soft threshold is a slight variation. Define the threshold for level $j$ as $\tau_j = K\,C(j)n^{-1/2}$, then $\tilde{\beta}^{(q)}_{j,z} \doteq \mathrm{sign}(\hat{\beta}^{(q)}_{jz})(\hat{\beta}^{(q)}_{jz} - \tau_j)_+$. The difference between the hard and soft approaches over a coefficient is depicted in Figure 2.5. The shape of soft thresholding justifies the name "wavelet shrinkage" given to its application (Donoho and Johnstone, 1995, 1998).



(a) Hard threshold

(b) Soft threshold

Figure 2.5. Effect of the different thresholding approaches for a threshold of $\tau = 0.5$ over a given beta coefficient.

Note that because of the characterisation of Sobolev spaces in terms of wavelet coefficients (Meyer (1992), Theorem 2.8), a bounded $\|\phi\|_{m,p} < \infty$ will "force" most of the $\beta^{(q)}_{j,z}$ to be small, making the choice $\tilde{\beta}^{(q)}_{j,z} = 0$ natural for this case; as opposed to shrinking the alpha coefficients, which are left intact and correspond to a purely linear estimator (Donoho and Johnstone, 1996).

Finally, another practical aspect of (2.32) is defining $J_0$ and $J_1$. In Donoho et al. (1996), the resolution is found to be $J_0(n) \propto \log(n/\log(n))$, where the proportionality constant depends on the parameters of the Sobolev or Besov space under consideration. One way to avoid pin pointing these is to choose $J_0$ and $J_1$ empirically, taking into account of the fact that the resolution is a smoothing parameter. In fact, the primary resolution level plays the role of bandwidth in the *linear* part of a thresholded wavelet estimator (the alphas), and so correct choice of this quantity can alleviate difficulties caused by over- or under-smoothing (Hall and Penev (2001)). In this work, we will present a way to determine these resolutions on a

and $J_1$. Ours similar to the later as it is more commonly used.

data-driven manner.

## 2.4 Shape-preserving density estimators

One major drawback, though, of the above wavelet-based estimators is that they are in general not 'shape-preserving'. When estimating a PDF $f$, that means that the resulting estimator $\hat{f}_J$ may neither be non-negative, nor integrate to one (Dechevsky and Penev, 1997, 1998). Usually, simple rescaling solves the integrability issue, but overcoming the non-negativity issue requires caution. Two approaches are possible (Vannucci and Vidakovic, 1997): one may truncate the estimate to its positive part and then re-normalize or one may estimate a transformed version of $f$, usually $\sqrt{f}$ or $\log(f)$, and then transform back to have a nonnegative estimate. The transformation $\sqrt{f}$ was introduced by Good and Gaskins (1971) and developed for the wavelet case by Pinheiro and Vidakovic (1997), whereas Leonard (1973); Silverman (1982); Gu and Qiu (1993) are credited with the introduction and study of estimation based on the $\log(f)$ transform, further developed for the wavelet case by Koo and Kim (1996).

Here, we focus on the square root transform. To estimate $g \doteq \sqrt{f}$, consider the univariate case. Clearly, $g \in L_2(\mathbb{R})$, as $\int_{\mathbb{R}} g^2(x)\,\mathrm{d}x = \int_{\mathbb{R}} f(x)\,\mathrm{d}x = 1$, hence we can write its expansion (2.13), viz.

$$g(x) = \sum_{z\in\mathbb{Z}} \alpha_{j_0,z}\varphi_{j_0,z}(x) + \sum_{j=j_0}^{\infty}\sum_{z\in\mathbb{Z}} \beta_{j,z}\psi_{j,z}(x), \tag{2.33}$$

where

$$\alpha_{j,z} = \int_{\mathbb{R}} \varphi_{j,z}(x) g(x)\,\mathrm{d}x = \int_{\mathbb{R}} \varphi_{j,z}(x)\sqrt{f}(x)\,\mathrm{d}x \tag{2.34}$$

$$\beta_{j,z} = \int_{\mathbb{R}} \psi_{j,z}(x) g(x)\,\mathrm{d}x = \int_{\mathbb{R}} \psi_{j,z}(x)\sqrt{f}(x)\,\mathrm{d}x. \tag{2.35}$$

Difficulty in estimating these coefficients arises as $\alpha_{j,z} = \mathbb{E}\{\varphi_{j,z}(X)/\sqrt{f}(X)\}$ and $\beta_{j,z} = \mathbb{E}\{\psi_{j,z}(X)/\sqrt{f}(X)\}$ can no longer be estimated directly by sample averages.

Pinheiro and Vidakovic (1997) got around the presence of the unknown factor

$1/\sqrt{f}$ in these expectations by plugging in a pilot estimator of $f$. This is,

$$\hat{\alpha}_{j,k} = \frac{1}{n} \sum_{i=1}^{n} \frac{\psi_{j,k}(X_i)}{\sqrt{\tilde{f}_n(X_i)}} \tag{2.36}$$

$$\hat{\beta}_{j,k} = \frac{1}{n} \sum_{i=1}^{n} \frac{\phi_{j,k}(X_i)}{\sqrt{\tilde{f}_n(X_i)}}, \tag{2.37}$$

where $\tilde{f}_n$ is a pilot estimator of $f$ that is computationally simple and gives sensible "weights" to $\varphi_{j,k}$'s and $\psi_{j,k}$'s. Their choice of a counting pilot

$$\tilde{f}_n(X_i) = \#\{X_j \in (X_i - r, X_i + r)\} \tag{2.38}$$

with radius $r \in \mathbb{R}^+$, seemed to produce good results and can be generalised in a direct manner to the multivariate case (Vannucci, 1995). They, however, did not pursue a detailed theoretical analysis of that estimator.

Rather, Penev and Dechevsky (1997) suggested a more elegant construction based on order statistics and spacings, e.g. a direct estimator for $\beta_{j,z}$ is

$$\frac{2}{\sqrt{\pi}} \cdot \frac{1}{\sqrt{n}} \left[ \sum_{m=1}^{n-1} \psi_{j,k} X_{(m+1)} \sqrt{X_{(m+1)} - X_{(m)}} \right] \tag{2.39}$$

where $X_{(m)}$ denote order statistics of the sample and $\sqrt{X_{(m+1)} - X_{(m)}}$ is the square root of the spacing between consecutive observations. Unfortunately, direct application of their idea is limited to the univariate case, as spacings are not defined in more than one dimension. Yet, the need for a multivariate extension of the 'Dechevsky–Penev' construction was explicitly called for by McFadden (2003) in his Nobel Prize lecture.

Cosma et al. (2007) and Peter and Rangarajan (2008) attempted such extension but lost much of the initial flavour of the idea. In fact, Cosma et al. (2007) gave up the idea of estimating $\sqrt{f}$ and enforced the non-negativity constraint by resorting to a non-negative 'father wavelet', making their construction very close in spirit to a spline-like estimator. This angle of attack has some advantages, e.g., it allows for dependent observations, and it remains valid under regularity conditions on $f$ milder than the usual belonging to some Besov space. However, it does not easily lend itself to the introduction of thresholding, thus it is unable to consummate the full potential of wavelet-based approaches.

In contrast, Peter and Rangarajan (2008) started from the expansion (2.33) but estimated the coefficients by Maximum Likelihood, which requires solving a high-dimensional optimisation problem and careful numerical treatment. Unfortunately, empirical evaluation of their approach seemed to obtain best results using a single-level expansion, i.e. alphas only, which leads to question their approach in light of known optimal theoretical results of nonlinear estimators. More recently, Peter et al. (2017) elaborated further by bringing Bayesian model selection into their approach, pointing out interesting links to Riemannian geometry, the hypersphere and the approach of Minimum description length (MDL). Central to their approach is the use of the Hellinger Distance (HD), from which we will profit too. Also, though rich in practical and simulation experiments, there is no further theoretical analysis of the depth that we present in here.

Finally, we can also mention Brown et al. (2010), which involves wavelet methods for estimating $\sqrt{f}$. However, those authors transform through binning the density estimation problem into a Poisson regression one, for which considering the square-root of $f$ is justified explicitly for its variance-stabilising effect. Their framework is thus very different to what is investigated here. For instance, it relies on a rule-of-thumb estimator for the initial resolution level, $J_n = \lfloor \log_2 n^{3/4} \rfloor$, and corresponding bin widths $w = 1/T$, where $T = 2^{J_n}$, effectively taking the form of $w \simeq \frac{1}{n^{3/4}}$ for a density defined in $[0,1]$. For the interested reader, it might be useful to contrast this approach with a preceding example in the Digital Signal Processing (DSP) literature of a wavelet-based estimation of the square-root of a density in they call the "discrete" case of histograms, Yoon and Vaidyanathan (2004). In contrast to binning, we will present a first principles, completely data-driven algorithm to calculate a reference initial resolution in a multivariate setting.

# Chapter 3

# A shape-preserving multivariate density estimator using wavelets

## 3.1 Definition of the estimator

### 3.1.1 Motivation

Let $\mathcal{X} = \{X_1, \ldots, X_n\}$ be a random sample from an unknown $d$-dimensional distribution $F$ admitting a density $f$ on $\mathbb{R}^d$. Denote by $X_{(k);i}$ the $k$th closest observation from $X_i$ among the other points of $\mathcal{X}$. Define $R_{(k);i} = \|X_{(k);i} - X_i\|$ the Euclidean distance between $X_i$ and $X_{(k);i}$, and

$$V_{(k);i} = c_0 R^d_{(k);i}, \qquad \text{where } c_0 = \frac{\pi^{d/2}}{\Gamma(d/2+1)}, \tag{3.1}$$

the volume of the ball of radius $R_{(k);i}$ centred at $X_i$ – hence it is the smallest ball centred at $X_i$ containing at least $k$ other observations from $\mathcal{X}$. It is known (Ranneby et al, 2005, Proposition 2) that, conditionally on $X_i$, and as $n \to \infty$,

$$nV_{(1);i} \rightsquigarrow \text{Exp}\{f(X_i)\}, \tag{3.2}$$

meaning that (Johnson et al., 1994, Section 10.5), as $n \to \infty$,

$$\sqrt{nV_{(1);i}} \xrightarrow{\mathcal{L}} \text{Rayleigh}\left[\{2f(X_i)\}^{-1/2}\right]. \tag{3.3}$$

Now, consider an arbitrary square-integrable function $\phi : \mathbb{R}^d \to \mathbb{R}$, and define

$$S_n = \frac{2}{\sqrt{\pi}} \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \phi(X_i) \sqrt{V_{(1);i}}. \tag{3.4}$$

By the Law of Iterated Expectations, we have

$$\mathbb{E}(S_n) = \mathbb{E}\left\{ \frac{2}{\sqrt{\pi}} \phi(X_i) \mathbb{E}\left( \sqrt{nV_{(1);i}} \, | X_i \right) \right\}.$$

The expectation of a Rayleigh($\sigma$)-random variable is known to be $\sigma\sqrt{\pi/2}$. If the convergence in law (3.3) implies the convergence of the moments (this is indeed the case here as will be formally derived later), then

$$\begin{aligned} \mathbb{E}(S_n) &\to \mathbb{E}\left\{ \frac{2}{\sqrt{\pi}} \phi(X_i) \frac{\sqrt{\pi}}{2\sqrt{f}(X_i)} \right\} \\ &= \int_{\mathbb{R}^d} \frac{\phi(x)}{\sqrt{f}(x)} f(x) \, \mathrm{d}x \\ &= \int_{\mathbb{R}^d} \phi(x) \sqrt{f}(x) \, \mathrm{d}x. \end{aligned}$$

Hence, $S_n$ is an asymptotically unbiased estimator of $\int_{\mathbb{R}^d} \phi(x)\sqrt{f}(x) \, \mathrm{d}x$. This fact naturally suggests estimating the wavelet coefficients (2.35) by statistics of type (3.4), which is the idea formally investigated in here.

### 3.1.2 Definition

Let $g = \sqrt{f}$, where $f$ is the $d$-dimensional density to estimate. As $g \in L_2(\mathbb{R}^d)$ always, we have, by (2.13),

$$g(x) = \sum_{z \in \mathbb{Z}^d} \alpha_{J_0,z} \varphi_{J_0,z}(x) + \sum_{j=J_0}^{\infty} \sum_{z \in \mathbb{Z}^d} \sum_{q \in Q_d} \beta_{j,z}^{(q)} \psi_{j,z}^{(q)}(x),$$

with, for all $j \in \mathbb{N}$, $z \in \mathbb{Z}^d$ and $q \in Q_d$,

$$\alpha_{j,z} = \int_{\mathbb{R}^d} \varphi_{j,z}(x)\sqrt{f}(x) \, \mathrm{d}x \qquad \text{and} \qquad \beta_{j,z}^{(q)} = \int_{\mathbb{R}^d} \psi_{j,z}^{(q)}(x)\sqrt{f}(x) \, \mathrm{d}x.$$

The approximation of $g$ to the resolution level $J \geq J_0$ is

$$g_J(x) = \sum_{z \in \mathbb{Z}^d} \alpha_{J_0,z} \varphi_{J_0,z}(x) + \sum_{j=J_0}^{J} \sum_{z \in \mathbb{Z}^d} \sum_{q \in Q_d} \beta_{j,z}^{(q)} \psi_{j,z}^{(q)}(x). \tag{3.5}$$

Now, motivated by the observations made in Section 3.1.1, we define the estimators of the wavelet coefficients $\alpha_{j,z}$ and $\beta_{j,z}^{(q)}$ in (3.5) as

$$\hat{\alpha}_{j,z} = \frac{\Gamma(k)}{\Gamma(k+1/2)} \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \varphi_{j,z}\left(X_i\right) \sqrt{V_{(k);i}}, \qquad j \in \mathbb{N}; z \in \mathbb{Z}^d \tag{3.6}$$

$$\hat{\beta}_{j,z}^{(q)} = \frac{\Gamma(k)}{\Gamma(k+1/2)} \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \psi_{j,z}^{(q)}\left(X_i\right) \sqrt{V_{(k);i}}, \qquad j \in \mathbb{N}; z \in \mathbb{Z}^d; q \in Q_d, \tag{3.7}$$

for some integer $k \geq 1$. The coefficient $\Gamma(k)/\Gamma(k+1/2)$ guarantees the consistency of these estimators, as will arise from the proof of Proposition 3.2.1 below. Note that, for $k=1$, $\Gamma(1)/\Gamma(3/2) = 2/\sqrt{\pi}$, as it was anticipated in Section 3.1.1. Also, in the case $d=1$, when the volume of a ball amounts to the width of an interval, (3.6) and (3.7) can easily be compared to the estimators in Penev and Dechevsky (1997) (their equations (3.2) and (3.3)). Although not identical, they definitely have the same flavor and are asymptotically equivalent.

Plugging (3.6) and (3.7) into the expansion (3.5) produces the estimator

$$\hat{g}_J(x) = \sum_{z \in \mathbb{Z}^d} \hat{\alpha}_{J_0,z} \varphi_{J_0,z}(x) + \sum_{j=J_0}^{J} \sum_{z \in \mathbb{Z}^d} \sum_{q \in Q_d} \hat{\beta}_{j,z}^{(q)} \psi_{j,z}^{(q)}(x). \tag{3.8}$$

For a set of coefficients $\{c_z; z \in \mathbb{Z}^d\}$ essentially defining a particular wavelet family, the father wavelet satisfies $\varphi(x) = \sum_{z \in \mathbb{Z}^d} c_z \varphi(2x - z)$ (and similar for the functions $\psi^{(q)}$'s); see Daubechies (1992). This implies that $\varphi_{j,z}(x) = \sum_{z' \in \mathbb{Z}^d} c_{z'} \varphi_{j+1,z'-2z}(x)$, which, in turn, carries over to the wavelet coefficients, viz. $\alpha_{j,z}^* = \sum_{z' \in \mathbb{Z}^d} c_{z'} \alpha_{j,z-2z'}^*$ (and similar for $\hat{\beta}_{j,z}^{*(q)}$). This so-called *dilation equation* is the key to expression (2.14). Now, substituting in (3.6) yields

$$\begin{aligned} \hat{\alpha}_{j,z} &= \frac{\Gamma(k)}{\Gamma(k+1/2)} \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \varphi_{j,z}\left(X_i\right) \sqrt{V_{(k);i}} \\ &= \frac{\Gamma(k)}{\Gamma(k+1/2)} \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \left\{ \sum_{z' \in \mathbb{Z}^d} c_{z'} \varphi_{j+1,z'-2z}\left(X_i\right) \right\} \sqrt{V_{(k);i}} \\ &= \sum_{z' \in \mathbb{Z}^d} c_{z'} \left\{ \frac{\Gamma(k)}{\Gamma(k+1/2)} \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \varphi_{j+1,z'-2z}\left(X_i\right) \sqrt{V_{(k);i}} \right\} \\ &= \sum_{z' \in \mathbb{Z}^d} c_{z'} \hat{\alpha}_{j+1,z'-2z}, \end{aligned}$$

and similar for $\hat{\beta}_{j,z}^{(q)}$ from (3.7). Hence, although the wavelet estimator developed in this paper is different in nature, the dilation equation applies to it as it does in the conventional case. This directly yields the simple and convenient expression for the estimator, viz.

$$\hat{g}_J(x) = \sum_{z\in\mathbb{Z}^d} \hat{\alpha}_{J+1,z}\varphi_{J+1,z}(x). \tag{3.9}$$

Squaring this up provides an estimator $\hat{f}_J$ of $f$. As already noted in Penev and Dechevsky (1997), estimating $f$ by squaring up an estimate of $\sqrt{f}$ has the additional advantage of providing an easy way for normalising the density estimate. Specifically, enforcing the condition $1 = \int_{\mathbb{R}^d} \hat{f}(x)\,\mathrm{d}x = \int_{\mathbb{R}^d} \hat{g}_J^2(x)\,\mathrm{d}x$ amounts to imposing

$$\sum_{z\in\mathbb{Z}^d} \hat{\alpha}^2_{J_0,z} + \sum_{j=J_0}^{J} \sum_{z\in\mathbb{Z}^d} \sum_{q\in Q_d} \hat{\beta}^{(q)2}_{j,z} = \sum_{z\in\mathbb{Z}^d} \hat{\alpha}^2_{J+1,z} = 1, \tag{3.10}$$

given that the wavelets are orthonormal. If this sum is not 1 after raw estimation of the coefficients by (3.6) and (3.7) but, say, is another constant $\kappa$, it is enough to divide each estimated coefficient by $\sqrt{\kappa}$ for enforcing (3.10). Conventional wavelet estimators do not enjoy such a convenient way of normalising.

In the following section, the asymptotic properties of the coefficient estimators (3.6) and (3.7) are obtained. The asymptotic properties of the estimator (3.8)–(3.9) for $\sqrt{f}$ and the ensuing estimator $\hat{f}_J = \hat{g}_J^2$ for $f$ will be obtained in Section 3.3.

## 3.2 Asymptotic properties of the estimators of the wavelet coefficients

Throughout this thesis we work under the following two standard assumptions.

**Assumption 3.2.1.** The sample $\mathcal{X} = \{X_1, \dots, X_n\}$ consists of i.i.d replications of a random variable $X \in \mathbb{R}^d$ whose distribution $F$ admits a density $f$.

**Assumption 3.2.2.** The functions $\varphi$ and $\psi^{(q)}$ $(q \in Q_d)$, have compact support on $\mathbb{R}^d$ and are bounded. Defining $\varphi_{J_0,z}(x) = 2^{dJ_0/2}\varphi(2^{J_0}x - z)$ and $\psi^{(q)}_{j,z}(x) = 2^{dj/2}\psi^{(q)}(2^j x - z)$, $\{\varphi_{J_0,z}, \psi^{(q)}_{j,z}; j = J_0, \dots, \infty, z \in \mathbb{Z}^d, q \in Q_d\}$ is an orthonormal basis of $L_2(\mathbb{R}^d)$.

Now, the main ingredient in (3.6) and (3.7) is $V_{(k);i}$, which is a quantity of type $k$th Nearest Neighbour ($k$-NN). Procedures based on $k$-NN ideas have always been very popular in nonparametric statistics, from Loftsgaarden and Quesenberry (1965); Devroye and Wagner (1977); Mack and Rosenblatt (1979); Hall (1983b) for density estimation to Lin and Jeon (2006); Gadat at al. (2016) for classification problems, Henze (1988); Mondal et al. (2015) for two-sample testing, and Delattre and Fournier (2017); Ebner et al. (2018) for entropy estimation. The underlying theory has been extensively studied in the literature (Percus and Martin, 1998; Evans et al., 2002; Evans, 2008; Baryshnikov et al., 2009; Kuljus and Ranneby, 2015); see Biau and Devroye (2015) for a recent comprehensive review. Good properties for such $k$-NN quantities require the underlying density $f$ to be well-behaved in the following sense.

**Assumption 3.2.3.** The density $f$ has convex compact support $C \subset \mathbb{R}^d$, with $\sup_{x,y \in C} \|x - y\| = c_1 < \infty$. It is bounded and bounded away from $0$ on $C$, i.e., there exist constants $a_1$ and $a_2$ such that $\inf_{x \in C} f(x) = a_1 > 0$ and $\sup_{x \in C} f(x) = a_2 < \infty$. In addition, $f$ is differentiable on $C$, with uniformly bounded partial derivatives of the first order.

We have then the following result.

**Proposition 3.2.1.** Under Assumptions 3.2.1-3.2.3, for all $j \in \{J_0, \dots, J\}$, $z \in \mathbb{Z}^d$ and $q \in Q_d$, the estimators (3.6) and (3.7) are such that

$$\mathbb{E}(\hat{\alpha}_{j,z}) = \alpha_{j,z} + O(n^{-1/d}), \qquad \mathbb{V}\mathrm{ar}(\hat{\alpha}_{j,z}) = k^3 \left\{ \frac{\Gamma(k)}{\Gamma(k+1/2)} \right\}^2 O(n^{-1})$$

$$\mathbb{E}(\hat{\beta}^{(q)}_{j,z}) = \beta^{(q)}_{j,z} + O(n^{-1/d}), \qquad \mathbb{V}\mathrm{ar}(\hat{\beta}^{(q)}_{j,z}) = k^3 \left\{ \frac{\Gamma(k)}{\Gamma(k+1/2)} \right\}^2 O(n^{-1}),$$

as $n \to \infty$. In particular, if $k$ is such that $k^{3/2}\Gamma(k)/\Gamma(k+1/2) = o(n^{1/2})$, then

$$\mathbb{E}\left\{(\hat{\alpha}_{j,z} - \alpha_{j,z})^2\right\} \to 0 \qquad \text{and} \qquad \mathbb{E}\left\{\left(\hat{\beta}^{(q)}_{j,z} - \beta^{(q)}_{j,z}\right)^2\right\} \to 0$$

as $n \to \infty$, and the estimators are consistent.

*Proof.* The proof makes use of an extension of Theorem 5.4 in Evans et al. (2002), and is given in Appendix. □

The condition $k^{3/2}\Gamma(k)/\Gamma(k+1/2) = o(n^{1/2})$ is obviously satisfied if $k$ keeps a fixed value. If it is the case, then we have directly $\hat{\alpha}_{j,z} = \alpha_{j,z} + O_P(n^{-1/\max(2,d)})$ and $\hat{\beta}^{(q)}_{j,z} = \beta^{(q)}_{j,z} + O_P(n^{-1/\max(2,d)})$ as $n \to \infty$, for all $j \in \{J_0, \dots, J\}$, $z \in \mathbb{Z}^d$ and $q \in Q_d$. It also allows $k$ to grow along with $n$. As $k \to \infty$, $\Gamma(k)/\Gamma(k+1/2) \sim k^{-1/2}$ and the condition is equivalent to $k = o(n^{1/2})$. It appears that the (first order) asymptotic bias of $\hat{\alpha}_{j,z}$ and $\hat{\beta}^{(q)}_{j,z}$ does not depend on $k$, while their (first order) asymptotic variance increases with it. This can be attributed to larger covariances among the $V_{(k);i}$'s as $k$ gets large, and suggests – at least at this level – to keep $k$ as small as possible, that is, to use $k = 1$ always. Below, the results are presented both for $k \doteq k_n$ satisfying $k^{3/2}\Gamma(k)/\Gamma(k+1/2) = o(n^{1/2})$ and for $k = 1$.

## 3.3 Asymptotic properties of the estimators of $\sqrt{f}$ and $f$

### 3.3.1 Pointwise consistency

In this section, the estimator $\hat{g}_J(x)$ (3.8)–(3.9) is first shown to be pointwise consistent for $\sqrt{f}(x)$ at all $x$. This essentially follows from the results of Section 3.2 through the theory of approximating kernels; see Bochner (1955) for early developments, and Meyer (1992); Härdle et al (1998) for the wavelet case. Then we have the following result.

**Proposition 3.3.1.** Under Assumptions 3.2.1-3.2.3, the estimator (3.8)–(3.9) is such that, at all $x \in C$,

(i) $\mathbb{E}\{\hat{g}_J(x)\} = K_{J+1}\sqrt{f}(x) + O(n^{-1/d})$,

(ii) $\left\{\frac{\Gamma(k+1/2)}{\Gamma(k)}\right\}^2 \frac{n}{k^3} \mathbb{V}\text{ar}\{\hat{g}_J(x)\} \le \kappa \int_{\mathbb{R}^d} K^2_{J+1}(x,y)\,\mathrm{d}y + O(n^{-1/d})$,

for some constant $\kappa < \infty$, as $n \to \infty$. Moreover, the order of the remainder terms holds uniformly in $x \in C$.

*Proof.* See Appendix. □

This result obviously implies the pointwise consistency of $\hat{g}_J(x)$ for $\sqrt{f}(x)$ at any fixed $x \in C$ provided that $k^{3/2}\Gamma(k)/\Gamma(k+1/2) = o(n^{1/2})$, in particular if $k$ is kept fixed.

### 3.3.2 Uniform $L_2$-consistency

Consistency in Mean Integrated Squared Error ($L_2$-consistency) of estimator (3.8)-(3.9) can now be established uniformly over large classes of functions, such as Sobolev classes introduced in 2.2.2.

It follows from Assumption 3.2.3 that there exists an integer $m \geq 1$ such that $f \in W^{m,2}(C)$: $f$ has uniformly bounded partial derivatives on $C$, which implies $f \in W^{1,\infty}(C)$, and as $W^{1,\infty}(C) \subset W^{1,2}(C)$, at least $m = 1$. Of course, more regular (i.e., smoother) densities $f$ allow for a higher value of $m$. In addition, under Assumption 3.2.3, $\sqrt{f} \in W^{m,2}(C)$ as well. This appears clearly from the multivariate version of Faà di Bruno's formula (see, e.g., Hardy (2006)), which reads here, for all $\alpha \in \mathbb{N}^d$ such that $|\alpha| \leq m$:

$$D^{\alpha}\sqrt{f} = \sum_{\xi \in \Xi} f^{1/2-|\xi|} \prod_{\beta \in \xi} D^{\beta} f,$$

where $\Xi$ is the set of all partitions $\xi$ of the elements of $\alpha$ and the product is over all 'blocks' $\beta$ of the partition $\xi$. Then the $L^2$-norm of the second factor in each term is bounded because $|\beta| \leq m$ and $f \in W^{m,2}(C)$, and the first factor $f^{1/2-|\xi|}$ is uniformly bounded for all $0 \leq |\xi| \leq m$, because $f$ is both bounded from above (case $|\xi| = 0$) and bounded away from $0$ (case $|\xi| \geq 1$). This also implies that, if $f \in B^{m,2}(L) = \{\phi \in W^{m,2}(C) : \|\phi\|_{m,2} \leq L\}$ for some constant $0 \leq L < \infty$, i.e., a ball of radius $L$ in $W^{m,2}(C)$, then $\sqrt{f} \in B^{m,2}(L')$ for some other constant $0 \leq L' < \infty$.

Now, suppose that the father wavelet $\varphi$ introduced in Assumption 3.2.2 is such that the induced kernel (2.19) satisfies the following assumption.

**Assumption 3.3.1.** The kernel $K$ (2.19) is such that $|K(x,y)| \leq F(x-y)$, for some square integrable function $F : \mathbb{R}^d \to \mathbb{R}$ with $\int_{\mathbb{R}^d} |x|^{\nu} F(x)\,\mathrm{d}x < \infty$ for all $\nu \in \mathbb{N}^d$ such that $|\nu| = m$. Moreover, for all $x \in \mathbb{R}^d$, $\int_{\mathbb{R}^d} (y-x)^{\nu'} K(x,y)\,\mathrm{d}y = \delta_{0,\nu'}$, for all $\nu' \in \mathbb{N}^d$ such that $|\nu'| \leq m-1$.

Here, for $x \in \mathbb{R}^d$ and $\nu \in \mathbb{N}^d$, $|x|^\nu = \prod_{k=1}^d |x_k|^{\nu_k}$, and $\delta_{\nu,\nu'}$ is the $d$-fold Kronecker delta, equal to 1 if $\nu_k = \nu'_k$ for all $k \in \{1, \ldots, d\}$ and 0 otherwise. This assumption is the multivariate version of Conditions H-N in (Härdle et al, 1998, Section 8.3), with the same interpretation as their Remark 8.3. In particular, it allows the pointwise results (Proposition 3.3.1) to be extended uniformly in $x$ and over suitable Sobolev classes of functions.

**Theorem 6.** *Under Assumptions 3.2.1-3.3.1, the estimator* (3.8)–(3.9) *is such that*

$$\sup_{f \in B^{m,2}(L)} \mathbb{E}\left(\|\hat{g}_J - \sqrt{f}\|_2^2\right) \leq \kappa_1 2^{-2Jm} + \kappa_2 n^{-2/d} + \kappa'_3 n^{-1} k^3 \left\{\frac{\Gamma(k)}{\Gamma(k+1/2)}\right\}^2 2^{dJ}, \quad (3.11)$$

*for some constants $\kappa_1, \kappa_2, \kappa'_3 < \infty$ and $n$ large enough.*

*Proof.* See Appendix. □

Clearly, the bound in the right-hand side of (3.11) is a non-decreasing function of $k$, which suggests to take $k = 1$ as it was already noted below Proposition 3.2.1. For that choice, we have directly:

**Corollary 6.1.** *Under Assumptions 3.2.1-3.3.1, the estimator* (3.8)–(3.9) *with $k = 1$ in (3.6)-(3.7) is such that*

$$\sup_{f \in B^{m,2}(L)} \mathbb{E}\left(\|\hat{g}_J - \sqrt{f}\|_2^2\right) \leq \kappa_1 2^{-2Jm} + \kappa_2 n^{-2/d} + \kappa_3 2^{dJ}/n,$$

*for some constants $\kappa_1, \kappa_2, \kappa_3 < \infty$ and $n$ large enough.*

The terms depending on $J$ are balanced for $2^J \propto n^{1/(2m+d)}$, in which case

$$\sup_{f \in B^{m,2}(L)} \mathbb{E}\left(\|\hat{g}_J - \sqrt{f}\|_2^2\right) \leq \kappa' n^{-\frac{2m}{2m+d}} + \kappa'' n^{-2/d},$$

for two constants $\kappa', \kappa'' < \infty$. Note that $\mathbb{E}(\|\hat{g}_J - \sqrt{f}\|_2^2)$ is the expected squared HD between the true $f$ and its estimator $\hat{g}_J^2$, hence is a meaningful risk measure as such. By the Cauchy-Schwarz inequality, it follows in any case

$$\|\hat{g}_J^2 - f\|_2^2 = \|(\hat{g}_J - \sqrt{f})(\hat{g}_J + \sqrt{f})\|_2^2 \leq \|\hat{g}_J - \sqrt{f}\|_2^2 \times \|\hat{g}_J + \sqrt{f}\|_2^2.$$

Assumptions 3.2.2 and 3.2.3 ensure that the second factor is bounded, whereby we have the following result about $\hat{g}_J^2$ as an estimator of the density $f$.

**Theorem 7.** *Under Assumptions 3.2.1-3.3.1, the estimator $\hat{g}_J^2$ with $k = 1$ in (3.6)–(3.7) and $2^J \propto n^{1/(2m+d)}$ is a uniformly $L_2$-consistent estimator of $f$, such that*

$$\sup_{f \in B^{m,2}(L)} \mathbb{E}\left(\|\hat{g}_J^2 - f\|_2^2\right) \leq \kappa' n^{-2m/(2m+d)} + \kappa'' n^{-2/d}, \tag{3.12}$$

*for some constants $\kappa', \kappa'' < \infty$.*

Note that the first term in the right-hand side of (3.12) is the optimal nonparametric rate of convergence in this situation, as per the classical results of Stone (1980). That term is dominated by the second one only for $d > 2m/(m-1)$. Hence we have the following corollary.

**Corollary 7.1.** *Under Assumptions 3.2.1-3.3.1, the estimator $\hat{g}_J^2$ with $k = 1$ in (3.6)–(3.7) and $2^J \propto n^{1/(2m+d)}$ is asymptotically optimal for $f$ uniformly over $B^{m,2}(L) \subset W^{m,2}(C)$, in the sense that, for $d \leq 2m/(m-1)$,*

$$\sup_{f \in B^{m,2}(L)} \mathbb{E}\left(\|\hat{g}_J^2 - f\|_2^2\right) \leq \kappa' n^{-2m/(2m+d)}.$$

As $2m/(m-1) > 2$, the estimator is always optimal in one and two dimensions. Under the classical mild smoothness assumption $m = 2$, it is optimal for $1 \leq d \leq 4$ – this probably covers most of the cases of practical interest, given that the optimal rate of convergence itself becomes very poor in higher dimensions (*Curse of Dimensionality*). In any case, for 'rough' densities $f$ ($m = 1$), the estimator reaches the optimal rate in all dimensions.

## 3.4 Numerical experiments

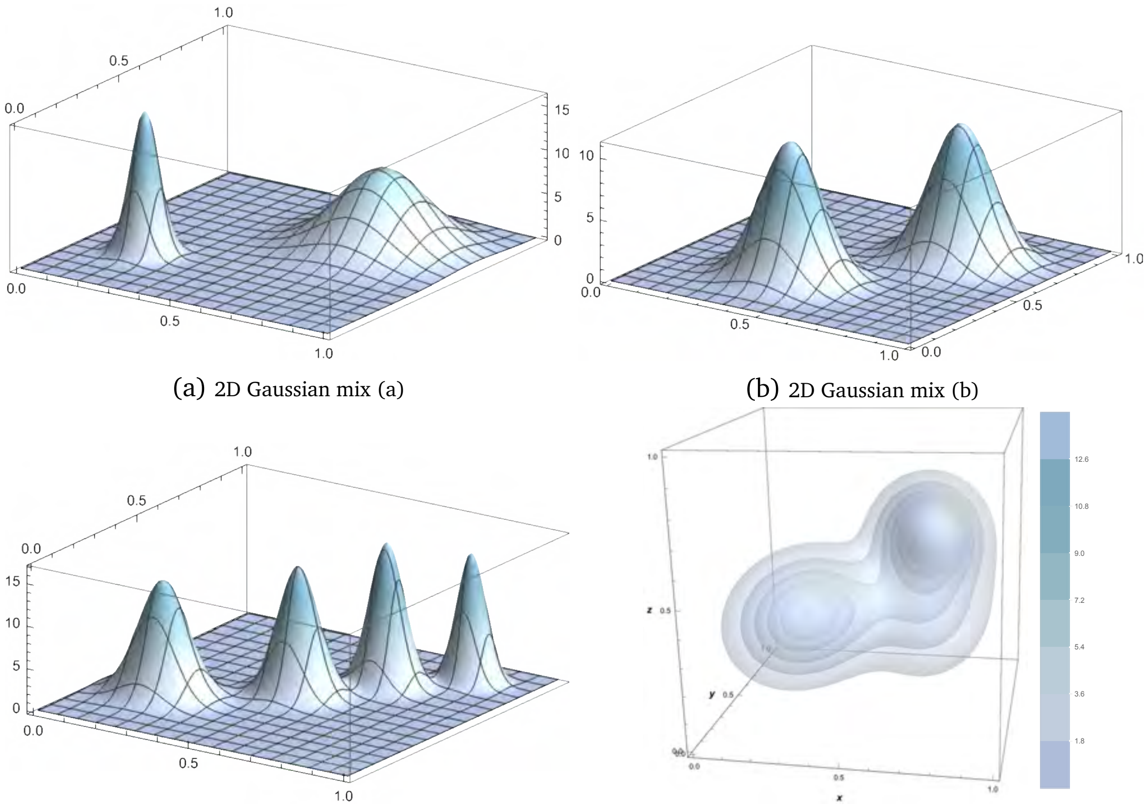
### 3.4.1 Simulation study

In this section the practical performance of the shape-preserving estimator $\hat{g}_J^2$ based on (3.8)–(3.9), normalised by forcing (3.10), is compared to that of the classical wavelet estimator. Normalisation is of course necessary for $\hat{f} = \hat{g}^2$ to be considered a PDF. Three bivariate ($d = 2$) and one trivariate ($d = 3$) Gaussian mixtures were considered: (a) two bivariate components, showing two peaks with very different covariance structures (Figure 3.1a); (b) two bivariate components, showing two similar peaks (Figure 3.1b); (c) a bivariate version of the 'smooth comb' (Marron and Wand, 1992), showing 4 peaks of decreasing spread (Figure 3.1c); and (d) three trivariate components producing a bimodal density (Figure 3.1d). The exact expressions are available in the Appendix B.1. Those where scaled and truncated to the unit hypercube $[0,1]^d$, in order to satisfy Assumption 3.2.3. Note that mixtures (a)-(c)-(d) exhibit peaks of different spread and orientation, features known to cause difficulties in density estimation.

For each density, $M = 500$ random samples of size $n = 2^\ell$, for $\ell \in \{7,\ldots,13\}$, i.e., from $n = 128$ up to $n = 8192$,[1] were generated, and our procedure was used on each of them for estimating $f$. Proper normalisation of all estimates was enforced through (3.10). The accuracy of a given estimate $\hat{f}$ was measured by the Integrated Squared Error (ISE) $\int_{[0,1]^d}\{\hat{f}(x) - f(x)\}^2\,\mathrm{d}x$ and the Squared Hellinger Distance (SHD) between $\hat{f}$ and $f$, i.e., $\frac{1}{2}\int_{[0,1]^d}\{\sqrt{\hat{f}}(x) - \sqrt{f}(x)\}^2\,\mathrm{d}x$. Both were approximated by Riemannian summing on a fine regular partition of $[0,1]^d$. The MISE and the Mean Squared Hellinger Distance (MSHD) of an estimator was then approximated by averaging the ISE's and SHD's over the $M = 500$ Monte Carlo replications, see Tables 3.1 and 3.2.

Estimators (3.6)-(3.7) were computed with bivariate wavelets $\varphi_{j,z}$ and $\psi_{j,z}^{(q)}$ obtained by tensor products of univariate Daubechies wavelets with 10 vanishing

[1] Sample sizes as powers of 2 are customary in the wavelet framework due to their suitability when resorting to the Fast Wavelet Transform, however the estimator described in Section 3.1.2 remains obviously valid for any arbitrary sample size $n$.



(a) 2D Gaussian mix (a)

(b) 2D Gaussian mix (b)

(c) 2D Comb (c)

(d) 3D Gaussian mix (d)

Figure 3.1. Densities used in the simulation study.

moments (Daubechies, 1992). In agreement with the asymptotic results, the value $k = 1$ in (3.6)-(3.7) was given primary focus, but $k \in \{2, 4, 8, .., \sqrt{n}\}$ were also tested to investigate the effect of $k$ in finite samples. For the three densities and all sample sizes, the choice $k = 1$ always lead to the final estimator with the smallest MISE or MSHD, or within statistical significance (given $M = 500$ Monte Carlo replications) to the estimator with the smallest MISE. Hence in Table 3.1 only the results for $k = 1$ are reported. In (3.8), the baseline resolution was taken $J_0 = 0$ and the resolution levels $J \in \{-1, 0, 1, 2, 3\}$ were considered – the case $J = -1$ is here defined as the estimator with the trend at baseline level $J_0 = 0$ only. For comparison, the density $f$ was also estimated on each sample by the classical wavelet estimator described in Härdle et al (1998), whose MISE and MSHD were approximated in the exact same way as above. For computing the HD, though, it was necessary to consider the non-negative part of that estimator.

The whole procedure was developed in Python, using the BallTree $k$-NN algorithm (Omohundro, 1989) and the PyWavelets library that supports a number of orthogonal and biorthogonal wavelet families.

Table 3.1. (Approximated) Mean Integrated Square Errors (MISE) for the shape-preserving estimator (SPWDE), the classical wavelet estimator (Class.) and the kernel density estimator (KDE) for different sample sizes. Wavelet estimators were computed for different values of $J+1$ ($J_0 = 0$). The smallest MISE is highlighted for each sample size.

**2D Gaussian mix (a)**

| n | J+1 | SP | Class. | KDE |
|---|---|---|---|---|
| 128 | 0 | 3.500 | 3.690 | 0.600 |
| | 1 | 2.485 | 2.763 | |
| | 2 | 1.316 | 1.087 | |
| | **3** | **1.077** | **0.676** | |
| | 4 | 4.306 | 1.727 | |
| 256 | 0 | 3.501 | 3.692 | 0.410 |
| | 1 | 2.474 | 2.761 | |
| | 2 | 1.227 | 1.055 | |
| | **3** | **0.634** | **0.488** | |
| | 4 | 2.218 | 0.915 | |
| 512 | 0 | 3.501 | 3.691 | 0.284 |
| | 1 | 2.465 | 2.758 | |
| | 2 | 1.196 | 1.039 | |
| | **3** | **0.416** | **0.388** | |
| | 4 | 1.066 | 0.472 | |
| 1024 | 0 | 3.502 | 3.691 | 0.193 |
| | 1 | 2.462 | 2.757 | |
| | 2 | 1.164 | 1.030 | |
| | **3** | **0.292** | 0.336 | |
| | 4 | 0.521 | **0.250** | |
| 2048 | 0 | 3.502 | 3.691 | 0.132 |
| | 1 | 2.460 | 2.757 | |
| | 2 | 1.148 | 1.026 | |
| | **3** | **0.235** | 0.310 | |
| | 4 | 0.258 | **0.136** | |
| | 5 | 1.432 | 0.480 | |
| 4096 | 0 | 3.503 | 3.691 | 0.084 |
| | 1 | 2.459 | 2.756 | |
| | 2 | 1.140 | 1.023 | |
| | 3 | 0.204 | 0.296 | |
| | **4** | **0.131** | **0.077** | |
| | 5 | 0.620 | 0.242 | |
| 8192 | 0 | 3.503 | 3.691 | 0.054 |
| | 1 | 2.459 | 2.756 | |
| | 2 | 1.132 | 1.022 | |
| | 3 | 0.188 | 0.290 | |
| | **4** | **0.067** | **0.048** | |
| | 5 | 0.290 | 0.123 | |

**2D Gaussian mix (b)**

| n | J+1 | SP | Class. | KDE |
|---|---|---|---|---|
| 128 | 0 | 4.172 | 4.267 | 0.418 |
| | 1 | 2.975 | 3.285 | |
| | **2** | **0.760** | 1.101 | |
| | 3 | 0.831 | **0.438** | |
| | 4 | 3.964 | 1.760 | |
| 256 | 0 | 4.171 | 4.268 | 0.279 |
| | 1 | 2.963 | 3.278 | |
| | 2 | 0.676 | 1.060 | |
| | **3** | **0.440** | **0.243** | |
| | 4 | 2.008 | 0.908 | |
| 512 | 0 | 4.171 | 4.268 | 0.179 |
| | 1 | 2.958 | 3.275 | |
| | 2 | 0.638 | 1.042 | |
| | **3** | **0.233** | **0.143** | |
| | 4 | 1.039 | 0.465 | |
| 1024 | 0 | 4.171 | 4.267 | 0.121 |
| | 1 | 2.955 | 3.273 | |
| | 2 | 0.619 | 1.033 | |
| | **3** | **0.119** | **0.096** | |
| | 4 | 0.519 | 0.238 | |
| 2048 | 0 | 4.171 | 4.268 | 0.077 |
| | 1 | 2.953 | 3.272 | |
| | 2 | 0.610 | 1.028 | |
| | **3** | **0.063** | **0.070** | |
| | 4 | 0.258 | 0.119 | |
| | 5 | 1.299 | 0.480 | |
| 4096 | 0 | 4.171 | 4.268 | 0.050 |
| | 1 | 2.953 | 3.272 | |
| | 2 | 0.609 | 1.026 | |
| | **3** | **0.034** | **0.058** | |
| | 4 | 0.127 | 0.060 | |
| | 5 | 0.588 | 0.243 | |
| 8192 | 0 | 4.171 | 4.268 | 0.032 |
| | 1 | 2.955 | 3.272 | |
| | 2 | 0.609 | 1.025 | |
| | **3** | **0.020** | 0.052 | |
| | 4 | 0.064 | **0.030** | |
| | 5 | 0.277 | 0.123 | |

**2D Comb (c)**

| n | J+1 | SP | Class. | KDE |
|---|---|---|---|---|
| 128 | 0 | 6.191 | 6.274 | 0.926 |
| | 1 | 5.118 | 5.252 | |
| | 2 | 3.532 | 3.875 | |
| | **3** | **1.271** | **1.275** | |
| | 4 | 3.365 | 1.756 | |
| 256 | 0 | 6.190 | 6.275 | 0.616 |
| | 1 | 5.108 | 5.249 | |
| | 2 | 3.436 | 3.833 | |
| | **3** | **0.869** | 1.078 | |
| | 4 | 1.776 | **0.891** | |
| 512 | 0 | 6.190 | 6.274 | 0.410 |
| | 1 | 5.096 | 5.245 | |
| | 2 | 3.388 | 3.812 | |
| | **3** | **0.656** | 0.985 | |
| | 4 | 0.956 | **0.481** | |
| 1024 | 0 | 6.190 | 6.274 | 0.270 |
| | 1 | 5.095 | 5.243 | |
| | 2 | 3.366 | 3.803 | |
| | 3 | 0.556 | 0.940 | |
| | **4** | **0.497** | **0.264** | |
| 2048 | 0 | 6.190 | 6.274 | 0.177 |
| | 1 | 5.093 | 5.243 | |
| | 2 | 3.353 | 3.799 | |
| | 3 | 0.500 | 0.915 | |
| | **4** | **0.253** | **0.149** | |
| | 5 | 1.081 | 0.479 | |
| 4096 | 0 | 6.190 | 6.274 | 0.114 |
| | 1 | 5.091 | 5.243 | |
| | 2 | 3.347 | 3.796 | |
| | 3 | 0.479 | 0.903 | |
| | **4** | **0.130** | **0.094** | |
| | 5 | 0.531 | 0.244 | |
| 8192 | 0 | 6.190 | 6.274 | 0.073 |
| | 1 | 5.088 | 5.242 | |
| | 2 | 3.345 | 3.795 | |
| | 3 | 0.468 | 0.897 | |
| | **4** | **0.068** | **0.065** | |
| | 5 | 0.262 | 0.121 | |

**3D Gaussian mix (d)**

| n | J+1 | SP | Class. | KDE |
|---|---|---|---|---|
| 128 | 0 | 2.889 | 3.020 | 0.591 |
| | 1 | 1.480 | 1.712 | |
| | **2** | **0.807** | **0.527** | |
| | 3 | 6.435 | 2.846 | |
| | 4 | 54.752 | 21.605 | |
| 256 | 0 | 2.887 | 3.019 | 0.425 |
| | 1 | 1.445 | 1.697 | |
| | **2** | **0.471** | **0.374** | |
| | 3 | 3.565 | 1.514 | |
| | 4 | 34.649 | 10.832 | |
| 512 | 0 | 2.887 | 3.021 | 0.305 |
| | 1 | 1.428 | 1.692 | |
| | **2** | **0.276** | **0.289** | |
| | 3 | 1.896 | 0.791 | |
| | 4 | 22.053 | 5.598 | |

| n | J+1 | SP | Class. | KDE |
|---|---|---|---|---|
| 1024 | 0 | 2.886 | 3.020 | 0.211 |
| | 1 | 1.420 | 1.688 | |
| | **2** | **0.185** | **0.246** | |
| | 3 | 0.975 | 0.407 | |
| | 4 | 12.523 | 2.942 | |
| 2048 | 0 | 2.887 | 3.020 | 0.151 |
| | 1 | 1.418 | 1.687 | |
| | **2** | **0.134** | 0.225 | |
| | 3 | 0.499 | **0.208** | |
| | 4 | 6.825 | 1.566 | |
| 4096 | 0 | 2.887 | 3.020 | 0.103 |
| | 1 | 1.416 | 1.686 | |
| | **2** | **0.107** | 0.215 | |
| | 3 | 0.250 | **0.108** | |
| | 4 | 3.044 | 0.825 | |

In terms of MISE, both estimators show comparable performance. The observed differences in MISE are low, sometimes giving preference to one estimator and

Table 3.2. (Approximated) Mean Squared Hellinger Distance (MSHD) for the shape-preserving estimator (SP), the classical wavelet estimator (Class.) and the kernel density estimator (KDE) for different sample sizes. Wavelet estimators were computed for different values of $J+1$ ($J_0 = 0$). The smallest MSHD is highlighted for each sample size.

**2D Gaussian mix (a)**

| n | J+1 | SP | Class. | KDE |
|---|---|---|---|---|
| 128 | 0 | 0.445 | 0.563 | 0.061 |
| | 1 | 0.264 | 0.387 | |
| | **2** | **0.117** | 0.212 | |
| | 3 | 0.122 | **0.116** | |
| | 4 | 0.358 | 0.224 | |
| 256 | 0 | 0.446 | 0.563 | 0.042 |
| | 1 | 0.262 | 0.387 | |
| | 2 | 0.104 | 0.208 | |
| | **3** | **0.073** | **0.087** | |
| | 4 | 0.230 | 0.137 | |
| 512 | 0 | 0.447 | 0.564 | 0.029 |
| | 1 | 0.261 | 0.387 | |
| | 2 | 0.098 | 0.204 | |
| | **3** | **0.045** | **0.069** | |
| | 4 | 0.133 | 0.082 | |
| 1024 | 0 | 0.447 | 0.564 | 0.019 |
| | 1 | 0.261 | 0.386 | |
| | 2 | 0.094 | 0.202 | |
| | **3** | **0.029** | 0.059 | |
| | 4 | 0.073 | **0.050** | |
| 2048 | 0 | 0.448 | 0.564 | 0.013 |
| | 1 | 0.262 | 0.386 | |
| | 2 | 0.092 | 0.201 | |
| | **3** | **0.020** | 0.053 | |
| | 4 | 0.040 | **0.032** | |
| | 5 | 0.157 | 0.070 | |
| 4096 | 0 | 0.448 | 0.564 | 0.009 |
| | 1 | 0.263 | 0.386 | |
| | 2 | 0.091 | 0.201 | |
| | **3** | **0.016** | 0.050 | |
| | 4 | 0.021 | **0.022** | |
| | 5 | 0.087 | 0.039 | |
| 8192 | 0 | 0.448 | 0.564 | 0.006 |
| | 1 | 0.263 | 0.386 | |
| | 2 | 0.090 | 0.201 | |
| | 3 | 0.014 | 0.048 | |
| | **4** | **0.011** | **0.016** | |
| | 5 | 0.047 | 0.021 | |

**2D Gaussian mix (b)**

| n | J+1 | SP | Class. | KDE |
|---|---|---|---|---|
| 128 | 0 | 0.570 | 0.643 | 0.044 |
| | 1 | 0.354 | 0.502 | |
| | **2** | **0.093** | 0.213 | |
| | 3 | 0.101 | **0.100** | |
| | 4 | 0.339 | 0.212 | |
| 256 | 0 | 0.571 | 0.643 | 0.029 |
| | 1 | 0.352 | 0.500 | |
| | 2 | 0.082 | 0.207 | |
| | **3** | **0.057** | **0.069** | |
| | 4 | 0.213 | 0.128 | |
| 512 | 0 | 0.571 | 0.643 | 0.020 |
| | 1 | 0.351 | 0.500 | |
| | 2 | 0.076 | 0.204 | |
| | **3** | **0.032** | **0.050** | |
| | 4 | 0.128 | 0.077 | |
| 1024 | 0 | 0.571 | 0.643 | 0.013 |
| | 1 | 0.352 | 0.499 | |
| | 2 | 0.073 | 0.203 | |
| | **3** | **0.018** | **0.040** | |
| | 4 | 0.071 | 0.044 | |
| 2048 | 0 | 0.571 | 0.643 | 0.009 |
| | 1 | 0.351 | 0.499 | |
| | 2 | 0.071 | 0.202 | |
| | **3** | **0.010** | 0.035 | |
| | 4 | 0.039 | **0.025** | |
| | 5 | 0.153 | 0.068 | |
| 4096 | 0 | 0.571 | 0.643 | 0.006 |
| | 1 | 0.352 | 0.499 | |
| | 2 | 0.071 | 0.202 | |
| | **3** | **0.005** | 0.032 | |
| | 4 | 0.020 | **0.015** | |
| | 5 | 0.085 | 0.038 | |
| 8192 | 0 | 0.571 | 0.643 | 0.004 |
| | 1 | 0.352 | 0.499 | |
| | 2 | 0.071 | 0.202 | |
| | **3** | **0.003** | 0.030 | |
| | 4 | 0.011 | **0.008** | |
| | 5 | 0.045 | 0.021 | |

**2D Comb (c)**

| n | J+1 | SP | Class. | KDE |
|---|---|---|---|---|
| 128 | 0 | 0.770 | 0.820 | 0.067 |
| | 1 | 0.546 | 0.693 | |
| | 2 | 0.309 | 0.438 | |
| | **3** | **0.100** | **0.174** | |
| | 4 | 0.244 | 0.192 | |
| 256 | 0 | 0.770 | 0.820 | 0.045 |
| | 1 | 0.545 | 0.693 | |
| | 2 | 0.299 | 0.432 | |
| | **3** | **0.067** | 0.154 | |
| | 4 | 0.145 | **0.120** | |
| 512 | 0 | 0.770 | 0.820 | 0.030 |
| | 1 | 0.544 | 0.692 | |
| | 2 | 0.294 | 0.430 | |
| | **3** | **0.050** | 0.145 | |
| | 4 | 0.085 | **0.078** | |
| 1024 | 0 | 0.770 | 0.820 | 0.021 |
| | 1 | 0.544 | 0.692 | |
| | 2 | 0.292 | 0.429 | |
| | **3** | **0.041** | 0.140 | |
| | 4 | 0.047 | **0.052** | |
| 2048 | 0 | 0.770 | 0.820 | 0.014 |
| | 1 | 0.544 | 0.692 | |
| | 2 | 0.290 | 0.428 | |
| | 3 | 0.036 | 0.137 | |
| | **4** | **0.025** | **0.036** | |
| | 5 | 0.100 | 0.056 | |
| 4096 | 0 | 0.770 | 0.820 | 0.009 |
| | 1 | 0.544 | 0.692 | |
| | 2 | 0.290 | 0.428 | |
| | 3 | 0.033 | 0.136 | |
| | **4** | **0.014** | **0.027** | |
| | 5 | 0.055 | 0.031 | |
| 8192 | 0 | 0.770 | 0.820 | 0.006 |
| | 1 | 0.544 | 0.692 | |
| | 2 | 0.289 | 0.428 | |
| | 3 | 0.032 | 0.135 | |
| | **4** | **0.007** | 0.022 | |
| | 5 | 0.029 | **0.017** | |

**3D Gaussian mix (d)**

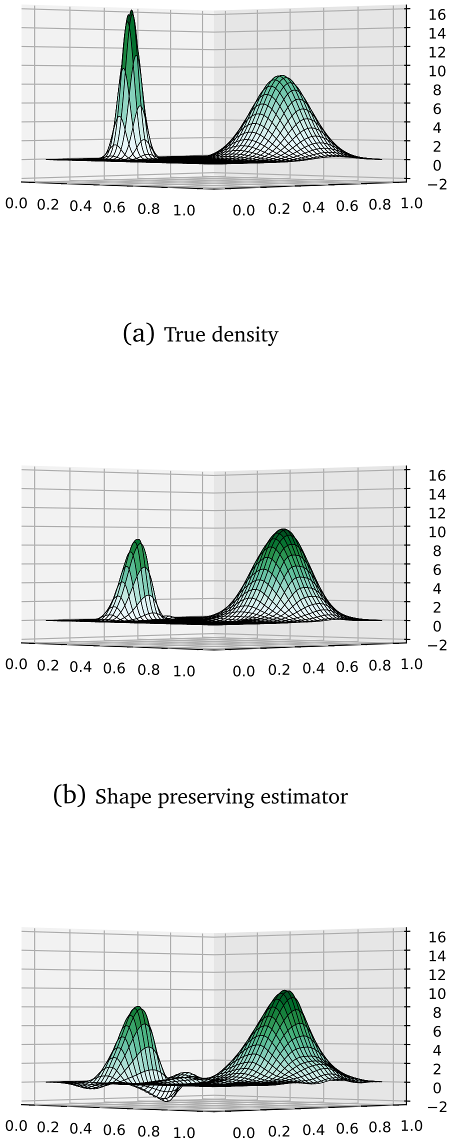
| n | J+1 | SP | Class. | KDE |
|---|---|---|---|---|
| 128 | 0 | 0.392 | 0.461 | 0.066 |
| | 1 | 0.152 | 0.257 | |
| | **2** | **0.115** | **0.120** | |
| | 3 | 0.480 | 0.368 | |
| | 4 | 0.869 | 0.911 | |
| 256 | 0 | 0.392 | 0.461 | 0.047 |
| | 1 | 0.146 | 0.254 | |
| | **2** | **0.068** | **0.091** | |
| | 3 | 0.352 | 0.239 | |
| | 4 | 0.767 | 0.688 | |
| 512 | 0 | 0.392 | 0.461 | 0.034 |
| | 1 | 0.143 | 0.253 | |
| | **2** | **0.041** | **0.075** | |
| | 3 | 0.235 | 0.150 | |
| | 4 | 0.670 | 0.499 | |
| 1024 | 0 | 0.393 | 0.461 | 0.024 |
| | 1 | 0.142 | 0.252 | |
| | **2** | **0.026** | **0.066** | |
| | 3 | 0.144 | 0.092 | |
| | 4 | 0.555 | 0.342 | |
| 2048 | 0 | 0.393 | 0.461 | 0.017 |
| | 1 | 0.142 | 0.252 | |
| | **2** | **0.017** | 0.062 | |
| | 3 | 0.083 | **0.055** | |
| | 4 | 0.422 | 0.222 | |
| 4096 | 0 | 0.393 | 0.461 | 0.012 |
| | 1 | 0.141 | 0.251 | |
| | **2** | **0.013** | 0.059 | |
| | 3 | 0.046 | **0.033** | |
| | 4 | 0.286 | 0.137 | |

sometimes to the other, without clear pattern. By contrast, in terms of the MSHD, the Shape-Preserving Wavelet-based Density Estimation (SPWDE) estimator clearly outperforms the classical estimator, with in some cases an MSHD twice as low. This can obviously be understood by the fact that the HD between densities is based on their respective square-roots. Hence, the SP estimator, primarily based on estimating $\sqrt{f}$, perfectly aligns with that criterion. Given the adequacy of the HD for assessing the proximity between densities, this clearly gives a real edge to the SP estimator over the classical one beyond solving automatically the negativity issue.

As an illustration, Figure 3.2 shows typical estimates for the shape-preserving estimator and the classical one for sample size $n = 4096$ ($k = 1$, $J_0 = 0$ and $J = 3$). Note how the classical estimator loses mass in areas of low density, even for this large sample.

Figure 3.2 also reveals how challenging it is, for both estimators, to re-construct two peaks of such different spread. In that respect, the introduction of a thresholding scheme would be helpful to allow a higher resolution to be selected while killing out any unwarranted noise. The shape-preserving estimator is expected to profit more from the introduction of such thresholding, as it is noted from Tables 3.1-3.2 that the classical estimator sometimes allows a higher resolution, already. More on this topic in Section 4.2.

Finally, as it may be thought of as an 'overall benchmark' in nonparametric density estimation, the classical multivariate kernel density estimator (Wand and Jones, 1995, Chapter 4) was also computed on the same $M = 500$ samples for each sample size for each of the 4 densities shown in Figure 3.1, and its MISE and MSHD reported in Tables 3.1-3.2 as well. A diagonal bandwidth matrix (with different diagonal entries) was used, selected using Cross-Validation (CV). It is seen that KDE does better than the (linear) SPWDE estimator for small sample sizes. An explanation may lie in the fact that variability of nearest-neighbour type statistics is high in small samples, directly affecting the coefficient estimators (3.6)-(3.7), and hence ultimately $\hat{g}_J$; see (A.6) in Appendix. On the other hand, for large sample sizes, the SP wavelet estimator catches up and even beats KDE. It is reasonable to believe that, with the above-mentioned thresholding, which is essentially the strength of the wavelet method, the SP estimator would outperform the competition in a more pronounced way, and this even for small sample sizes. We will



(a) True density

(b) Shape preserving estimator

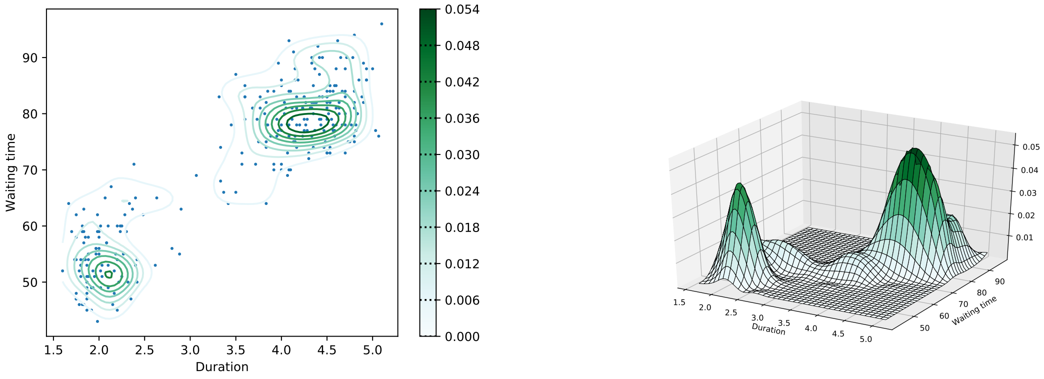
(c) Classical estimator

Figure 3.2. Comparisons of estimates for Gaussian Mixture (a), $n = 4096$, $k = 1$ and $J_0 = 0$ and $J = 3$.

present a data driven thresholding algorithm in Section 4.3 along with some insights into its asymptotic behaviour.

### 3.4.2 Real data: Old Faithful geyser

Old Faithful geyser is a very active geyser in the Yellowstone National Park, Wyoming, USA. Data on eruption times and waiting times (both in minutes) between erup-



(a) Scatter and contour plot

(b) 3D-density estimate

Figure 3.3. Old Faithful dataset

tions of Old Faithful form a well-known bivariate data set of $n = 272$ observations. In particular, it was used for illustration in Vannucci (1995), in a review of different types of wavelet density estimators. The shape-preserving estimator was computed on these data using Daubechies wavelets with 7 vanishing moments (as in Vannucci (1995)). Visually, the most appealing result was obtained with $J_0 = 0$ and $J = 2$, producing the estimate shown in Figure 3.3. As opposed to Figure 6 in Vannucci (1995), the shape-preserving estimator shows some small bumps of potential interest near the main peaks. In view of the raw data (scatter plot, left panel), this seems legitimate.

# Chapter 4

# The non-linear shape-preserving wavelet-based density estimator

## 4.1 Overview

Recall that the shape-preserving wavelet-based estimator for the square root of a density was defined by (3.8) as

$$\hat{g}_J(x) = \sum_{z \in \mathbb{Z}^d} \hat{\alpha}_{J_0,z} \varphi_{J_0,z}(x) + \sum_{j=J_0}^{J} \sum_{z \in \mathbb{Z}^d} \sum_{q \in Q_d} \hat{\beta}^{(q)}_{j,z} \, \psi^{(q)}_{j,z}(x)$$

and that, as mentioned in Subsection 2.3.2, these estimators have the ability to capture local phenomena by applying a threshold to the coordinates in the wavelet decomposition. Among the various thresholding approaches, we focus here on hard thresholding of beta coefficients whereby only the beta coefficients that are larger in magnitude than a given value are kept. This can be done for instance as in (2.31)

$$\tilde{\beta}^{(q)}_{jz} \doteq \begin{cases} \hat{\beta}^{(q)}_{jz}, & \text{if } \left|\hat{\beta}^{(q)}_{jz}\right| > K\,C(j) n^{-1/2} \\ 0, & \textit{otherwise} \end{cases},$$

where $K$ is to be determined and $C(j)$ makes the threshold level-dependant.

Aligned with Proposition 3.2.1 and remarks thereafter, we continue to use $k = 1$, i.e. nearest neighbour distances. However, several parameters still need to be defined:

- the initial resolution level $J_0$
- the highest level $J$ (usually referred to as $J_1$ in the literature) or, equivalently, the number of additional levels in the beta expansion;
- the cut-off point or threshold as defined by $K$ and resolution-varying function $C(j)$;
- or, in a general hard thresholding like (2.31), find $\tau_{j,z}^{(q)} = 0, 1$ in $\tilde{\beta}_{j,z}^{(q)} = \tau_{j,z}^{(q)} \hat{\beta}_{j,z}^{(q)}$ that select or remove coefficients from the wavelet series;
- the wavelet basis $\{\varphi_{j,z}, \psi_{j,z}^{(q)}\}$.

In addition, the results regarding near optimal asymptotic behaviour presented in Subsection 3.3.2 were announced under some assumptions that, as with traditional wavelet estimators, were formulated in terms of the regularity of the estimated density. In practice this is a drawback since, in general, it is impossible to know the parameters of the functional class where the function sits (Härdle et al, 1998). To address this, the chapter presents data driven methods that define several of the parameters required above, making possible the practical application of our methodology.

As pointed out in remarks following Theorem 6 and its corollary, the Hellinger distance is a meaningful and natural risk measure for the shape-preserving estimator based on the square root of the density. In here, we use this metric along with Leave-one-out Cross-Validation (LOO-CV) to propose a framework whereby the parameters above can be defined.

This chapter is structured as follows. In Section 4.2, we introduce the framework and present two methods to find a suitable *best resolution* using the HD. Based on this, Section 4.3 further extends the framework to select $C(j)$ and find $K$ or $\tau_{j,z}^{(q)}$ to make possible a hard threshold, non-linear definition of the estimator. Finally, in Section 4.4, simulation results are shown demonstrating the practicality of the above procedures.

## 4.2 Resolution selection

As pointed out in Vannucci and Vidakovic (1997), one of the major issues in orthogonal series density estimation is choosing the amount of basis functions to use. In the specific case of wavelet series, this is usually posed as finding the right (projecting) scale parameter at which to truncate the wavelet series (3.9). It is described in Hall and Penev (2001) as the *primary resolution* level, which plays a similar role to the bandwidth parameter in the kernel density estimator, and so a correct choice of this quantity can alleviate difficulties caused by over-smoothing, when too small, or over-fitting, when too large (Vannucci and Vidakovic, 1997).

In fact, wavelet-based estimators are forgiving of over-smoothing. As noted in Hall and Penev (2001), the penalty for a very large degree of over-smoothing is only a logarithmic function of sample size in asymptotic terms and the reason that taking a small (conservative) primary resolution level performs relatively well. However, wavelet estimators are no more "forgiving of under-smoothing" than are conventional kernel estimators, and so choosing a too large resolution level can degrade performance fairly quickly . This has been demonstrated also in experiments for our shape-preserving estimator in the linear case (Chapter 3), where increasing the resolution level produces a sharp decline in performance.

In fact, contrary to Hall and Penev (2001) where the authors propose a "multiple cross-validation" algorithm based on subregions, constructed via visual inspection of a pilot estimate, that takes the *minimum* of the corresponding best resolutions among those; here we present a CV algorithm to find an ideal resolution level that is then used as the *maximum* resolution when it comes to calculate a non-linear threshold of coefficients (see Theorem 9). So, instead of picking a $J_0$ for alphas and then adding some additional levels $J_0 \ldots J_1$ of betas, our approach here can be understood as finding a reasonably good $J+1$ in (3.9) and working our way backwards towards (3.8), pruning some betas in the process, to increase the effectiveness of the initial estimate.

Marron (1987) presents and discusses various data-driven methods for choosing the bandwidth in delta sequence estimators, i.e. linear estimators of the form

$\hat{f}_\lambda(x) = n^{-1}\sum_{i=1}^n \boldsymbol{\delta}_\lambda(x, X_i)$, where $\boldsymbol{\delta}_\lambda : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is indexed by the smoothness parameter $\lambda$. Kernel-based and orthogonal series estimators are of this form (Walter, 1992). In kernel density estimation, $\boldsymbol{\delta}_h(x, X_i) = \frac{1}{h}K((x - X_i)/h)$. In wavelet-based density estimation, alpha only expansions are also of this form, where $\boldsymbol{\delta}_j(x, X_i) = K_j(x, X_i)$, $K_j$ defined as (2.20). To see this, take the alpha only part of (2.25) and rewrite the estimator

$$\begin{aligned}\hat{f}_j(x) &= \sum_{z\in\mathbb{Z}^d} \hat{\alpha}_{j,z}\varphi_{j,z}(x) \\ &= \sum_{z\in\mathbb{Z}^d} \left(\frac{1}{n}\sum_i^n \varphi_{j,z}(X_i)\right)\varphi_{j,z}(x) \\ &= n^{-1}\sum_i^n \sum_{z\in\mathbb{Z}^d} \varphi_{j,z}(X_i)\varphi_{j,z}(x) \\ &= n^{-1}\sum_i^n K_j(X_i, x). \end{aligned} \tag{4.1}$$

Among the methods surveyed in Park and Marron (1990) to select $\lambda$, the most widely studied is least squares CV, proposed by Rudemo (1982) and Bowman (1984). This has been shown, for the kernel density estimator, to asymptotically converge to the optimum under very weak conditions (Stone (1984); Burman (1985)). In fact, Marron (1987) shows that choosing the bandwidth $\lambda$ by these methods in the somewhat general framework of delta sequences is, in a strong sense, asymptotically equivalent to minimising the MISE. For the square root estimator, this is the Hellinger distance, which motivates the definitions below.

As in this section we focus on determining an optimal resolution level, let $\hat{g}_J$ represent the alphas-only estimator at level $J$ of the square root of $f$, as it is similarly done for the same problem in the traditional wavelet estimator (Tribouley, 1995). The HD between that estimator and the true density $f$ is

$$HD(\hat{g}_J^2, f)^2 = \frac{1}{2}\int \left(|\hat{g}_J(x)| - \sqrt{f(x)}\right)^2 \mathrm{d}x = 1 - \int |\hat{g}_J(x)|\sqrt{f(x)}\,\mathrm{d}x, \tag{4.2}$$

assuming that $\hat{g}_J$ has been normalised, i.e. that $\int \hat{g}_J^2(x)\,\mathrm{d}x = 1$ by enforcing (3.10) as explained there.

A subtlety here that seemed important in our numerical experiments is that although $\hat{g}_J^2 = f$ is shape-preserving, $\hat{g}_J$ can be negative. Hence the use of $|\hat{g}_J| = \sqrt{\hat{g}_J^2}$ above. The last integral is the Bhattacharyya Coefficient (BC) $\mathcal{B}_J$ between the estimator $\hat{g}_J^2$ and $f$ (Bhattacharyya, 1946). Minimising the HD is of course equivalent to maximising this term, although it depends on $f$ and is therefore not available in practice.

Let $\hat{h}_J$ be the standard wavelet-based density estimator (2.25) without the betas. Let $S^{(-i)} = \{X_j : j \neq i\}$ be the sample excluding $X_i$ and $\hat{h}_J^{(-i)}$ the same estimator constructed from $S^{(-i)}$. It can be easily shown that, asymptotically as $n \to \infty$,

$$\mathbb{E}\left\{\frac{1}{n}\sum_{i=1}^{N}\hat{h}_J^{(-i)}(X_i)\right\} \to \mathbb{E}\left\{\int \hat{h}_J(x)f(x)\,\mathrm{d}x\right\},$$

which suggests to use as estimator for the risk $ISE(\hat{h}_J, f) = \left\|\hat{h}_J - f\right\|^2 = \left\|\hat{h}_J\right\|^2 + 2\int \hat{h}_J(x)f(x)\,\mathrm{d}x + \|f\|^2$ the quantity

$$CV(J) = \int \hat{h}_J^2(x)\,\mathrm{d}x - \frac{2}{n}\sum_{i=1}^{N}\hat{h}_J^{(-i)}(X_i), \tag{4.3}$$

which, although it does not include $\|f\|^2 = \int f^2$, can be used to determine a best resolution level $J$ as the missing, unknown term does not depend on $J$ (Tribouley, 1995). This formulation is nothing but the traditional LOO-CV estimator for the parameter $J$ (Stone, 1974).

Going back to the BC, the occurrence of $\sqrt{f}$ under the integral sign immediately suggests a similar approach to (3.4), where $\left|\hat{g}_J^{(-i)}\right|$ plays the role of the square integrable $\phi$ and where we "replace" $\sqrt{f(X_i)}$ by $\frac{2}{\sqrt{\pi}\,n}\sqrt{V_{(1);i}}$, thus constructing an empirical BC via LOO-CV.

We need to be more explicit than in Chapter 3 regarding normalisation of the estimator. Here, we denote by $\mathring{g}$ the unnormalised, raw estimator (3.8), whereas the normalised estimator becomes $\hat{g} = \frac{1}{\|\mathring{g}\|}\mathring{g}$. Note, this normalised version is the one we used to report our results there. Thus, based on (3.6), (3.7) and (3.8) (excluding the betas) we define the leave-one-out versions below.

Let the super-script $(-i)$ denote our estimator calculated from the sample minus the $i$-th observation. The alphas-only version of the raw estimator, $\mathring{g}_J^{(-i)}$, is

$$\mathring{g}_J^{(-i)}(x) = \sum_z \hat{\alpha}_{J,z}^{(-i)} \varphi_{J,z}(x) \tag{4.4}$$

$$= \sum_z \left( \frac{2}{\sqrt{\pi(n-1)}} \sum_{i' \neq i} \varphi_{J,z}(X_{i'}) \sqrt{V_{(k);i'}^{(-i)}} \right) \varphi_{J,z}(x). \tag{4.5}$$

From this the normalised estimator excluding observation $i$ is

$$\left\| \mathring{g}_J^{(-i)} \right\|^2 = \sum_z \left( \hat{\alpha}_{J,z}^{(-i)} \right)^2 \tag{4.6}$$

$$\hat{g}_J^{(-i)}(x) = \frac{1}{\left\| \tilde{g}_J^{(-i)} \right\|} \mathring{g}_J^{(-i)}(x). \tag{4.7}$$

Then, by our previous discussion, we have that

$$\widehat{\mathcal{B}}_J^{(v)} = \frac{2}{\sqrt{\pi n}} \sum_i \left| \hat{g}_J^{(-i)}(X_i) \right| \sqrt{V_{(1);i}} \tag{4.8}$$

is a reasonable approximation to $\mathcal{B}_J^{(v)} = \int |\hat{g}_J(x)| \sqrt{f(x)} \, \mathrm{d}x$ - we use the superscript $(v)$ to indicate the use of the normalised $\hat{g}_J$ (we will use $(u)$ for the unnormalised counterpart).

With the above, we determine the smoothing parameter $J$ as done for delta estimators (Marron, 1987) using the Least Squares (LS) approach. For an i.i.d sample $S$ of size $n$, the data-driven, best resolution level is calculated as

$$\hat{J}_n^{(v)} = \operatorname*{argmax}_j \frac{2}{\sqrt{\pi n}} \sum_i \left| \hat{g}_j^{(-i)}(X_i) \right| \sqrt{V_{(1);i}}. \tag{4.9}$$

This is our first method.

A slightly different approach (Geenens and Lafaye de Micheaux, 2020) is to remove $\|\hat{g}\| = 1$ in derivation (4.2) so as to deal with the $L^2$ distance between the unnormalised $\mathring{g}$ and $\sqrt{f}$, which leads to

$$\frac{1}{2} \left\| |\mathring{g}_J| - \sqrt{f} \right\|^2 = \frac{1}{2} \int \mathring{g}_J(x)^2 dx - \int |\mathring{g}_J(x)| \sqrt{f(x)} \, \mathrm{d}x + \frac{1}{2}. \tag{4.10}$$

The middle term can be subject to the same treatment as above. Although $\mathring{g}_J$ is not a PDF and we have a correcting $\frac{1}{2}\left\|\mathring{g}_J\right\|$ term, with a slight abuse of Bhattacharyya 's terminology, we still refer to below using

$$\mathcal{B}_j^{(u)} = \int |\mathring{g}_j(x)| \sqrt{f(x)}\,\mathrm{d}x - \frac{1}{2}\int \mathring{g}_j(x)^2\,\mathrm{d}x. \tag{4.11}$$

This leads to the alternative best resolution formulation

$$\begin{aligned}\hat{J}_n^{(u)} &= \underset{j}{\operatorname{argmax}}\,\widehat{\mathcal{B}}_j^{(u)} \\ &= \underset{j}{\operatorname{argmax}}\left\{\frac{2}{\sqrt{\pi}n}\sum_i \left|\mathring{g}_j^{(-i)}(x_i)\right|\sqrt{V_{(1);i}} - \frac{1}{2}\left\|\mathring{g}_j\right\|^2\right\}. \end{aligned} \tag{4.12}$$

The advantage of this approach is not computational (we know that $\|\mathring{g}_j\|$ can be easily calculated), but procedural - it helps us to determine the performance of this estimator a lot easier because it does not have a ratio form with the parameter being optimised in the denominator.

Let the true $J$ for $\hat{g}_J$ (resp. $\mathring{g}_J$) be $J_n^{*(v)} = \operatorname{argmax}_j \mathcal{B}_j^{(v)}$ (resp. $J_n^{*(u)} = \operatorname{argmax}_j \mathcal{B}_j^{(u)}$). Although we know $\mathring{g}_J \to \sqrt{f}$ in HD when $2^J \propto n^{1/(2m+d)}$ for $k = 1$ (see Corollary 6.1 and following remark), it remains to be seen that the optimum $\mathcal{B}_j^{(v)}$ converges in some sense to the optimal $J$ for $\hat{g}_J$.

To do this, we use the same arguments outlined in Rudemo (1982); Hall (1982, 1983b) adapted to our setting. Namely, that the estimated resolution level will be asymptotically optimal as the best (true) resolution if $\mathcal{B}^{(m)}_{J^{*(m)}}/\mathcal{B}^{(m)}_{\hat{J}_n^{(m)}} \to 1$ in probability. For this, we adapt the strategy and results of Marron (1987) about the asymptotic optimality of smoothing parameter determination via LOO-CV. Those are general enough to apply to the two step process outlined in the overview: resolution selection and thresholding parameter finding.

Denote by $A_n$ the search parameter space for both steps, the resolution level and the thresholding parameter. We introduce the following assumptions.

**Assumption 4.2.1.** Denote $\#\left(A_n\right)$ be the cardinality of the parameter space for the smoothing parameter. Then $\#\left(A_n\right) \leqslant \mathcal{C}n^{\rho}$ for some $\mathcal{C}, \rho > 0$, i.e. it grows at most algebraically fast.

**Assumption 4.2.2.** There are two positive constants $\mathcal{C}', \epsilon > 0$, such that for any possible parameter choice $\lambda \in A_n$, $\mathcal{C}'^{-1} n^{\epsilon} \le \lambda \le \mathcal{C}' n^{1-\epsilon}$.

**Theorem 8.** *Under assumptions 3.2.1-3.3.1, 4.2.1-4.2.2, for both, the normalised $(m) = (v)$ and unnormalised $(m) = (u)$ methods of $J$ resolution selection, $\hat{J}_n^{(m)}$ is asymptotically optimal. This is, the true BC at the true optimum level $J^{*(m)}$ is asymptotically equal to the true BC calculated at the optimum $\hat{J}_n^{(m)}$ found using the estimated coefficient. Formally,*

$$\lim_{n\to\infty} \frac{\mathcal{B}^{(m)}_{J^{*(m)}}}{\mathcal{B}^{(m)}_{\hat{J}_n^{(m)}}} = 1 \qquad a.s.$$

The proof for $(m) = (u)$, found in the appendix, is a recast of Theorem 2 in Marron (1987) for delta sequences, where the metric under consideration is the $L^2$ distance in the hypersphere (4.10). There are some other rather technical assumptions added in Marron (1987), but they are proven to hold for kernel and orthogonal series estimators in Marron and Härdle (1986); Marron (1987) under the other assumptions that we adopted here throughout the thesis. Assumption 4.2.1 is simple enough for this and the following algorithm as we will discuss, leaving Assumption 4.2.2 as the only other important assumption we have added.

The major subtlety in adopting the approach of Marron (1987) lies in the fact that our alphas-only estimator is not exactly a delta sequence. So, instead of (4.1) above, we have

$$\begin{aligned}
\mathring{g}_J(x) &= \sum_z \hat{\alpha}_{J,z} \varphi_{J,z}(x) \\
&= \sum_z \left( \frac{2}{\sqrt{\pi n}} \sum_i^n \varphi_{J,z}(X_i) \sqrt{V_{(1);i}} \right) \varphi_{J,z}(x) \\
&= \sum_i^n \left( \sum_z \varphi_{J,z}(X_i) \varphi_{J,z}(x) \right) \frac{2}{\sqrt{\pi n}} \sqrt{V_{(1);i}} \\
&= \sum_i^n K_J(X_i, x) \frac{2}{\sqrt{\pi n}} \sqrt{V_{(1);i}} \\
&= \sum_i^n \boldsymbol{\delta}_J(X_i, x) F_{i,n}, \qquad (4.13)
\end{aligned}$$

where the factor $\frac{1}{n}$ representing the Dirac mass at $X_i$, $\delta_{X_i}$, is replaced by a factor

$F_{i,n}$ (a random variable) that represents the square root of the density at $i$. Thus, a delta sequence estimator can be seen as $\hat{f}_\lambda(x) = \sum_i \boldsymbol{\delta}_\lambda(X_i, x)\delta_{X_i}$ (see (2.28)); whereas our construction is, in an abstract sense[1], $\mathring{g}_j(x) = \sum_i \boldsymbol{\delta}_j(X_i, x)\sqrt{\delta_{X_i}}$. The case $(m) = (v)$ follows naturally as asymptotically $\mathcal{B}_j^{(u)} \to \mathcal{B}_j^{(v)} - \frac{1}{2}$ as $n \to \infty$.

## 4.3 Thresholding coefficients

Our thresholding algorithm is again based on HD (4.2) and an empirical approximation similar to (4.8), along with the same variant around the use of $\mathring{g}_J$ instead of $\hat{g}_J$. First, fix $J_0$ and $J_1$. Note that in (2.31) and by using $C(j) = \sqrt{j - J_0 + 1}$ (Delyon and Juditsky, 1996), $\hat{\beta}_{j,z}^{(q)}$ is chosen if

$$\frac{\left|\hat{\beta}_{j,z}^{(q)}\right|}{\sqrt{j - J_0 + 1}} > \lambda, \tag{4.14}$$

where we have absorbed $\sqrt{n}$ into the unknown constant $\lambda$. Call the left hand side $\lambda_{j,z,q}$ and let $\lambda_{j^{[t]},z^{[t]},q^{[t]}}$ be the corresponding descending ordering of those lambdas between levels $J_0$ and $J_1$. Let us use these indexes to reorder the betas from largest to smallest and, with a little bit of abuse of notation, call them $\beta_{[t]}$, identifying $[t]$ with corresponding $(j^{[t]}, z^{[t]}, q^{[t]})$. We can now write a hard-threshold version of our unnormalised estimator with coefficients up to $\lambda_{[\tau]}$ (see Figure 4.1) as

$$\mathring{g}_{[\tau]}(x) = \sum_{z \in \mathbb{Z}^d} \hat{\alpha}_{J_0,z}\varphi_{J_0,z}(x) + \sum_{t=1}^{\tau} \hat{\beta}_{[t]}\, \psi_{[t]}(x). \tag{4.15}$$

With a slight abuse of notation, we can think of all the alpha coefficients as represented by $\alpha_{j,z} = \beta_{j,z}^{(0)}$ and being prepended to the list of selected indexes (always retained) and likewise $\varphi_{j,z}$ represented by $\psi_{j,z}^{(0)}$. With this notation, the expression above can be put succinctly as

$$\mathring{g}_{[\tau]}(x) = \sum_{t=1}^{\tau} \hat{\beta}_{[t]}\, \psi_{[t]}(x). \tag{4.16}$$

The same normalisation as the original algorithm and a similar variant for the LOO-CV formulation should allow us to construct $\hat{g}_{[\tau]}^{(-i)}$. Now, similar to deriva-

[1] Of course, to be understood formally, as the square root of the Dirac's $\delta_x$ function does not make sense. The reader, however, may be familiar with the work of Craven (1985).

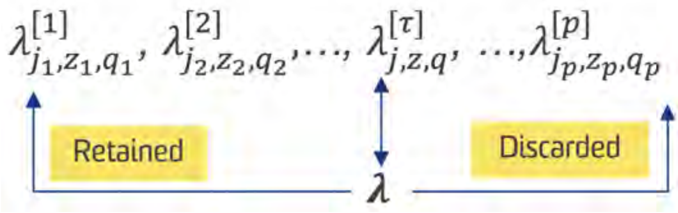

$\lambda^{[1]}_{j_1,z_1,q_1}, \lambda^{[2]}_{j_2,z_2,q_2}, \ldots, \lambda^{[\tau]}_{j,z,q}, \ldots, \lambda^{[p]}_{j_p,z_p,q_p}$
Retained
Discarded
$\lambda$

Figure 4.1. Thresholding coefficients up to $\lambda$

tion (4.13), hard thresholding as per (2.31) up to $\lambda_{[\tau]}$ can be written as a delta sequence as

$$\mathring{g}_{[\tau]} = \sum_{i}^{n} \boldsymbol{\delta}_{[\tau]}\left(x, X_i\right) F_{i,n}, \tag{4.17}$$

where

$$\boldsymbol{\delta}_{[\tau]}\left(x, X_i\right) = \sum_{t=1}^{\tau} \psi_{[t]}(x)\psi_{[t]}\left(X_i\right). \tag{4.18}$$

As mentioned above, we are expressing our estimator like that to make the results of Marron (1987) for orthogonal series density estimators available. It is worth adding that our notation $\hat{g}_{[\tau]}$ emphasises the fact that, after fixing the levels, there are only a finite number of cut points for the threshold constant $\lambda$ that matter in a given sample $\{X_i\}$, although it is of course meaningful to express, in all generality, the hard-thresholding estimator for an arbitrary $\lambda$ as $\hat{g}_\lambda$.

Proceeding in a similar fashion to the smoothing parameter $J$ above, define the empirical BC for a given $\lambda_{[\tau]}$ as

$$\widehat{\mathcal{B}}^{(v)}_{[\tau]} = \frac{2}{\sqrt{\pi n}} \sum_{i} \left|\hat{g}^{(-i)}_{[\tau]}(X_i)\right| \sqrt{V_{(1);i}}, \tag{4.19}$$

and the corresponding empirical optimum $\hat{\lambda}^{(v)}_n$ for samples of size $n$ as

$$\hat{\lambda}^{(v)}_n = \underset{\lambda_{[\tau]}}{\operatorname{argmax}} \frac{2}{\sqrt{\pi n}} \sum_{i} \left|\hat{g}^{(-i)}_{[\tau]}(X_i)\right| \sqrt{V_{(1);i}}. \tag{4.20}$$

Let the true $\lambda$ be $\lambda^{*(v)} = \sup_\lambda \mathcal{B}^{(v)}_\lambda = \sup_\lambda \int |\hat{g}_\lambda(x)| \sqrt{f(x)}\, \mathrm{d}x$, with $\hat{g}_\lambda$ as previously described. The unnormalised variants $\widehat{\mathcal{B}}^{(u)}_{[\tau]}$ and $\hat{\lambda}^{(u)}_n$ can be defined in similar fashion as in previous section, viz.

$$\hat{\lambda}_n^{(u)} = \underset{\lambda_{[\tau]}}{\operatorname{argmax}} \widehat{\mathcal{B}}^{(u)}_{\lambda_{[\tau]}} \tag{4.21}$$

$$\widehat{\mathcal{B}}^{(u)}_{\lambda_{[\tau]}} = \frac{2}{\sqrt{\pi n}} \sum_i \left|\mathring{g}^{(-i)}_{[\tau]}(x_i)\right| \sqrt{V_{(1);i}} - \frac{1}{2} \left\|\mathring{g}_{[\tau]}\right\|^2 . \tag{4.22}$$

Then we have a similar result as for the resolution level

**Theorem 9.** *Fix $J_1 = \hat{J}_n^{(m)}$ as per (4.9) and (4.12) for $(m) = (v), (u)$ respectively. Fix $J_0 \leq J_1$ with $J_1 - J_0 = O(\log n)$. Then under assumptions 3.2.1-3.3.1, 4.2.1-4.2.2, where the parameter space here is $A_n = \{\lambda_{j,z,q}\}$, the quantities $\hat{\lambda}_n^{(m)}$ as defined by (4.20) and (4.21) are asymptotically optimal, this is*

$$\lim_{n\to\infty} \frac{\mathcal{B}^{(m)}_{\lambda^{*(m)}}}{\mathcal{B}^{(m)}_{\hat{\lambda}_n^{(m)}}} = 1 \; a.s.$$

*for $(m) = (v), (u)$.*

The sketch of the proof uses the same tools as Theorem 8 for $\hat{J}_n^{(m)}$ and can be found in the appendix.

Note that in our algorithm the optimal resolution level is $J_1$, with $J_0$ chosen few levels below. As the constant in $J_1 - J_0 = O(\log n)$ is undetermined, one can follow the standard practice of picking up few levels difference between these two.

Now, we propose a novel variation around (4.14). In Donoho and Johnstone (1996), the right hand side of (2.31), $K\,C(j)n^{1/2}$, depends on $\sigma$, the standard deviation of the terms $\psi^{(q)}_{j,z}(X_i) - \mathbb{E}\left[\psi^{(q)}_{j,z}(X_i)\right]$, which in turn is linked to the errors $\hat{\beta}_{j,z} - \beta_{j,z}$. As $\hat{g}^{(-i)}_{[\tau]}$ requires calculating $\left(\hat{\beta}^{(q)}_{j,z}\right)^{(-i)}$, we can use these to calculate an empirical $\hat{\sigma}^{(q)}_{j,z}$ leading to this novel threshold formulation

$$\frac{\left|\hat{\beta}^{(q)}_{jz}\right|}{\hat{\sigma}^{(q)}_{j,z}} > \lambda \tag{4.23}$$

where $\left(\hat{\sigma}^{(q)}_{j,z}\right)^2 = \mathbb{Var}\left\{\left(\hat{\beta}^{(q)}_{j,z}\right)^{(-i)}\right\}$ is the jackknife estimator of variance of each beta coefficient (Tukey, 1958; Miller, 1974; Efron and Stein, 1981). One by-product of above is that we have eliminated the need for a $C(j)$ in (2.31), a subject of theoretical controversy (Donoho et al, 1995; Donoho and Johnstone, 1996; Delyon and Juditsky, 1996), pushing further for a fully data-driven threshold design.

Asymptotically, this is equivalent to our algorithm above but our simulations have shown this approach to have benefits in most cases. Future work is required to analyse this particular threshold method further.

## 4.4 Numerical experiments for non-linear estimator

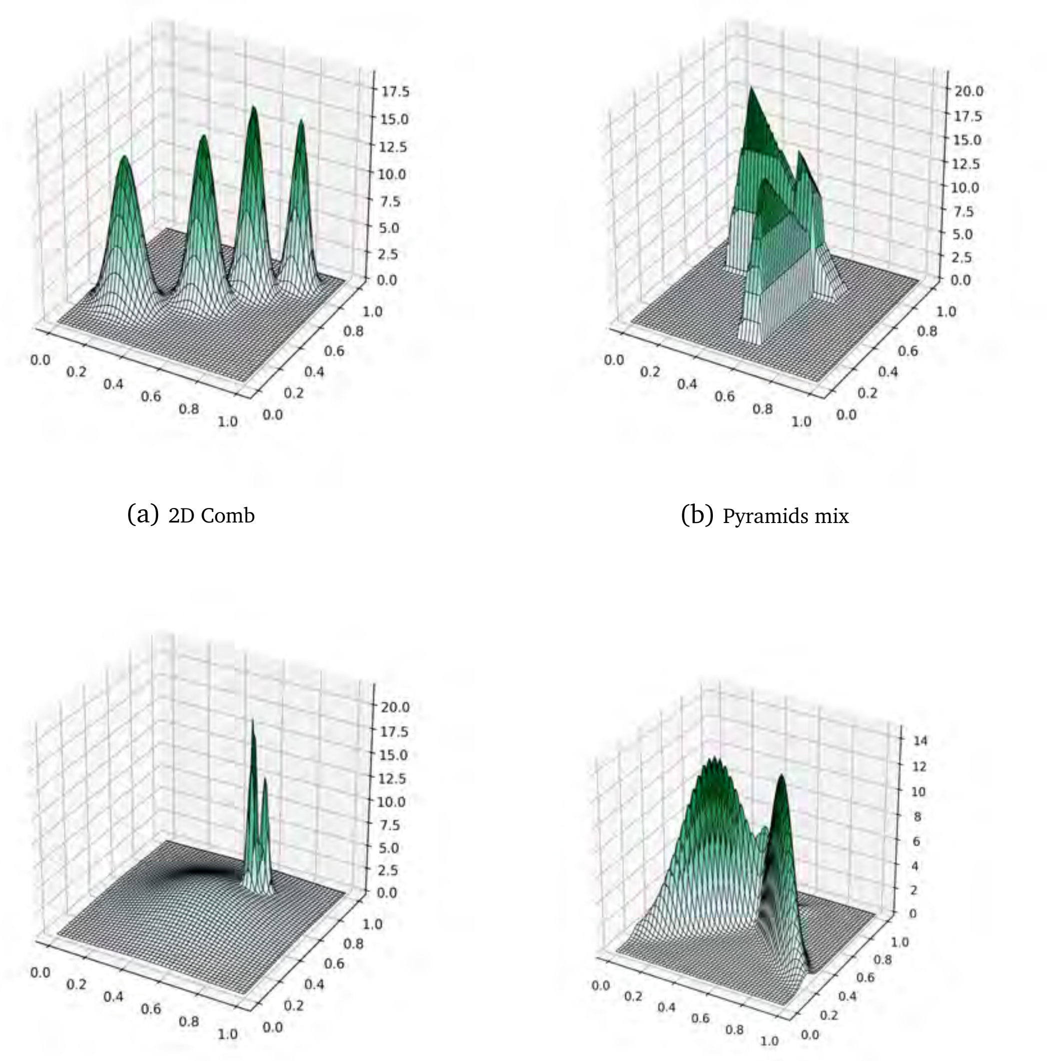
Our algorithm for the non-linear case has two stages: first, determine the best resolution to use and then calculate the hard-threshold value for beta coefficients around that resolution. In the same manner, this section follows the same strategy: first, it presents the performance of the best resolution level, comparing against its true value, and second, it shows results for the various thresholding options and discuss them.

### 4.4.1 Resolution level

In this section we present simulation results to determine the best resolution level using the algorithms outlined in Section 4.2. Recall that we have two approaches, $\hat{J}_n^{(v)}$ and $\hat{J}_n^{(u)}$, defined by (4.9) and (4.12), using the normalised and unnormalised estimator respectively. Mixtures of different number of components, smoothness and covariance structures were tested against a few wavelet bases with different regularity. Densities used are shown in Figure 4.2: (a) is a similar 2D Gaussian comb as in Chapter 3; (b) is a mixture of two *pyramid* densities[2], i.e. non smooth; (c) is a 2D Gaussian mixture with very different spreads; and (d) a mixture of Gaussians with very elongated covariances in different directions. Analytic forms are in the appendix.

For each of these densities and for different sizes, 100 samples of i.i.d observations were generated. Unlike experiments in the previous chapter, we opted here to have sample sizes not following a power of two progression to highlight inter-

[2] These *pyramids* are constructed by taking the tensor product of two piecewise functions and ensuring it integrates to one.



(a) 2D Comb

(b) Pyramids mix

(c) 2D Gaussian mix 1

(d) 2D Gaussian mix 2

Figure 4.2. Densities used in best $J$ simulation study.

actions between the resolution level and the sample size. As the true density is available, we calculated the HD of $\hat{g}_n^2$ to $f$ and from this, computed the theoretical best $J_n^{*(m)}$, $(m) = (v), (u)$, for each sample. We compared the difference between the estimated $\hat{J}_n^{(m)}$ and the true $J_n^{*(m)}$ for the different wavelet bases and the two algorithms $(m) = (v), (u)$, producing Figure 4.3 to Figure 4.6.

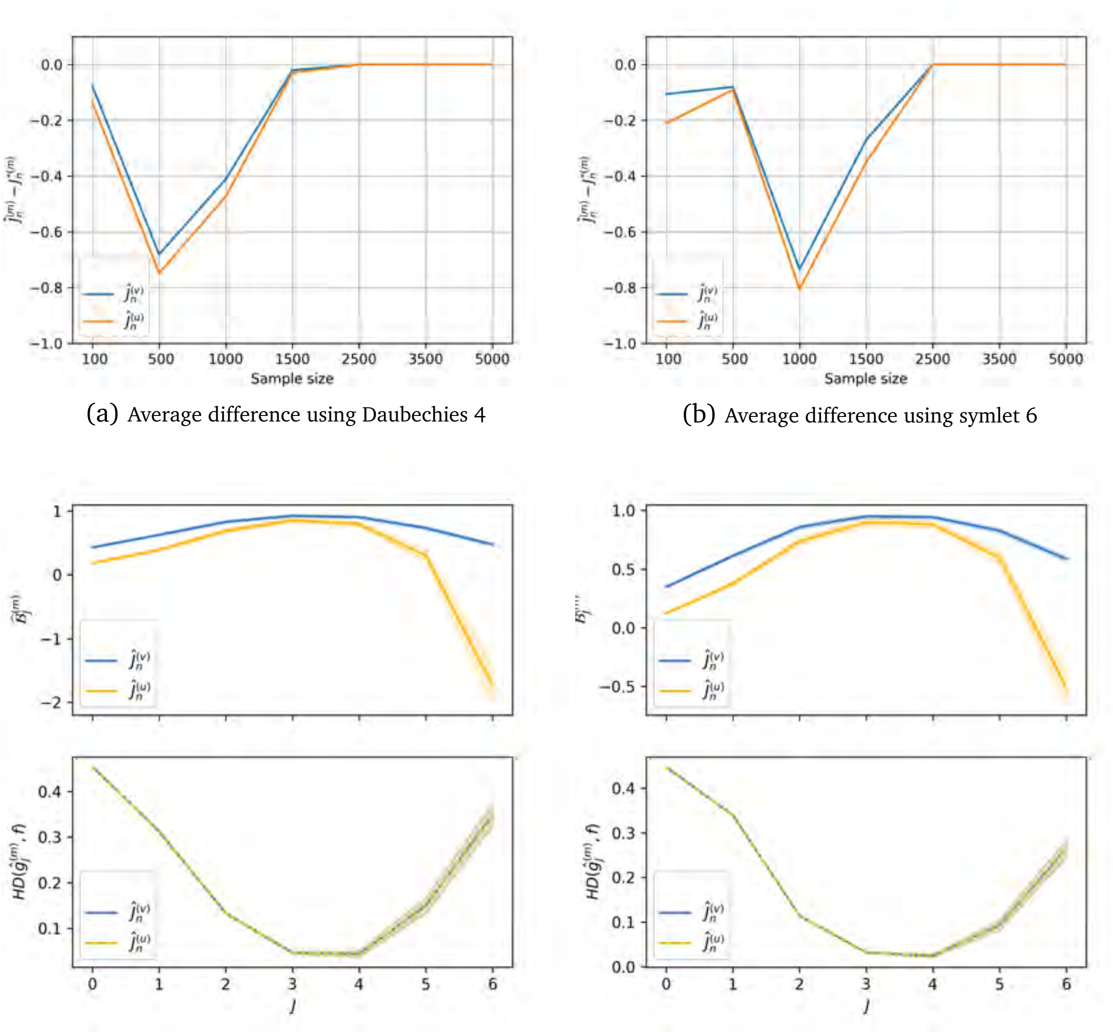
Figure 4.3 illustrates the performance of the method for Figure 4.3. In (a) and (b), the point of highest average difference between calculated and true $J$ is at $n = 500$ and $n = 1000$ for the Daubechies 4 and symlet 6 wavelet basis respectively. Below (a), in (c), we present the corresponding $\widehat{B}_J^{(m)}$ plots for both $(m) = (v), (u)$ for the

Daubechies wavelet at sample size $n = 500$. In these plots, the normed version, $\hat{J}_n^{(v)}$, lies between $[0, 1]$, whereas $\hat{J}_n^{(u)}$ diverges as the estimator overfits the sample as $J$ increases. However, the impact overall seems minimal as what matters is the position of the peak. For comparison, the calculated HD to $f$ is displayed at the bottom of (c). Note that both, $(v)$ and $(u)$ methods produce the same HD - the blue and amber lines are dotted so they are clearly superimposed on each other. This is expected: even if we use the unnormed version of $\widehat{B}_J^{(u)}$ to determine $J$, the resulting function is always normalised, i.e. a *bona fide* density. Below (b), in (d), the case $n = 1000$ for the symlet 6 wavelet is shown. Similarly to (c), the peak of both curves for the BC, normalised and unnormalised, appear very flat. The true HD on the other hand, seems well defined at $J = 4$. The result is that, sometimes, the estimated $\hat{J}^{(m)}$ may have a peak at $J = 3$ instead of $4$, producing the average difference shown in the plot above. However, in these cases, it seems to be under-predicting, which results in a conservative $J$.

A similar situation is depicted in Figure 4.4. Here, the average difference reaches $-1$ at relatively large sample sizes. Here the bottom of the $HD$ curve is slightly tilted on the $J = 5$ side whereas the peak of both curves, $\widehat{B}_J^{(v)}$ and $\widehat{B}_J^{(u)}$, is on the $J = 4$ side. In this case, it makes sense to approach the problem of picking a resolution level in the continuum, as suggested in (Hall and Penev, 2001). See Chapter 6, Discussion.

In the next section, we will see that despite this apparent shortcoming, further extending the estimator by applying thresholding does indeed improve its performance. Finally, here we also demonstrate that our shape-preserving estimator works well in the biorthogonal case which we describe briefly in Chapter 6. Specifically, here we extended our wavelet estimator for biorthogonal wavelets using the estimated coefficients (6.1) and the norm (6.3).

The density in Figure 4.2 (c) is a typical example of suitability of wavelet thresholding in density estimation, as it exhibits components with high locality. At a relatively large sample size, we see in Figure 4.5, (a) and (b), that the resolution level seems to be underpredicted for $n = 5000$. In this particular case, the corresponding curves (c) and (d) appear to have a plateau in the range $J = 2 \ldots 4$.



(a) Average difference using Daubechies 4

(b) Average difference using symlet 6

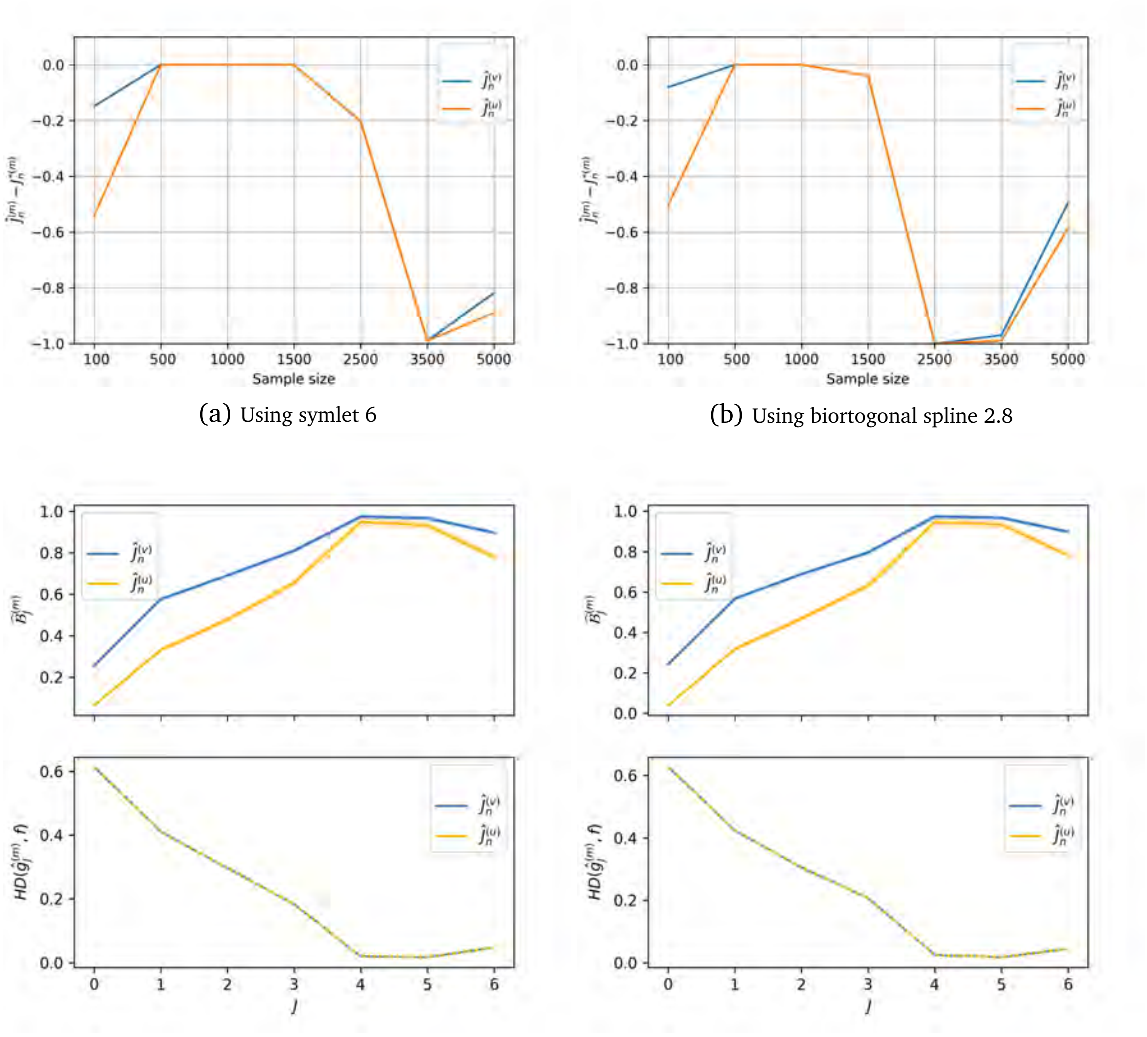
(c) $\widehat{B}_J^{(m)}$ and $HD$ for $n = 500$

(d) $\widehat{B}_J^{(m)}$ and $HD$ for $n = 1000$

Figure 4.3. Results in best J methods over 100 simulations for each sample size for the density shown in Figure 4.2 (a). Top row, average difference in calculated $\hat{J}_n^{(m)}$ versus true value $J_n^{*(m)}$ for two wavelet bases and a few sample sizes. Bottom row, comparison between optimisation curves for $\widehat{B}_J^{(m)}$ and corresponding true Hellinger distances for two sample sizes across different resolution levels. Note that the Hellinger distances are identical for the two methods, as expected.

We believe the practitioner should always inspect the spread of the optimisation curve to discover potential issues like this. A technique like a continuum resolution level may alleviate the problem. It is worth adding that, as can be seen in Subsection 3.4.1, our estimator seems to have faster decay in performance after the optimum than the classical one. This is the reason why we use this optimum resolution level $\hat{J}^{(m)}$ as the upper limit $J$ in (3.5) and leave $J_0$ as few levels below but leave for future research the theoretical reasons behind this phenomenon.

We conclude these examples on best resolution level choice with a typical anisotropic density, Figure 4.2 (d). As can be seen, unlike the issues with locality shown above,



(a) Using symlet 6

(b) Using biortogonal spline 2.8

(c) $\widehat{B}_J^{(m)}$ and $HD$ for $n = 3500$

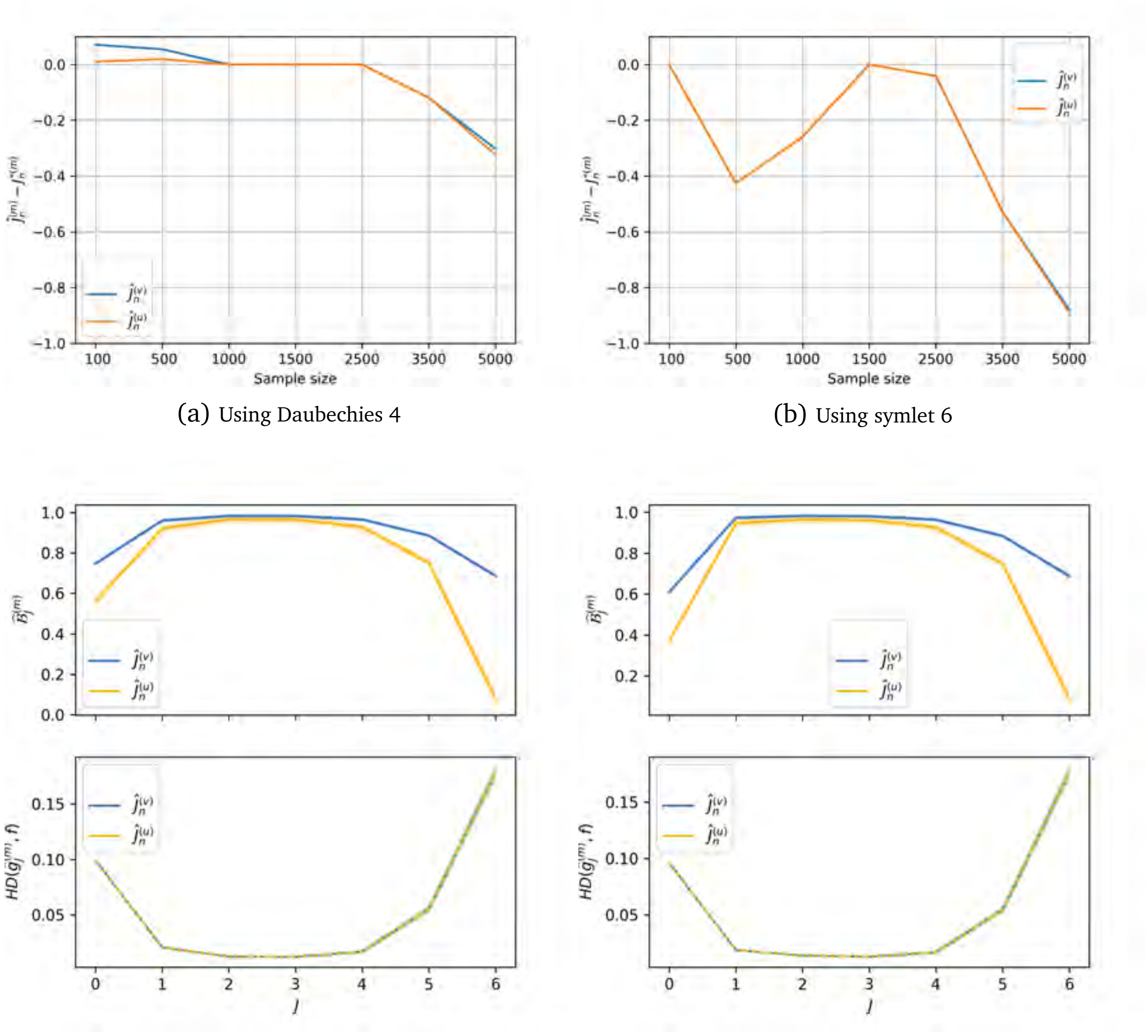
(d) $\widehat{B}_J^{(m)}$ and $HD$ for $n = 3500$

Figure 4.4. Best $J$ difference for Figure 4.2 (b).

the algorithm seems to perform well under the anisotropic case. At $n = 5000$, it seems to have found the right resolution and the $\widehat{B}_J^{(m)}$, $(m) = (v),(u)$, and $HD$ curves seem aligned.

### 4.4.2 Thresholding

Recall that we introduced two optimisation solutions, $\hat{\lambda}_n^{(v)}$ and $\hat{\lambda}_n^{(u)}$, corresponding to the normalised and unnormalised approximation of the BC. We also presented two possible hard-thresholding inequalities, (4.14) and (4.23), corresponding to a *level-dependent* threshold (Donoho and Johnstone, 1996; Delyon and Juditsky, 1996) and our novel threshold involving the estimation of the variance of $\hat{\beta}_{j,z}^{(q)}$. More over, for comparison, we also used the universal threshold $\left|\hat{\beta}_{jz}^{(q)}\right| > \lambda$, originally defined for the classical estimator (Donoho et al, 1995). For the wavelets



(a) Using Daubechies 4

(b) Using symlet 6

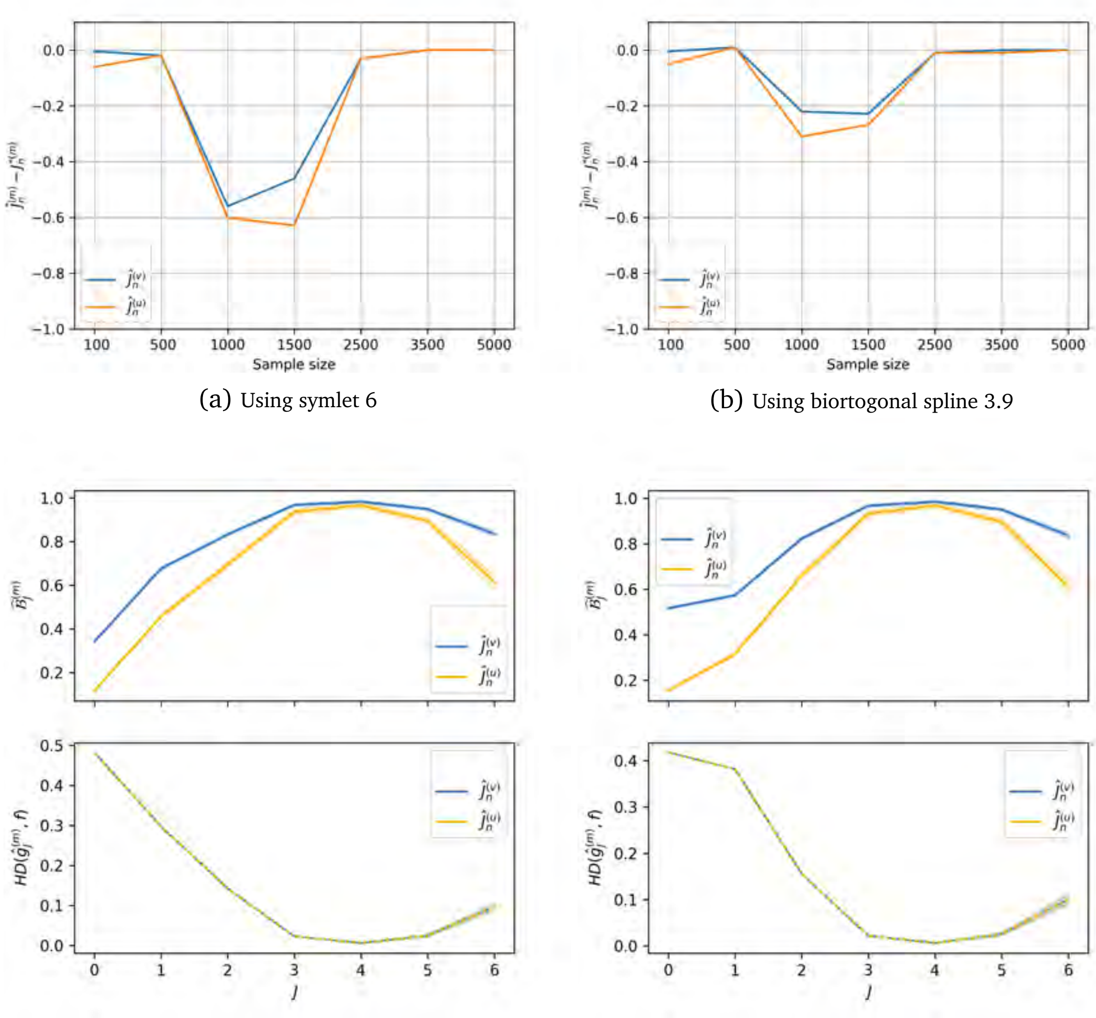
(c) $\widehat{B}_J^{(m)}$ and $HD$ for $n$ = 5000

(d) $\widehat{B}_J^{(m)}$ and $HD$ for $n$ = 5000

Figure 4.5. Best $J$ difference for Figure 4.2 (c).

basis, we used just symlets with 3 and 4 vanishing moments, and Daubechies wavelet with 4 vanishing movements. The number of resolution levels for betas was 1, 2 or 3, *below* the best $J$ calculated by the procedure in previous section. In all, we passed each density and sample size through a battery of $2 \text{x} 3 \text{x} 3 \text{x} 3 = 54$ combinations, which we briefly report below.

Again, we picked sample sizes not following a $2^K$ geometric progression to capture the effect of discretisation imposed by integer resolution levels. Here, we focused this final analysis on two combs and two mixtures, all Gaussian mixtures, in the spirit of Figure 3.1 (a) and (d), in turn inspired by Marron and Wand (1992). Figure 4.7 (a) is a kurtotic, bimodal mixture made out of three Gaussians; (b) is a simple mixture of two Gaussians with different spread; (c) is similar to the claw



(a) Using symlet 6

(b) Using biortogonal spline 3.9

(c) $\widehat{B}_J^{(m)}$ and $HD$ for $n = 5000$

(d) $\widehat{B}_J^{(m)}$ and $HD$ for $n = 5000$

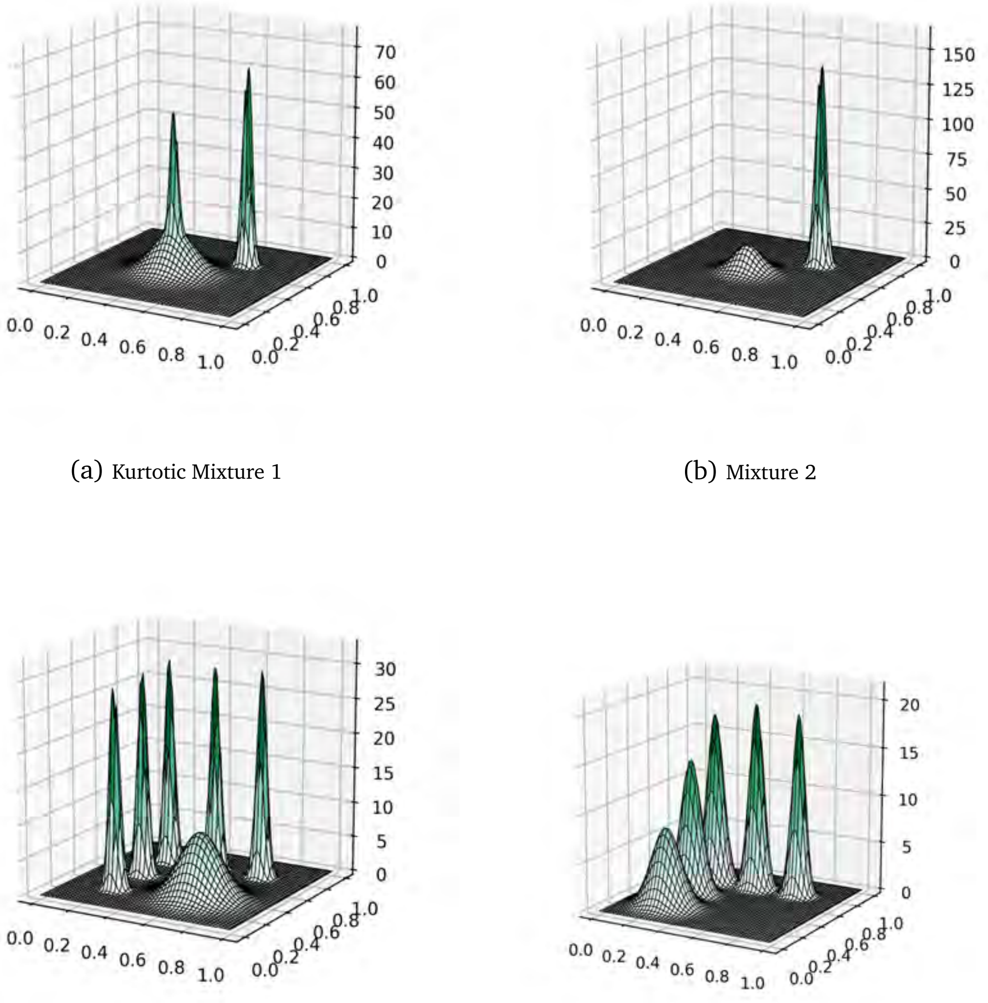
Figure 4.6. Best $J$ difference for Figure 4.2 (d).

density in Marron and Wand (1992); and (d) is akin to the smooth comb there but in 2D. The analytic forms are in the appendix.

For each sample size, we generated 100 samples, we ran the best $J$ algorithm corresponding to the normalised and unnormalised optimisation targets, followed by runs of the 9 combinations of the remaining parameters explained above. For comparison, we ran a multivariate KDE estimator with a Gaussian kernel and a bandwidth (covariance) matrix $H$

$$\hat{f}(x) = \frac{1}{n}\sum_{i=1}^{n} |H|^{-1/2} (2\pi)^{-d/2} e^{-\frac{1}{2}(x-X_i)^T H^{-1}(x-X_i)}.$$

At the end, the HD was calculated using a grid method.



(a) Kurtotic Mixture 1

(b) Mixture 2

(c) 2D Comb 1 (claw)

(d) 2D Smooth comb

Figure 4.7. Densities used in hard threshold simulation study.

Simulation results for these densities, using the Daubechies 4 and Symlet 3 wavelets are summarised in Table 4.1 to Table 4.4. The columns on those are: $n$ is the sample size; $\widehat{\mathcal{B}}_J^{(m)}$ is the Bhattacharyya formula used, $(m) = (v), (u)$; $\Delta J$ is the number of levels below $\hat{J}_n^{(m)}$; KDE is the median HD (approximated as explained above) for the KDE estimator with covariance $H$ calculated via CV using maximum likelihood (Seabold and Perktold, 2010); $\lambda$, $\lambda\sqrt{\Delta J}$ and $\lambda\sigma^B$ are the different hard-threshold algorithms, corresponding to the simple universal threshold $\left|\hat{\beta}_{jz}^{(q)}\right| > \lambda$, (4.14) and (4.23) respectively. For each of these thresholding algorithms, first quantile $Q_1$, median $Med$ and $Q3$ are reported. We have used bold face to highlight the cases in which the median HD for these estimators is less than or equal to the corresponding median for the KDE. Note that we are comparing the kernel estimator, which does not drop any observation and in essence has $n$ free parameters, against our method that aims to reduce the number of parameters using thresholding. Reducing the number of free parameters without compromising performance is a key requirement in big data applications. So, the fact that our algorithm achieves similar performance in several cases is noteworthy.

We start our analysis by highlighting some facts in Table 4.1 for the kurtotic mixture (Figure 4.7 (a)). In these experiments as well as in the others we performed, one observes a slight decrease in performance by using more than $1$ level of delta estimators but it is always within margin of error. Other thresholding options like soft and block thresholding are worth experimenting with to see if this phenomenon is different. Indeed, in our real data analysis, without any *a priori* knowledge of the truth, it seemed $2$ levels produced better density estimates (see next section). We note also that in this experiment and the others, our novel hard thersholding of formula (4.23) performed better than the other well-known strategies. Finally, for this kurtotic mixture, the performance of our estimator is better towards the big sample sizes[3].

Gaussian Mixture 2 (Figure 4.7 (b)) is an interesting case where one can see our non linear estimator performing better than KDE for small sample sizes (Table 4.2). We can add, for instance, that for $n = 250$, $\Delta J = 1$ the median of number of coefficients was $187$ and $183$ for the normed and unnormed algorithms respectively (see supplementary Table B.2).

In Table 4.3, 2D Comb (claw) (Figure 4.7 (c)), we see a similar phenomenon to Table 4.1 but here using Symlet with 3 vanishing moments. In this case and the next one, less regularity is better. See the plots for all these tables in Appendix B. Note that in many cases, the KDE median estimate is outside the $Q1 - Q3$ range. In fact, it is quite remarkable that at $n = 6000$ one can get a HD close to or better than the KDE by using less that $500$ coefficients.

[3] The fact that wavelet methods are showing their advantages at larger sample sizes and high signal-to-noise ratios is well documented in the literature (see, e.g., Hall et al. (2018)) and we observe this phenomenon being confirmed in our present simulations.

| **n** | $\widehat{\mathcal{B}}_J^{(m)}$ | $\Delta J$ | $\lambda$ | | | $\lambda\sqrt{\Delta J}$ | | | $\lambda\,\sigma^B$ | | | KDE |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | $Q_1$ | $Med$ | $Q_3$ | $Q_1$ | $Med$ | $Q_3$ | $Q_1$ | $Med$ | $Q_3$ | |
| 250 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0805 | 0.0881 | 0.0952 | 0.0805 | 0.0881 | 0.0952 | 0.0810 | 0.0889 | 0.0962 | 0.0410 |
| | | 2 | 0.0812 | 0.0871 | 0.0948 | 0.0828 | 0.0885 | 0.0960 | 0.0820 | 0.0898 | 0.0983 | |
| | | 3 | 0.0816 | 0.0871 | 0.0944 | 0.0828 | 0.0879 | 0.0942 | 0.0820 | 0.0898 | 0.0983 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0845 | 0.0904 | 0.0966 | 0.0845 | 0.0904 | 0.0966 | 0.0859 | 0.0924 | 0.0963 | |
| | | 2 | 0.0838 | 0.0882 | 0.0972 | 0.0842 | 0.0891 | 0.0971 | 0.0868 | 0.0927 | 0.0989 | |
| | | 3 | 0.0838 | 0.0879 | 0.0952 | 0.0836 | 0.0879 | 0.0940 | 0.0868 | 0.0927 | 0.0990 | |
| 500 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0374 | 0.0447 | 0.0781 | 0.0374 | 0.0447 | 0.0781 | 0.0315 | 0.0396 | 0.0785 | 0.0294 |
| | | 2 | 0.0427 | 0.0514 | 0.0775 | 0.0464 | 0.0583 | 0.0775 | 0.0340 | 0.0407 | 0.0785 | |
| | | 3 | 0.0438 | 0.0523 | 0.0776 | 0.0482 | 0.0591 | 0.0776 | 0.0341 | 0.0407 | 0.0786 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0387 | 0.0772 | 0.0810 | 0.0387 | 0.0772 | 0.0810 | 0.0340 | 0.0778 | 0.0807 | |
| | | 2 | 0.0467 | 0.0767 | 0.0809 | 0.0512 | 0.0771 | 0.0810 | 0.0367 | 0.0781 | 0.0810 | |
| | | 3 | 0.0481 | 0.0768 | 0.0809 | 0.0518 | 0.0768 | 0.0810 | 0.0367 | 0.0781 | 0.0810 | |
| 1000 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0215 | 0.0239 | 0.0275 | 0.0215 | 0.0239 | 0.0275 | 0.0186 | 0.0203 | 0.0229 | 0.0199 |
| | | 2 | 0.0226 | 0.0255 | 0.0299 | 0.0241 | 0.0282 | 0.0326 | 0.0192 | 0.0214 | 0.0241 | |
| | | 3 | 0.0242 | 0.0264 | 0.0309 | 0.0261 | 0.0295 | 0.0336 | 0.0192 | 0.0215 | 0.0242 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0218 | 0.0239 | 0.0271 | 0.0218 | 0.0239 | 0.0271 | 0.0187 | 0.0205 | 0.0230 | |
| | | 2 | 0.0229 | 0.0258 | 0.0302 | 0.0248 | 0.0287 | 0.0331 | 0.0192 | 0.0216 | 0.0241 | |
| | | 3 | 0.0243 | 0.0274 | 0.0310 | 0.0265 | 0.0303 | 0.0339 | 0.0193 | 0.0217 | 0.0242 | |
| 1500 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0154 | 0.0174 | 0.0198 | 0.0154 | 0.0174 | 0.0198 | 0.0139 | **0.0156** | 0.0171 | 0.0157 |
| | | 2 | 0.0166 | 0.0186 | 0.0211 | 0.0176 | 0.0200 | 0.0226 | 0.0144 | 0.0164 | 0.0178 | |
| | | 3 | 0.0172 | 0.0192 | 0.0220 | 0.0179 | 0.0205 | 0.0231 | 0.0145 | 0.0164 | 0.0179 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0152 | 0.0174 | 0.0198 | 0.0152 | 0.0174 | 0.0198 | 0.0139 | **0.0153** | 0.0169 | |
| | | 2 | 0.0165 | 0.0186 | 0.0211 | 0.0174 | 0.0200 | 0.0226 | 0.0143 | 0.0163 | 0.0176 | |
| | | 3 | 0.0171 | 0.0192 | 0.0220 | 0.0179 | 0.0205 | 0.0231 | 0.0144 | 0.0163 | 0.0176 | |
| 2000 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0133 | 0.0145 | 0.0165 | 0.0133 | 0.0145 | 0.0165 | 0.0119 | **0.0132** | 0.0145 | 0.0136 |
| | | 2 | 0.0139 | 0.0151 | 0.0169 | 0.0141 | 0.0163 | 0.0175 | 0.0121 | **0.0133** | 0.0151 | |
| | | 3 | 0.0143 | 0.0156 | 0.0177 | 0.0148 | 0.0164 | 0.0180 | 0.0122 | **0.0134** | 0.0151 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0133 | 0.0145 | 0.0164 | 0.0133 | 0.0145 | 0.0164 | 0.0119 | **0.0131** | 0.0144 | |
| | | 2 | 0.0140 | 0.0151 | 0.0169 | 0.0144 | 0.0163 | 0.0175 | 0.0121 | **0.0133** | 0.0150 | |
| | | 3 | 0.0143 | 0.0156 | 0.0174 | 0.0148 | 0.0164 | 0.0179 | 0.0122 | **0.0134** | 0.0150 | |
| 3000 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0105 | 0.0110 | 0.0120 | 0.0105 | 0.0110 | 0.0120 | 0.0095 | **0.0100** | 0.0107 | 0.0109 |
| | | 2 | 0.0107 | 0.0116 | 0.0124 | 0.0109 | 0.0121 | 0.0130 | 0.0097 | **0.0103** | 0.0110 | |
| | | 3 | 0.0108 | 0.0116 | 0.0127 | 0.0111 | 0.0122 | 0.0132 | 0.0097 | **0.0103** | 0.0110 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0105 | 0.0110 | 0.0120 | 0.0105 | 0.0110 | 0.0120 | 0.0094 | **0.0100** | 0.0107 | |
| | | 2 | 0.0107 | 0.0116 | 0.0124 | 0.0110 | 0.0121 | 0.0129 | 0.0097 | **0.0103** | 0.0111 | |
| | | 3 | 0.0108 | 0.0117 | 0.0126 | 0.0111 | 0.0122 | 0.0133 | 0.0097 | **0.0104** | 0.0110 | |
| 4000 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0086 | **0.0091** | 0.0099 | 0.0086 | **0.0091** | 0.0099 | 0.0081 | **0.0087** | 0.0092 | 0.0094 |
| | | 2 | 0.0091 | 0.0095 | 0.0105 | 0.0094 | 0.0097 | 0.0107 | 0.0082 | **0.0087** | 0.0094 | |
| | | 3 | 0.0091 | 0.0096 | 0.0106 | 0.0094 | 0.0100 | 0.0109 | 0.0082 | **0.0087** | 0.0094 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0087 | **0.0092** | 0.0101 | 0.0087 | **0.0092** | 0.0101 | 0.0081 | **0.0087** | 0.0092 | |
| | | 2 | 0.0091 | 0.0096 | 0.0106 | 0.0094 | 0.0098 | 0.0108 | 0.0081 | **0.0088** | 0.0093 | |
| | | 3 | 0.0091 | 0.0097 | 0.0107 | 0.0095 | 0.0100 | 0.0112 | 0.0082 | **0.0088** | 0.0093 | |
| 6000 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0068 | 0.0075 | 0.0081 | 0.0068 | 0.0075 | 0.0081 | 0.0065 | **0.0070** | 0.0077 | 0.0072 |
| | | 2 | 0.0070 | 0.0076 | 0.0082 | 0.0074 | 0.0079 | 0.0085 | 0.0067 | **0.0072** | 0.0077 | |
| | | 3 | 0.0073 | 0.0077 | 0.0080 | 0.0077 | 0.0081 | 0.0088 | 0.0067 | **0.0072** | 0.0077 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0067 | 0.0073 | 0.0078 | 0.0067 | 0.0073 | 0.0078 | 0.0064 | **0.0069** | 0.0073 | |
| | | 2 | 0.0068 | 0.0074 | 0.0080 | 0.0071 | 0.0076 | 0.0084 | 0.0065 | **0.0070** | 0.0074 | |
| | | 3 | 0.0069 | 0.0074 | 0.0080 | 0.0072 | 0.0077 | 0.0085 | 0.0065 | **0.0070** | 0.0074 | |

Table 4.1. (Approximated) Mean Squared Hellinger Distance (MSHD) for the various non linear estimator algorithms using the Daubechie 4 wavelet for the density Kurtotic Mixture 1 (Figure 4.7 (a)). See text for column descriptions. Corresponding number of coefficients found in table B.1

The 2D smooth comb (Figure 4.7 (d)) turned out to be somewhat disappointing. Again, we saw good performance of the estimators built using Symlet 3 for small sample sizes - although KDE tended to be inside the $Q1-Q3$ quantile range, which arguably puts it at the same level. As the sample size increases, the estimators' performance improves as expected, but it cannot keep with KDE which seems to

| n | $\widehat{\mathcal{B}}_J^{(m)}$ | $\Delta J$ | $\lambda$ | | | $\lambda\sqrt{\Delta J}$ | | | $\lambda\sigma^B$ | | | KDE |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | $Q_1$ | $Med$ | $Q_3$ | $Q_1$ | $Med$ | $Q_3$ | $Q_1$ | $Med$ | $Q_3$ | |
| 250 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0365 | 0.0401 | 0.0479 | 0.0365 | 0.0401 | 0.0479 | 0.0250 | **0.0300** | 0.0344 | 0.0351 |
| | | 2 | 0.0426 | 0.0483 | 0.0538 | 0.0456 | 0.0510 | 0.0596 | 0.0289 | **0.0326** | 0.0377 | |
| | | 3 | 0.0441 | 0.0496 | 0.0562 | 0.0460 | 0.0527 | 0.0621 | 0.0295 | **0.0336** | 0.0386 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0365 | 0.0399 | 0.0463 | 0.0365 | 0.0399 | 0.0463 | 0.0246 | **0.0289** | 0.0340 | |
| | | 2 | 0.0421 | 0.0483 | 0.0545 | 0.0456 | 0.0510 | 0.0597 | 0.0290 | **0.0326** | 0.0372 | |
| | | 3 | 0.0447 | 0.0502 | 0.0562 | 0.0468 | 0.0527 | 0.0625 | 0.0292 | **0.0348** | 0.0380 | |
| 500 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0231 | 0.0267 | 0.0329 | 0.0231 | 0.0267 | 0.0329 | 0.0176 | **0.0206** | 0.0238 | 0.0247 |
| | | 2 | 0.0254 | 0.0298 | 0.0346 | 0.0259 | 0.0320 | 0.0370 | 0.0185 | **0.0223** | 0.0265 | |
| | | 3 | 0.0253 | 0.0302 | 0.0355 | 0.0263 | 0.0306 | 0.0377 | 0.0188 | **0.0223** | 0.0269 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0230 | 0.0267 | 0.0326 | 0.0230 | 0.0267 | 0.0326 | 0.0176 | **0.0202** | 0.0235 | |
| | | 2 | 0.0251 | 0.0298 | 0.0353 | 0.0266 | 0.0320 | 0.0372 | 0.0185 | **0.0223** | 0.0262 | |
| | | 3 | 0.0264 | 0.0302 | 0.0355 | 0.0263 | 0.0318 | 0.0379 | 0.0190 | **0.0223** | 0.0269 | |
| 1000 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0149 | **0.0159** | 0.0183 | 0.0149 | **0.0159** | 0.0183 | 0.0128 | **0.0140** | 0.0155 | 0.0163 |
| | | 2 | 0.0153 | 0.0169 | 0.0195 | 0.0156 | 0.0177 | 0.0198 | 0.0132 | **0.0146** | 0.0162 | |
| | | 3 | 0.0156 | 0.0171 | 0.0198 | 0.0157 | 0.0179 | 0.0207 | 0.0132 | **0.0146** | 0.0162 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0149 | **0.0159** | 0.0184 | 0.0149 | **0.0159** | 0.0184 | 0.0128 | **0.0140** | 0.0155 | |
| | | 2 | 0.0154 | 0.0169 | 0.0196 | 0.0158 | 0.0179 | 0.0198 | 0.0132 | **0.0146** | 0.0162 | |
| | | 3 | 0.0156 | 0.0171 | 0.0199 | 0.0158 | 0.0180 | 0.0207 | 0.0132 | **0.0146** | 0.0162 | |
| 1500 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0114 | **0.0124** | 0.0141 | 0.0114 | **0.0124** | 0.0141 | 0.0102 | **0.0114** | 0.0126 | 0.0128 |
| | | 2 | 0.0119 | 0.0130 | 0.0147 | 0.0123 | 0.0136 | 0.0155 | 0.0105 | **0.0115** | 0.0129 | |
| | | 3 | 0.0118 | 0.0130 | 0.0150 | 0.0123 | 0.0136 | 0.0157 | 0.0105 | **0.0115** | 0.0129 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0114 | **0.0124** | 0.0141 | 0.0114 | **0.0124** | 0.0141 | 0.0102 | **0.0114** | 0.0126 | |
| | | 2 | 0.0119 | 0.0130 | 0.0147 | 0.0123 | 0.0136 | 0.0155 | 0.0105 | **0.0115** | 0.0129 | |
| | | 3 | 0.0118 | 0.0130 | 0.0150 | 0.0123 | 0.0133 | 0.0157 | 0.0105 | **0.0115** | 0.0129 | |
| 2000 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0098 | **0.0104** | 0.0113 | 0.0098 | **0.0104** | 0.0113 | 0.0090 | **0.0097** | 0.0107 | 0.0110 |
| | | 2 | 0.0103 | 0.0111 | 0.0120 | 0.0105 | 0.0113 | 0.0124 | 0.0090 | **0.0099** | 0.0108 | |
| | | 3 | 0.0101 | 0.0111 | 0.0121 | 0.0103 | 0.0112 | 0.0125 | 0.0090 | **0.0099** | 0.0108 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0099 | **0.0104** | 0.0113 | 0.0099 | **0.0104** | 0.0113 | 0.0090 | **0.0097** | 0.0107 | |
| | | 2 | 0.0103 | 0.0111 | 0.0120 | 0.0105 | 0.0113 | 0.0124 | 0.0090 | **0.0099** | 0.0108 | |
| | | 3 | 0.0101 | 0.0111 | 0.0121 | 0.0104 | 0.0112 | 0.0125 | 0.0090 | **0.0099** | 0.0108 | |
| 3000 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0078 | **0.0087** | 0.0096 | 0.0078 | **0.0087** | 0.0096 | 0.0075 | **0.0084** | 0.0091 | 0.0087 |
| | | 2 | 0.0082 | 0.0090 | 0.0099 | 0.0083 | 0.0090 | 0.0100 | 0.0075 | **0.0085** | 0.0091 | |
| | | 3 | 0.0081 | 0.0089 | 0.0097 | 0.0082 | 0.0091 | 0.0098 | 0.0075 | **0.0085** | 0.0091 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0078 | **0.0087** | 0.0096 | 0.0078 | **0.0087** | 0.0096 | 0.0075 | **0.0084** | 0.0090 | |
| | | 2 | 0.0082 | 0.0090 | 0.0099 | 0.0083 | 0.0090 | 0.0100 | 0.0075 | **0.0085** | 0.0091 | |
| | | 3 | 0.0081 | 0.0089 | 0.0097 | 0.0082 | 0.0091 | 0.0098 | 0.0075 | **0.0085** | 0.0091 | |
| 4000 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0069 | **0.0072** | 0.0080 | 0.0069 | **0.0072** | 0.0080 | 0.0065 | **0.0070** | 0.0076 | 0.0073 |
| | | 2 | 0.0070 | 0.0074 | 0.0080 | 0.0071 | 0.0075 | 0.0082 | 0.0066 | **0.0071** | 0.0076 | |
| | | 3 | 0.0070 | **0.0073** | 0.0080 | 0.0071 | 0.0074 | 0.0081 | 0.0066 | **0.0071** | 0.0076 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0069 | **0.0072** | 0.0080 | 0.0069 | **0.0072** | 0.0080 | 0.0065 | **0.0070** | 0.0076 | |
| | | 2 | 0.0070 | 0.0074 | 0.0080 | 0.0072 | 0.0075 | 0.0082 | 0.0066 | **0.0071** | 0.0076 | |
| | | 3 | 0.0070 | 0.0074 | 0.0081 | 0.0071 | 0.0075 | 0.0082 | 0.0066 | **0.0071** | 0.0076 | |
| 6000 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0059 | 0.0062 | 0.0065 | 0.0059 | 0.0062 | 0.0065 | 0.0057 | 0.0061 | 0.0064 | 0.0057 |
| | | 2 | 0.0060 | 0.0063 | 0.0066 | 0.0061 | 0.0064 | 0.0068 | 0.0057 | 0.0061 | 0.0064 | |
| | | 3 | 0.0060 | 0.0064 | 0.0066 | 0.0061 | 0.0065 | 0.0067 | 0.0057 | 0.0061 | 0.0064 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0059 | 0.0062 | 0.0065 | 0.0059 | 0.0062 | 0.0065 | 0.0057 | 0.0061 | 0.0064 | |
| | | 2 | 0.0060 | 0.0063 | 0.0066 | 0.0061 | 0.0064 | 0.0068 | 0.0057 | 0.0061 | 0.0064 | |
| | | 3 | 0.0060 | 0.0064 | 0.0066 | 0.0061 | 0.0065 | 0.0067 | 0.0057 | 0.0061 | 0.0064 | |

Table 4.2. (Approximated) Mean Squared Hellinger Distance (MSHD) for the various non linear estimator algorithms using the Daubechie 4 wavelet for the density Mixture 2 (Figure 4.7 (b)). See text for column descriptions. Corresponding number of coefficients found in table B.2

improve faster. Despite this, we stress on the fact that the discrepancy in the risk values is non significant. Maybe this could reappear if we increase the sample size and the best $J$ level hits a sweet spot of good performance, but of course we are not sure. For more on this, specially in regards to potential improvements, visit our discussion in Chapter 6. Nonetheless, it is interesting to note that one achieves

| n | $\widehat{\mathcal{B}}_J^{(m)}$ | $\Delta J$ | $\lambda$ | | | $\lambda\sqrt{\Delta J}$ | | | $\lambda\sigma^B$ | | | KDE |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | $Q_1$ | $Med$ | $Q_3$ | $Q_1$ | $Med$ | $Q_3$ | $Q_1$ | $Med$ | $Q_3$ | |
| 250 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0609 | 0.0664 | 0.0714 | 0.0609 | 0.0664 | 0.0714 | 0.0629 | 0.0671 | 0.0749 | 0.0574 |
| | | 2 | 0.0622 | 0.0658 | 0.0720 | 0.0623 | 0.0651 | 0.0718 | 0.0634 | 0.0693 | 0.0754 | |
| | | 3 | 0.0610 | 0.0656 | 0.0708 | 0.0603 | 0.0652 | 0.0691 | 0.0634 | 0.0693 | 0.0753 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0609 | 0.0664 | 0.0718 | 0.0609 | 0.0664 | 0.0718 | 0.0630 | 0.0676 | 0.0752 | |
| | | 2 | 0.0620 | 0.0658 | 0.0725 | 0.0623 | 0.0655 | 0.0718 | 0.0644 | 0.0695 | 0.0754 | |
| | | 3 | 0.0606 | 0.0651 | 0.0708 | 0.0603 | 0.0656 | 0.0695 | 0.0646 | 0.0695 | 0.0754 | |
| 500 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0497 | 0.0523 | 0.0564 | 0.0497 | 0.0523 | 0.0564 | 0.0498 | 0.0530 | 0.0571 | 0.0386 |
| | | 2 | 0.0501 | 0.0531 | 0.0559 | 0.0503 | 0.0533 | 0.0564 | 0.0504 | 0.0533 | 0.0571 | |
| | | 3 | 0.0499 | 0.0528 | 0.0553 | 0.0499 | 0.0524 | 0.0554 | 0.0506 | 0.0534 | 0.0571 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0501 | 0.0528 | 0.0567 | 0.0501 | 0.0528 | 0.0567 | 0.0504 | 0.0537 | 0.0572 | |
| | | 2 | 0.0503 | 0.0532 | 0.0560 | 0.0504 | 0.0533 | 0.0559 | 0.0509 | 0.0537 | 0.0571 | |
| | | 3 | 0.0499 | 0.0528 | 0.0557 | 0.0499 | 0.0523 | 0.0551 | 0.0509 | 0.0537 | 0.0571 | |
| 1000 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0233 | 0.0266 | 0.0334 | 0.0233 | 0.0266 | 0.0334 | 0.0210 | **0.0249** | 0.0300 | 0.0257 |
| | | 2 | 0.0259 | 0.0299 | 0.0361 | 0.0275 | 0.0322 | 0.0398 | 0.0222 | 0.0262 | 0.0314 | |
| | | 3 | 0.0269 | 0.0307 | 0.0388 | 0.0301 | 0.0344 | 0.0421 | 0.0224 | 0.0263 | 0.0315 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0234 | 0.0281 | 0.0451 | 0.0234 | 0.0281 | 0.0451 | 0.0216 | 0.0261 | 0.0452 | |
| | | 2 | 0.0261 | 0.0305 | 0.0450 | 0.0276 | 0.0330 | 0.0449 | 0.0227 | 0.0270 | 0.0454 | |
| | | 3 | 0.0272 | 0.0315 | 0.0450 | 0.0306 | 0.0349 | 0.0449 | 0.0228 | 0.0270 | 0.0454 | |
| 1500 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0170 | **0.0190** | 0.0214 | 0.0170 | **0.0190** | 0.0214 | 0.0161 | **0.0177** | 0.0197 | 0.0205 |
| | | 2 | 0.0185 | 0.0207 | 0.0224 | 0.0202 | 0.0222 | 0.0246 | 0.0168 | **0.0183** | 0.0199 | |
| | | 3 | 0.0197 | 0.0215 | 0.0240 | 0.0216 | 0.0239 | 0.0262 | 0.0167 | **0.0183** | 0.0202 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0170 | **0.0189** | 0.0214 | 0.0170 | **0.0189** | 0.0214 | 0.0160 | **0.0177** | 0.0200 | |
| | | 2 | 0.0185 | 0.0207 | 0.0226 | 0.0201 | 0.0222 | 0.0246 | 0.0167 | **0.0183** | 0.0203 | |
| | | 3 | 0.0197 | 0.0216 | 0.0240 | 0.0218 | 0.0239 | 0.0266 | 0.0167 | **0.0182** | 0.0202 | |
| 2000 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0145 | **0.0161** | 0.0177 | 0.0145 | **0.0161** | 0.0177 | 0.0135 | **0.0147** | 0.0159 | 0.0174 |
| | | 2 | 0.0158 | 0.0178 | 0.0190 | 0.0170 | 0.0189 | 0.0204 | 0.0140 | **0.0149** | 0.0163 | |
| | | 3 | 0.0169 | 0.0185 | 0.0197 | 0.0182 | 0.0198 | 0.0220 | 0.0140 | **0.0149** | 0.0163 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0145 | **0.0161** | 0.0176 | 0.0145 | **0.0161** | 0.0176 | 0.0135 | **0.0147** | 0.0157 | |
| | | 2 | 0.0157 | 0.0175 | 0.0188 | 0.0170 | 0.0189 | 0.0205 | 0.0140 | **0.0149** | 0.0163 | |
| | | 3 | 0.0169 | 0.0184 | 0.0198 | 0.0182 | 0.0200 | 0.0221 | 0.0139 | **0.0149** | 0.0164 | |
| 3000 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0119 | **0.0126** | 0.0138 | 0.0119 | **0.0126** | 0.0138 | 0.0112 | **0.0120** | 0.0129 | 0.0138 |
| | | 2 | 0.0127 | **0.0135** | 0.0143 | 0.0133 | 0.0142 | 0.0150 | 0.0114 | **0.0123** | 0.0130 | |
| | | 3 | 0.0133 | 0.0139 | 0.0150 | 0.0138 | 0.0146 | 0.0158 | 0.0115 | **0.0123** | 0.0130 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0119 | **0.0126** | 0.0136 | 0.0119 | **0.0126** | 0.0136 | 0.0111 | **0.0119** | 0.0128 | |
| | | 2 | 0.0127 | **0.0133** | 0.0143 | 0.0133 | 0.0142 | 0.0149 | 0.0112 | **0.0122** | 0.0129 | |
| | | 3 | 0.0133 | 0.0139 | 0.0149 | 0.0138 | 0.0146 | 0.0158 | 0.0113 | **0.0123** | 0.0130 | |
| 4000 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0105 | **0.0113** | 0.0119 | 0.0105 | **0.0113** | 0.0119 | 0.0098 | **0.0105** | 0.0110 | 0.0115 |
| | | 2 | 0.0112 | 0.0118 | 0.0127 | 0.0115 | 0.0121 | 0.0131 | 0.0101 | **0.0107** | 0.0114 | |
| | | 3 | 0.0115 | 0.0120 | 0.0129 | 0.0117 | 0.0125 | 0.0132 | 0.0101 | **0.0107** | 0.0114 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0105 | **0.0113** | 0.0119 | 0.0105 | **0.0113** | 0.0119 | 0.0098 | **0.0105** | 0.0110 | |
| | | 2 | 0.0112 | 0.0118 | 0.0127 | 0.0115 | 0.0121 | 0.0131 | 0.0101 | **0.0107** | 0.0114 | |
| | | 3 | 0.0115 | 0.0120 | 0.0129 | 0.0117 | 0.0125 | 0.0132 | 0.0101 | **0.0107** | 0.0114 | |
| 6000 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0091 | 0.0093 | 0.0098 | 0.0091 | 0.0093 | 0.0098 | 0.0083 | **0.0088** | 0.0091 | 0.0092 |
| | | 2 | 0.0093 | 0.0097 | 0.0100 | 0.0093 | 0.0098 | 0.0102 | 0.0085 | **0.0088** | 0.0092 | |
| | | 3 | 0.0093 | 0.0097 | 0.0101 | 0.0094 | 0.0099 | 0.0104 | 0.0085 | **0.0088** | 0.0092 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0089 | 0.0093 | 0.0098 | 0.0089 | 0.0093 | 0.0098 | 0.0082 | **0.0087** | 0.0090 | |
| | | 2 | 0.0092 | 0.0097 | 0.0100 | 0.0093 | 0.0098 | 0.0103 | 0.0083 | **0.0088** | 0.0092 | |
| | | 3 | 0.0092 | 0.0097 | 0.0103 | 0.0093 | 0.0099 | 0.0104 | 0.0084 | **0.0088** | 0.0091 | |

Table 4.3. (Approximated) Mean Squared Hellinger Distance (MSHD) for the various non linear estimator algorithms using the Symlet 3 wavelet for the density 2D Comb 1 (claw) (Figure 4.7 (c)). See text for column descriptions. Corresponding number of coefficients found in table B.3

a remarkably low HD using as little as 163 coefficients[4].

[4] 163 is the median of number of coefficients for the case $n = 6000$, $\widehat{\mathcal{B}}_J^{(u)}$, $\Delta J = 1$, $\lambda\sigma^B$ in the table. See complementary results in Table B.4

| **n** | $\widehat{\mathcal{B}}_J^{(m)}$ | $\Delta J$ | $\lambda$ | | | $\lambda\sqrt{\Delta J}$ | | | $\lambda\,\sigma^B$ | | | KDE |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | $Q_1$ | $Med$ | $Q_3$ | $Q_1$ | $Med$ | $Q_3$ | $Q_1$ | $Med$ | $Q_3$ | |
| 250 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0241 | **0.0282** | 0.0347 | 0.0241 | **0.0282** | 0.0347 | 0.0221 | **0.0265** | 0.0323 | 0.0312 |
| | | 2 | 0.0244 | **0.0283** | 0.0348 | 0.0249 | **0.0296** | 0.0351 | 0.0224 | **0.0277** | 0.0336 | |
| | | 3 | 0.0245 | **0.0281** | 0.0336 | 0.0244 | **0.0282** | 0.0324 | 0.0224 | **0.0277** | 0.0336 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0241 | **0.0283** | 0.0351 | 0.0241 | **0.0283** | 0.0351 | 0.0217 | **0.0265** | 0.0322 | |
| | | 2 | 0.0248 | **0.0286** | 0.0349 | 0.0248 | **0.0294** | 0.0354 | 0.0223 | **0.0277** | 0.0335 | |
| | | 3 | 0.0246 | **0.0280** | 0.0336 | 0.0244 | **0.0278** | 0.0325 | 0.0223 | **0.0277** | 0.0335 | |
| 500 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0176 | **0.0196** | 0.0229 | 0.0176 | **0.0196** | 0.0229 | 0.0159 | **0.0186** | 0.0206 | 0.0207 |
| | | 2 | 0.0176 | **0.0199** | 0.0229 | 0.0177 | **0.0200** | 0.0232 | 0.0161 | **0.0190** | 0.0213 | |
| | | 3 | 0.0172 | **0.0194** | 0.0230 | 0.0170 | **0.0196** | 0.0224 | 0.0161 | **0.0190** | 0.0213 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0176 | **0.0199** | 0.0231 | 0.0176 | **0.0199** | 0.0231 | 0.0159 | **0.0186** | 0.0206 | |
| | | 2 | 0.0177 | **0.0199** | 0.0229 | 0.0178 | **0.0200** | 0.0232 | 0.0161 | **0.0190** | 0.0214 | |
| | | 3 | 0.0172 | **0.0194** | 0.0230 | 0.0171 | **0.0196** | 0.0224 | 0.0161 | **0.0190** | 0.0214 | |
| 1000 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0126 | **0.0132** | 0.0148 | 0.0126 | **0.0132** | 0.0148 | 0.0122 | **0.0132** | 0.0144 | 0.0138 |
| | | 2 | 0.0125 | **0.0132** | 0.0146 | 0.0126 | **0.0133** | 0.0146 | 0.0125 | **0.0133** | 0.0147 | |
| | | 3 | 0.0124 | **0.0132** | 0.0146 | 0.0124 | **0.0131** | 0.0144 | 0.0125 | **0.0133** | 0.0147 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0126 | **0.0132** | 0.0148 | 0.0126 | **0.0132** | 0.0148 | 0.0122 | **0.0132** | 0.0144 | |
| | | 2 | 0.0125 | **0.0132** | 0.0146 | 0.0126 | **0.0133** | 0.0146 | 0.0125 | **0.0133** | 0.0147 | |
| | | 3 | 0.0124 | **0.0132** | 0.0146 | 0.0124 | **0.0131** | 0.0144 | 0.0125 | **0.0133** | 0.0147 | |
| 1500 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0111 | 0.0118 | 0.0127 | 0.0111 | 0.0118 | 0.0127 | 0.0110 | 0.0118 | 0.0126 | 0.0109 |
| | | 2 | 0.0110 | 0.0117 | 0.0126 | 0.0110 | 0.0118 | 0.0126 | 0.0110 | 0.0118 | 0.0127 | |
| | | 3 | 0.0111 | 0.0117 | 0.0126 | 0.0110 | 0.0118 | 0.0126 | 0.0110 | 0.0118 | 0.0127 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0111 | 0.0118 | 0.0127 | 0.0111 | 0.0118 | 0.0127 | 0.0110 | 0.0118 | 0.0126 | |
| | | 2 | 0.0110 | 0.0117 | 0.0126 | 0.0110 | 0.0118 | 0.0126 | 0.0111 | 0.0118 | 0.0127 | |
| | | 3 | 0.0111 | 0.0117 | 0.0126 | 0.0110 | 0.0118 | 0.0126 | 0.0111 | 0.0118 | 0.0127 | |
| 2000 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0105 | 0.0110 | 0.0118 | 0.0105 | 0.0110 | 0.0118 | 0.0105 | 0.0111 | 0.0118 | 0.0092 |
| | | 2 | 0.0105 | 0.0110 | 0.0117 | 0.0104 | 0.0109 | 0.0117 | 0.0106 | 0.0111 | 0.0119 | |
| | | 3 | 0.0105 | 0.0109 | 0.0116 | 0.0104 | 0.0109 | 0.0117 | 0.0106 | 0.0111 | 0.0119 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0105 | 0.0110 | 0.0118 | 0.0105 | 0.0110 | 0.0118 | 0.0105 | 0.0111 | 0.0118 | |
| | | 2 | 0.0104 | 0.0110 | 0.0117 | 0.0105 | 0.0109 | 0.0117 | 0.0106 | 0.0111 | 0.0119 | |
| | | 3 | 0.0105 | 0.0109 | 0.0116 | 0.0104 | 0.0109 | 0.0117 | 0.0106 | 0.0111 | 0.0119 | |
| 3000 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0097 | 0.0100 | 0.0104 | 0.0097 | 0.0100 | 0.0104 | 0.0097 | 0.0100 | 0.0104 | 0.0073 |
| | | 2 | 0.0096 | 0.0100 | 0.0103 | 0.0096 | 0.0100 | 0.0104 | 0.0097 | 0.0101 | 0.0105 | |
| | | 3 | 0.0096 | 0.0100 | 0.0104 | 0.0096 | 0.0099 | 0.0103 | 0.0097 | 0.0101 | 0.0105 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0097 | 0.0100 | 0.0104 | 0.0097 | 0.0100 | 0.0104 | 0.0097 | 0.0100 | 0.0104 | |
| | | 2 | 0.0096 | 0.0100 | 0.0103 | 0.0096 | 0.0100 | 0.0104 | 0.0097 | 0.0101 | 0.0105 | |
| | | 3 | 0.0096 | 0.0100 | 0.0104 | 0.0096 | 0.0099 | 0.0103 | 0.0097 | 0.0101 | 0.0105 | |
| 4000 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0093 | 0.0095 | 0.0098 | 0.0093 | 0.0095 | 0.0098 | 0.0093 | 0.0097 | 0.0099 | 0.0062 |
| | | 2 | 0.0092 | 0.0095 | 0.0098 | 0.0092 | 0.0095 | 0.0097 | 0.0093 | 0.0097 | 0.0100 | |
| | | 3 | 0.0092 | 0.0095 | 0.0098 | 0.0092 | 0.0095 | 0.0097 | 0.0093 | 0.0097 | 0.0100 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0093 | 0.0095 | 0.0098 | 0.0093 | 0.0095 | 0.0098 | 0.0093 | 0.0097 | 0.0100 | |
| | | 2 | 0.0093 | 0.0095 | 0.0098 | 0.0093 | 0.0095 | 0.0098 | 0.0093 | 0.0097 | 0.0100 | |
| | | 3 | 0.0092 | 0.0095 | 0.0098 | 0.0092 | 0.0095 | 0.0098 | 0.0093 | 0.0097 | 0.0100 | |
| 6000 | $\widehat{\mathcal{B}}_J^{(v)}$ | 1 | 0.0088 | 0.0090 | 0.0091 | 0.0088 | 0.0090 | 0.0091 | 0.0089 | 0.0090 | 0.0092 | 0.0048 |
| | | 2 | 0.0088 | 0.0090 | 0.0092 | 0.0088 | 0.0089 | 0.0091 | 0.0089 | 0.0090 | 0.0092 | |
| | | 3 | 0.0088 | 0.0089 | 0.0091 | 0.0088 | 0.0089 | 0.0091 | 0.0089 | 0.0090 | 0.0092 | |
| | $\widehat{\mathcal{B}}_J^{(u)}$ | 1 | 0.0088 | 0.0090 | 0.0091 | 0.0088 | 0.0090 | 0.0091 | 0.0089 | 0.0090 | 0.0092 | |
| | | 2 | 0.0088 | 0.0089 | 0.0091 | 0.0088 | 0.0089 | 0.0091 | 0.0089 | 0.0090 | 0.0092 | |
| | | 3 | 0.0088 | 0.0089 | 0.0091 | 0.0088 | 0.0089 | 0.0091 | 0.0089 | 0.0090 | 0.0092 | |

Table 4.4. (Approximated) Mean Squared Hellinger Distance (MSHD) for the various non linear estimator algorithms using the Symlet 3 wavelet for the density 2D Smooth comb (Figure 4.7 (d)). See text for column descriptions. Corresponding number of coefficients found in Table B.4

### 4.4.3 Real data: Old Faithful geyser

We close this chapter by revisiting the real data example of Subsection 3.4.2, the Old Faithful geyser dataset, and providing insights on practical usage of the estimator. From this point of view, we addressed three of the five points mentioned in Section 4.1, although in a slightly, surprising different way.

- For the final resolution level $J_1$, the practitioner can use any of the two $\hat{J}_n^{(v)}$, $\hat{J}_n^{(u)}$ algorithms of Section 4.2
- They can use a cut-off point or selection method as defined by (4.23), which does not require a resolution-varying function $C(j)$;

However, we still have two important unknowns

- The lowest level $J_0$ or equivalently, the number of additional levels in the beta expansion, and
- the wavelet basis $\{\varphi_{j,z}, \psi_{j,z}^{(q)}\}$.

As can be seen in the results of our simulation studies, the number of levels for the beta coefficients had most of the time the surprising effect of increasing the HD due to the additional number of coefficients to consider for hard-thresholding. We believe the introduction of soft thresholding may alleviate this, as it has a shrinking effect. As the Old Faithful geyser dataset has 272 points in $\mathbb{R}^2$, i.e. it can be seen as a $544$ dimensional data vector, one would expect that this has to be greater than the number of free parameters in the alpha and beta coefficients, thus providing a simple upper bound on the number of levels.

Picking the right wavelet basis seems more challenging and here we offer a practical guide on how to achieve this with our estimator. Figure 4.8 and Figure 4.9 show the optimisation curve generated when we apply formula (4.23) for the Daubechies $3$ and Symlet $4$ wavelet bases using $2$ levels of beta coefficients. Symlet $4$ produces a smoother density, of course, using $321$ alphas and betas. Daubechies $3$ requires fewer at $162$ coefficients. The corresponding $\widehat{\mathcal{B}}_J^{(v)}$ values are (approx.) 0.91383 (Daubechies $3$) and 0.91719 (Symlet $4$) which, if not by the inclination for a smoother density, tells us that indeed Symlet $4$ should be preferred in this case.

For comparison, see the plot of the best KDE estimate using the Gaussian kernel, where the bandwidth has been determined using CV, Figure 4.10.

Contrast above with the optimisation curves if one increases the number of vanish-

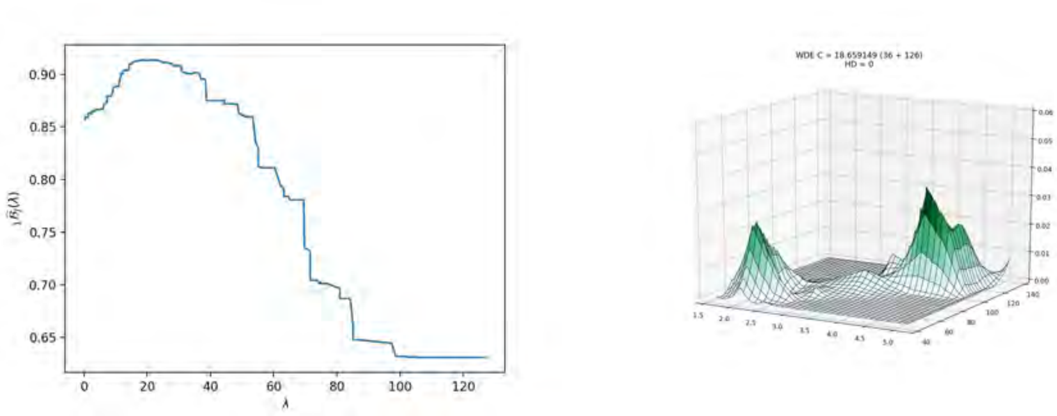


Figure 4.8. Optimisation curve, left, and density, right, for the Old Faithful geyser dataset using Daubechies 3 and $\Delta J = 2$.

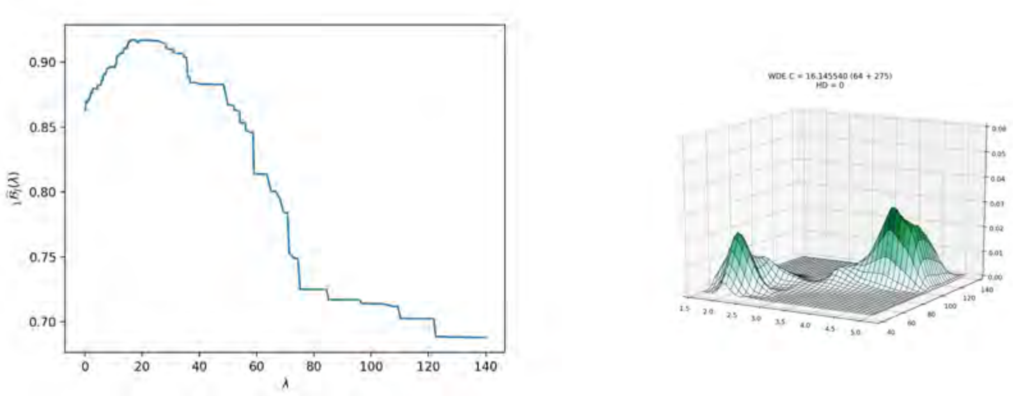


Figure 4.9. Optimisation curve, left, and density, right, for the Old Faithful geyser dataset using Symlet 4 and $\Delta J = 2$.

ing moments in both cases. Using Daubechies 4 wavelet with $\Delta J = 1$ (Figure 4.11), one can see that most of the curve lies on the right hand side of its maximum, indicating that the threshold is not going to be much effective. Indeed, the resulting density seems over-smoothed as compared to the ones above and the situation is not much different for $\Delta J = 2$ as shown in Figure 4.12, or for Symlet 5 for $\Delta J = 1$ (Figure 4.13) and $\Delta J = 2$ (Figure 4.14).

We finalise this chapter with results using biorthogonal bases, the most general wavelet construction and of interest in the recent literature on multidimensional, sparse wavelet expansions. It is worth noting that we used the tensor product method to construct multivariate wavelet expansions (e.g. (Daubechies, 1992), Ch. 10) as opposed to the anisotropic methods of curvelets (Candès and Donoho, 2004, 2005), shearlets (Labate et al., 2005) and $\alpha$-molecules (Grohs et al., 2013).

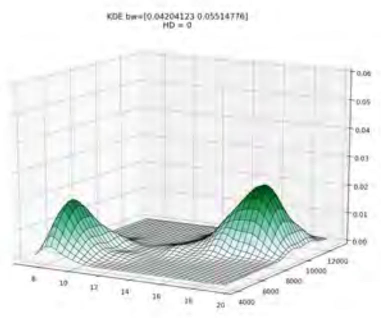


Figure 4.10. Kernel density estimator for Old Faithful geyser dataset

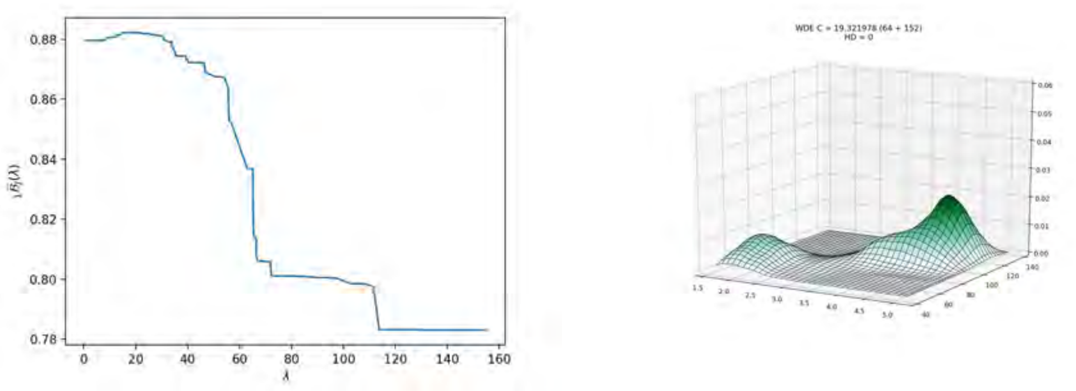


Figure 4.11. Optimisation curve, left, and density, right, for the Old Faithful geyser dataset using Daubechies $4$ and $\Delta J = 1$.

Although the results we present here are indeed promising, theoretical and practical work is required to bring those novel algorithms to our construction as it is well-known that in the general anisotropic case the number of non-zero beta coefficients using the tensor product is $O(n)$, i.e. not optimal for a general sparse representation (Starck et al., 2010).

We first show the optimisation curve and the biorthogonal, spline wavelets of Cohen and Daubechies (1992) for the case $2.6$ - this is, $6$ vanishing moments on the deconstruction filters and $2$ on the synthesis phase. We used $2$ levels of betas (Figure 4.15). In this and the following results, we used the normed version of the BC (4.19) and the corresponding algorithm (4.20). The result is not smooth: this uses a linear spline ($2$ vanishing moments) to reconstruct the signal but the optimisation curve seems balanced as described in the Daubechies $3$ and Symlet $4$ results above. Reversing the role of these bases using now $6.2$, one gets a smoother den-

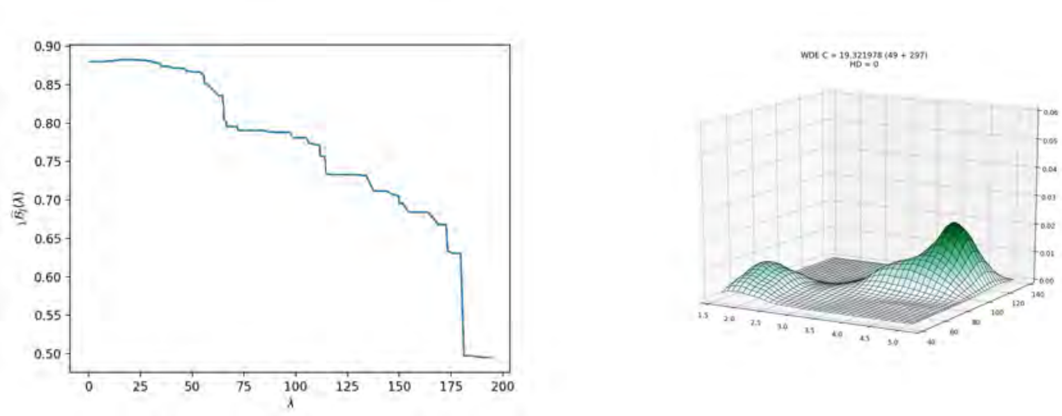


Figure 4.12. Optimisation curve, left, and density, right, for the Old Faithful geyser dataset using Daubechies 4 and $\Delta J = 2$.

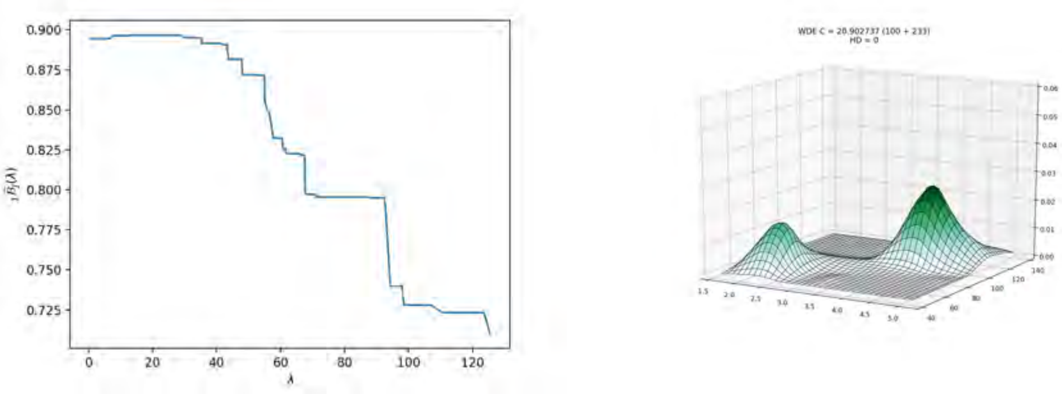


Figure 4.13. Optimisation curve, left, and density, right, for the Old Faithful geyser dataset using Symlet 5 and $\Delta J = 2$.

sity and the optimisation curve is still well-behaved, producing Figure 4.16. This change brings an improvement on the values for $\widehat{\mathcal{B}}^{(v)}_{[\tau]}$ of 0.92118 and 0.92689 for $2.6$ and $6.2$ respectively. Taking the case $6.2$ even further, we produced densities for $\Delta J = 3, 4$, shown in Figure 4.17. They are very similar, again showing peaks and jumps better as is typically the case in hard thresholding (Donoho et al., 1996), and with slight improvement on the normed BC of 0.92701 and 0.92704 for $\Delta J = 3$ and $\Delta J = 4$ levels respectively. The number of free parameters (coefficients) in the result seems also under control, 270 and 342 respectively.

We finalise this section showing the case of using similar regularity on both basis. In their seminal paper (Cohen and Daubechies, 1992), the authors mention the convenience of using different vanishing moments in the deconstruction and synthesis phase, for speed and rate of compression. As we did in here, using the filter

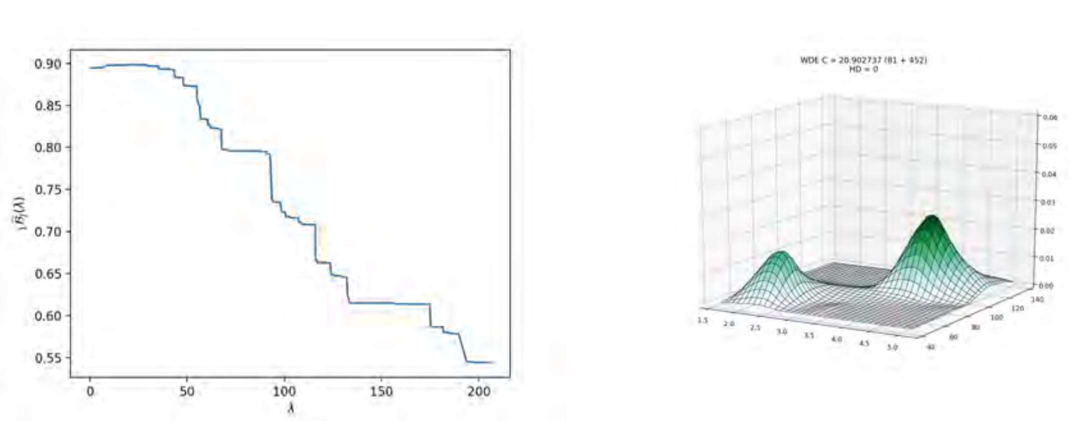


Figure 4.14. Optimisation curve, left, and density, right, for the Old Faithful geyser dataset using Symlet 5 and $\Delta J = 2$.

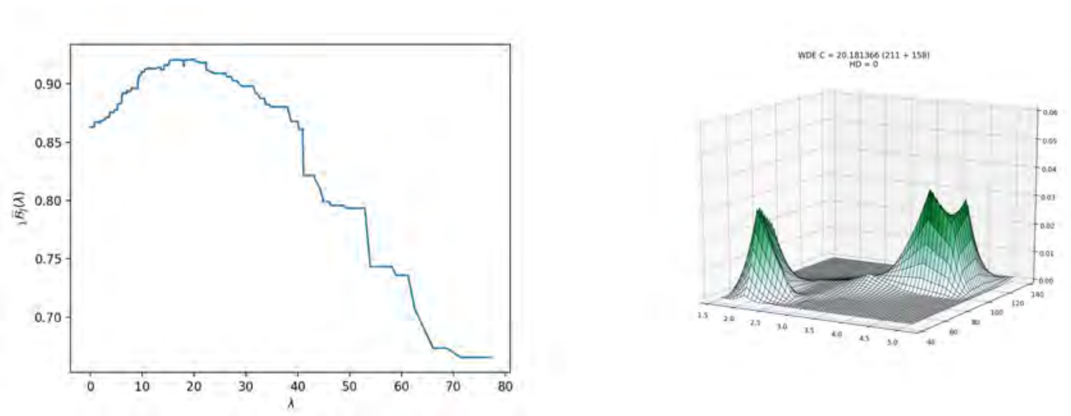


Figure 4.15. Optimisation curve, left, and density, right, for the Old Faithful geyser dataset using biorthogonal spline wavelets $2.6$ and $\Delta J = 2$.

with the highest number of vanishing moments produces a smooth result despite the fact that the deconstruction filter has a lower regularity. On the other hand, in Figure 4.18 we show the optimisation curve using the biorthogonal $3.3$ wavelet for our three different threshold strategies and using the normed BC. The end result, shown in (d) for the (c) case (the others are quite similar), has to be discarded and here is why. One can see that in (a) and (b), there are few segments, representing few $\lambda_{[\tau]}$ cut points on the right of the maximum. This means that very few beta coefficients were preserved and the threshold is too strict, leaving the estimator very similar to $g_{J_0}$, which is 2 levels below the optimum found by the first stage of the algorithm. Also, in the case of using the threshold based on the empirical variance, (4.23), the curve has several local maxima, which can be considered problematic[5].

[5] We also checked the estimator picking the maximum on the peak to the left of 4.18(c) and the

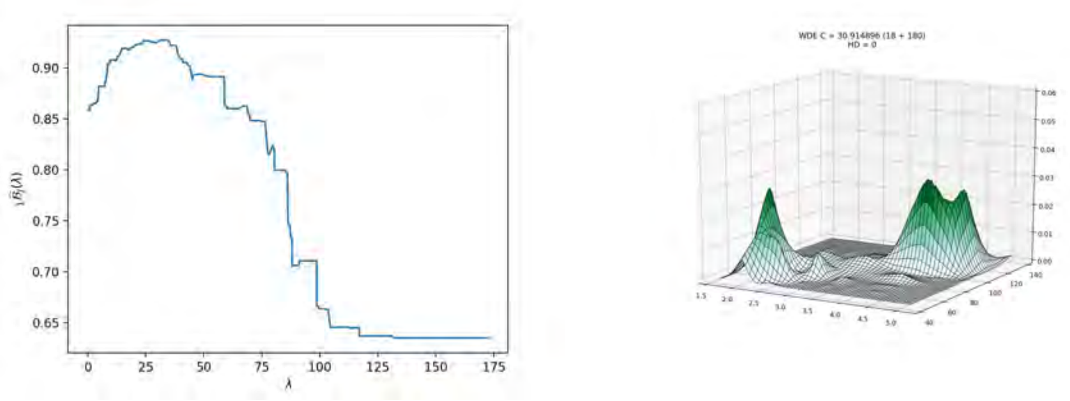


Figure 4.16. Optimisation curve, left, and density, right, for the Old Faithful geyser dataset using biorthogonal spline wavelets 6.2 and $\Delta J = 2$.

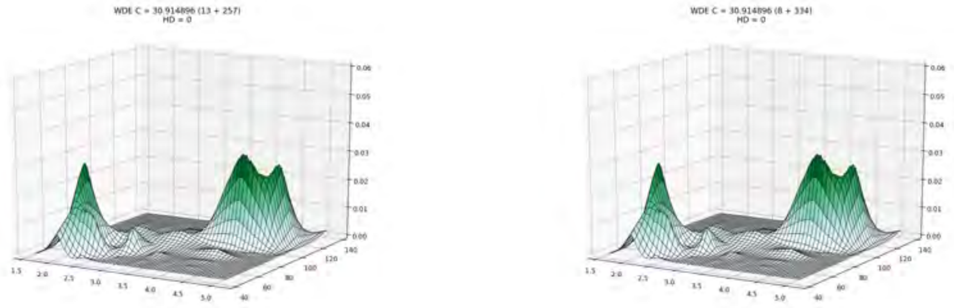
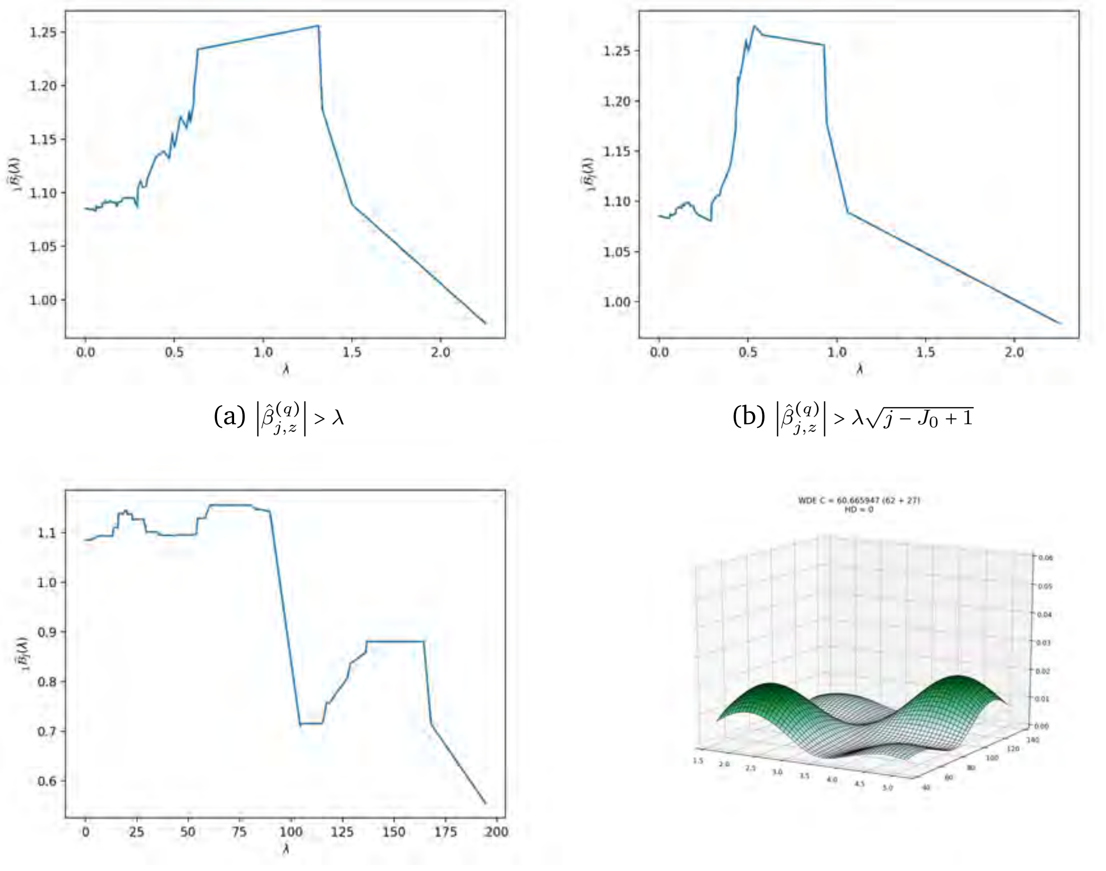


Figure 4.17. Plot of density for the Old Faithful geyser dataset using biorthogonal spline wavelets 6.2 with (left) $\Delta J = 3$ and (right) $\Delta J = 4$.

In summary, the results using spline wavelets with 6.2 vanishing moments seemed the best and, aligned with classical theory of hard thresholding for the classical estimator (Donoho et al., 1996), with better peaks and jumps[6]. With all above considerations, we just illustrated the kind of analysis that it is possible with our estimator in practice and how, guided by the optimisation curve, one can make an informed decision of which wavelet basis to chose.

generated density does not improve as most of the important betas still lie to the left of that point. It is worth adding that although in this figure it is quite evident that we have multiple peaks, most of the curves shown so far exhibit local maxima to the right of the main peak but with tiny jumps. This makes the BC difficult to use as optimisation target itself in our setting.

[6] Of course, this is very subjective, and we encourage the reader to compare for instance with Jiang and Provost (2011), Peherstorfer et al. (2014), or Kovacs et al. (2017).



(a) $\left|\hat{\beta}^{(q)}_{j,z}\right| > \lambda$

(b) $\left|\hat{\beta}^{(q)}_{j,z}\right| > \lambda\sqrt{j - J_0 + 1}$

(c) $\left|\hat{\beta}^{(q)}_{j,z}\right| > \lambda\,\hat{\sigma}^{(q)}_{j,z}$

(d) Density result for (c)

Figure 4.18. Optimisation curves for different threshold strategies using biorthogonal splines 3.3 for the Old Faithful geyser dataset $\Delta J = 2$

# Chapter 5

# Image analysis application

## 5.1 Preliminaries

In this chapter we apply and extend the methods developed in Chapter 3 and Chapter 4 to a couple of corpus in machine learning, the MNIST and the Fashion-MNIST datasets.

The MNIST dataset is a large database of labelled handwritten digits of the postal office commonly used as benchmark for machine learning and image processing algorithms. It consists of 60000 images of the ten digits, each set of approximately 6000 observations, along with 10000 test images. The digits are monochrome, anti-aliased and normalised to a $28 \times 28$ pixel box. A sample of ten samples for each digit is shown in Figure 5.1. This was featured in the original paper that introduced the convolutional neural network architecture LeCun et al. (1998) as a *modified NIST* - specifically adjusted to improve statistical homogeneity of the data[1]. This was one of the first success stories of convolutional neural networks, in this case applied to a Optical Character Recognition (OCR), further exploiting the idea of backpropagation and "representation" developed for neural networks in Rumelhart et al. (1986). A more detailed explanation of the method as it applies to this dataset can be found in Efron and Hastie (2016).

[1] NIST has two sources of digits, some written by censor bureau personnel and other by high school students, having arguably different statistical properties. That was corrected by MNIST.

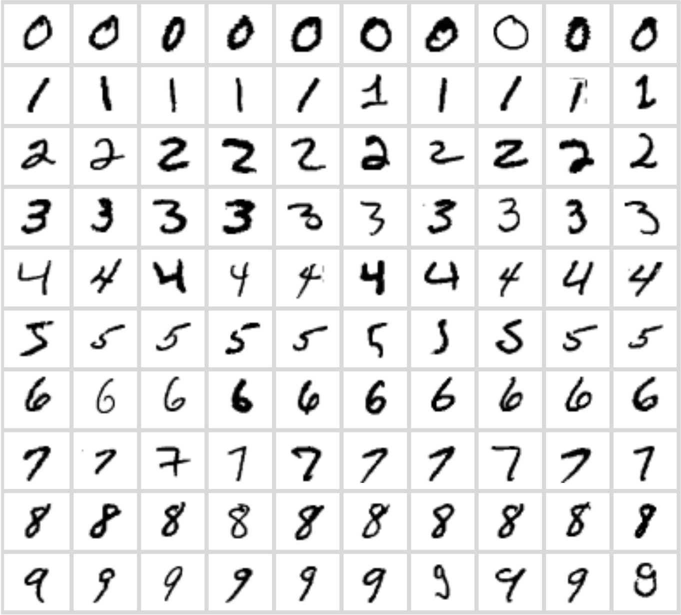


Figure 5.1. Sample of 10 observations from the MNIST dataset for each digit

The second, Fashion-MNIST, is a more recent and perhaps less known dataset[2] designed as a drop-in replacement to MNIST (Xiao et al., 2017). It follows identical binary format representation, so existing algorithms that work with MNIST can be readily used on this data. It consists of processed images from the Zalando's e-commerce store, which offers fashion articles in different categories for men, women, kids and neutral. The original pictures of the products were processed through a standardised pipeline to produce pictures of 28 x 28 pixels. The products in the store are featured from different angles, but this dataset contains only the front version of each product. To facilitate comparison with MNIST, the products are divided in 10 categories as well, see Figure 5.2. This dataset was built as a more realistic benchmark for image processing algorithms as it contains images of real objects, albeit in a restricted setting and with very small size compared with real pictures. Nonetheless, it appears to be a more challenging task than MNIST for several algorithms reported in the paper, having on average a 10% difference in accuracy between the two datasets. Although it is claimed there are *near duplicates* in the original Fashion-MNIST between the training and test sets, artificially inflating test accuracy results(Geier, 2019), we used the original published dataset as the differences to be under 1% and there is a component of subjectivity in this

[2] According to Google scholar, as of today, there are more than 1,300 pre-prints and papers citing this dataset.

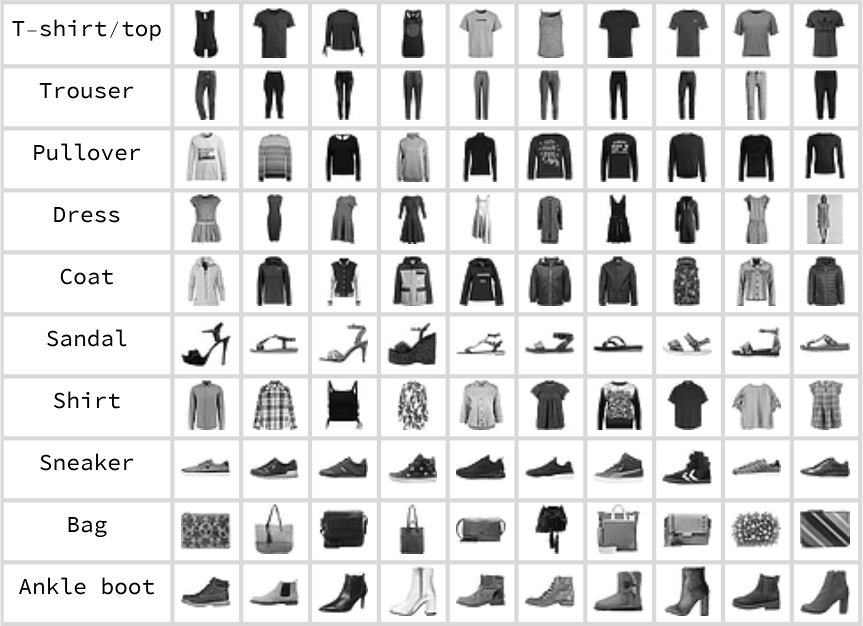


Figure 5.2. Sample of 10 observations from the Fashion-MNIST dataset for each category

analysis.

Let's describe the problem more formally. Let $\mathbf{W}$ and $\mathbf{H}$ be the spaces corresponding to width and height indexes. Theoretically they can be the $[0, 1]$ interval, but in practice they are just the integer indexes. Likewise, $[0, 1]$ can represent the different values of grey although in digitalised images they are values in $\{0 \dots 255\}$. An image is then a function $f : \mathbf{W} \times \mathbf{H} \rightarrow [0, 1]$. Let $\mathcal{I}$ be the space of images, a subspace of $L^2$, with images normalised such that $\int_{\mathbf{W} \times \mathbf{H}} f^2(w, h) \, \mathrm{d}w \, \mathrm{d}h = 1$. This normalisation makes each image the square root of a PDF and $\mathcal{I}$ a subset of the hypersphere, a Riemannian manifold - an approach similar to Srivastava et al. (2007); Peter and Rangarajan (2008). Thus, the problem for both, MNIST and Fashion-MNIST, is to built a function $C : \mathcal{I} \rightarrow \{0 \dots 9\}$, that correctly identifies the label (digit or category) of a given image $f$. Note that in MNIST and Fashion-MNIST, an image $f$ always belong to a class. This is of course an instance of *supervised learning*. One can think of each MNIST digit as a cloud of points on the hypersphere, like in Figure 5.3. Because images are originally positive functions, i.e. $f \geq 0$, observations for each class are in the *positive octant* of the hypersphere as shown in the picture. Now, imagine that each class of images $\mathcal{C}_i \subset \mathcal{I}$ is somewhat paired with $\pi_i$, a PDF on $\mathcal{I}$, such that a classifier $C$ can be defined as $C(f) = \mathrm{argmax}_i \, \pi_i(f)$. Cast this way, a classification algorithm can be seen as a problem of estimating such $\pi_i$ for

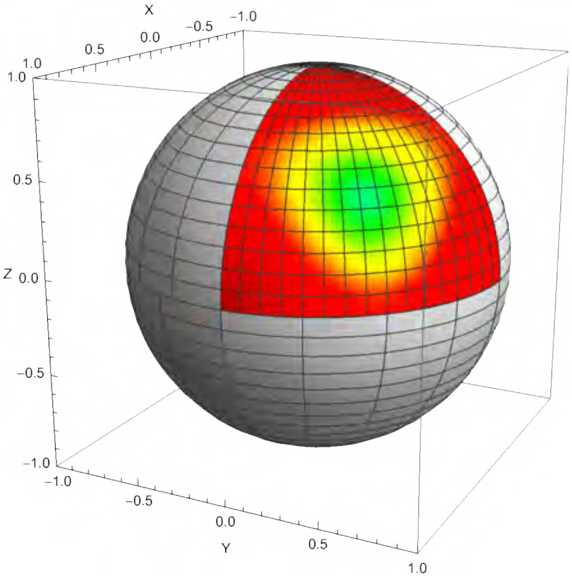


Figure 5.3. The all positive *octant* on the $S^2$ hypersphere as embedded in $R^3$.

each class based on a sample for each. This problem can be solved in a nonparametric way using nearest neighbours, originally called *discriminatory analysis* (Fix and Hodges, 1951, 1952; Cover and Hart, 1967).

To develop this method, a notion of distance is required in the sample space. Thus, one key advantage of working on the hypersphere, a Riemannian manifold, is that close formulas for geodesics, exponential maps, inverse of exponential maps (logs) and other quantities are available in analytic form. Besides, in recent developments in computer vision and machine learning, there is now interest in using spaces with non-Euclidean geometries, aiming to capture intrinsic properties of the data, e.g. Pizer and Marron (2017).

## 5.2 Landmark nearest neighbours on the hypersphere

In general, $k$-NN classification works by simply picking the most common label appearing among the $k$-NN of a given point $f$. It is a nonparametric algorithm as described in Chapter 2 where, similarly to KDE, each observation is kept as a parameter of the classifier. For multivariate datasets, this makes the classification function $C$ having $O(nd)$ complexity, where $n$ is the number of samples and $d$ is their dimension, which is not practical for big data applications (Cai and Chen, 2015; Deng et al., 2016). For instance, in the MNIST case, using a mid range laptop, the Riemannian distance between a test image and an observation takes around 0.4 ms. So, obtaining the closest digit among the 60000 images in the dataset can take around 25 secs, a performance not suitable for most applications. It is natural to think of reducing the dimension of the data or in selecting a much smaller subset of observations in such a way that they are somehow representative of the set and no accuracy is lost.

A popular technique for dimensionality reduction is spectral embedding. Spectral embedding has a long history (Chung and Graham, 1997; Weiss, 1999) and it is based on the connections between the eigen decomposition of the Laplacian matrix of a graph and graph partitioning. This method has been extended and applied in the setting of an *affinity* matrix between observations (Ng et al., 2002). The affinity or similarity matrix is a measure of the closeness between objects that assigns to each pair of observations a value $\operatorname{aff}(x_i, x_j) = a_{i,j}$ in $[0, 1]$, with $0$ representing no affinity and $1$ maximum similarity. The embedding is done by selecting the top $d$ eigenvectors in the corresponding eigenvalue decomposition and use them to calculate a projection of the observations in the lower dimensional space $\mathbb{R}^d$. Figure 5.4 and Figure 5.5 show an example embedding into $\mathbb{R}^3$ for the different digits of MNIST. There are several ways to convert a metric into an affinity. The most widely used are the linear and exponential methods, which will be explained in Results section.

Reducing the number of relevant observations is also an important strategy. Yan et al. (2009) focus on reducing the execution time of an spectral embedding. It presents what the authors call $k$-means approximate spectral clustering (KASP). This algorithm uses k-means clustering on the original data to obtain centroids $y_{1..k}$ and build a matching table to associate each observation to its closest centroid $y$. Then run spectral clustering on the centroids to obtain an $m$-way cluster membership for each centroid. Finally, recover the membership for each observation by looking up the membership of the matched centroid. In Cai and Chen (2015) the authors use instead a method based on sparse encoding to find a good approximation to the matrix factorisation of the affinity matrix. The basis vectors used in the sparse representation are called *landmarks* and every observation has

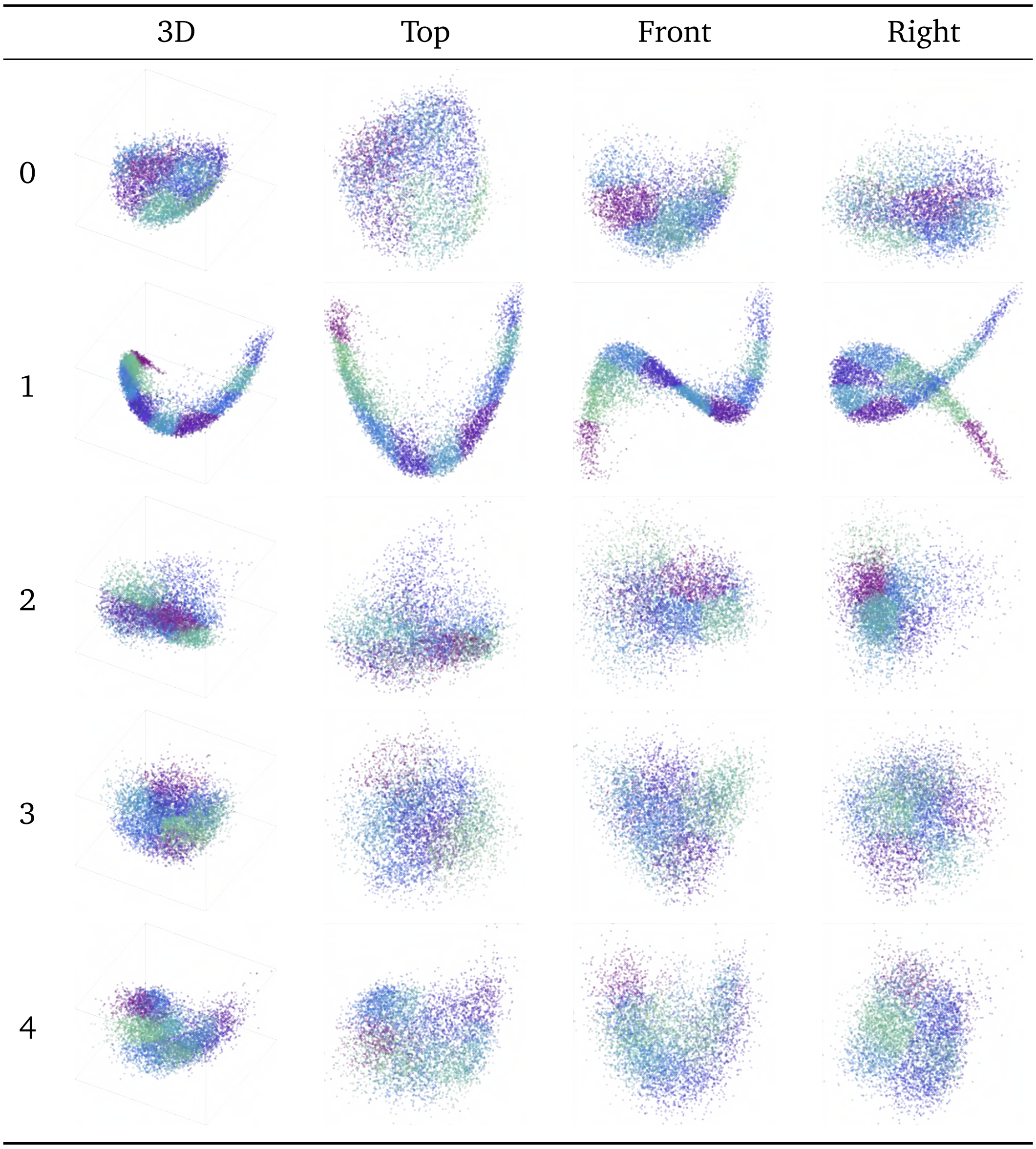


Figure 5.4. $R^3$ embedding of MNIST digits "0" to "4" with 10 clusters.

a sparse representation in this basis. Again, the purpose is to reduce the computational costs associated with spectral clustering of the affinity matrix. Deng et al. (2016) applies this method in the context of "big data". In the above methods, the norms used are the Euclidean norms in the respective spaces. In here, we present a similar approach to those but based on the hypersphere.

The algorithm is a straightforward application of spectral embedding and goes as follows (see Algorithm 1). First, pairwise distances are calculated for the given image category and then converted to an affinity matrix. This is used to generate a spectral embedding into $p$ dimensions of the image category $C$. Over these projected vectors $V \subset \mathbb{R}^p$, standard $k$-means is calculated to get $l$ clusters $K_c \subset$

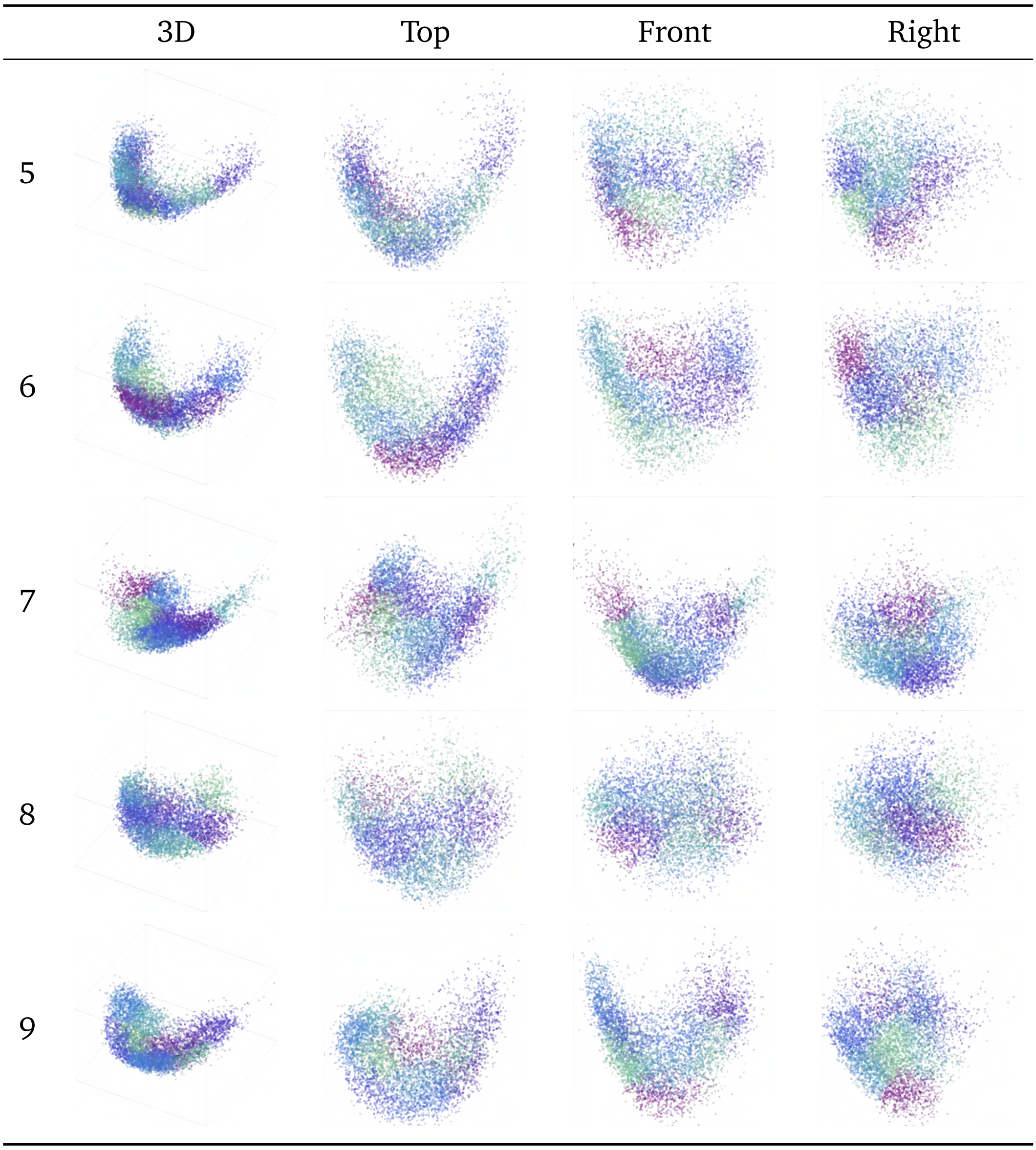


Figure 5.5. $R^3$ embedding of MNIST digits "5" to "9" with 10 clusters.

$\{1, \dots, \#(\iota_C)\}$. Finally, we look back the images $\iota_C$ and calculate the Karcher mean for each cluster based on the indexes of the corresponding vectors $K_c$. We explain each of these steps below.

As hinted by these plots, the densities of the different MNIST categories into the hypersphere are not convex. That is, the geodesic between two images within the same category is not guaranteed to travel within the same digit. This further motivates the need to find some sort of "coverage" of the density by some landmarks which would represent neighbouring images and that, in turn, will serve as a reference for $k$-NN. This is captured in the next step, in which we use a standard implementation of $k$-means to find those clusters.

**Algorithm 1** Simple landmark selection on the hypersphere

```
function LANDMARKS($\iota_C, p, l$)
    // $\iota_C$: array of images in a particular category $C$
    // $p$: integer, dimensions in projection space, $p \geq 1$
    // $l$: integer, number of landmarks, $l \geq 1$
    for all $\iota_i \in C$, $\iota_{i'} \in C$, $i' > i$, do
        $D_{i,i'} = \delta(\iota_i, \iota_{i'})$ // Pairwise distance on the hypersphere
    end for
    $A \leftarrow$ TOAFFINITY($D$)
    $E \leftarrow$ SPECTRALEMBEDDING($A, p$)
    $V \leftarrow$ PROJECT($E, \iota_C$)
    $K \leftarrow$ KMEANS($V, l$)
    for $c \leftarrow 1 \dots l$ do
        $S_c \leftarrow \{\iota_i : \iota_i \in K_c\}$
        $m_c \leftarrow$ KARCHERMEAN($S_c$)
    end for
    return $\{m_c : c = 1 \dots l\}$
end function
```

In these figures, we have represented, by using different colours, 10 clusters for each $\mathbb{R}^3$ embedding as an example. In practice, we used a larger number of dimensions and a much larger number of clusters. Finally, we need to "invert" those clusters into the hypersphere in order to find corresponding landmarks. To do this, we calculate the Karcher mean for each found cluster in the hypersphere. The Karcher mean is a generalisation of the centre of mass, i.e. the mean, in the context of Riemannian manifolds (Grove, 1976; Karcher, 1977)[3]. It is defined as the point that minimises the distance, using the manifold's intrinsic metric, to a set of points

$$\mu = \arg\min_{p \in M} \sum_{i=1}^{N} d^2 (p, x_i) . \tag{5.1}$$

The distance $d$ on the hypersphere, defined as the length of the geodesic between two points, can also be easily calculated by $d(x_i, x_j) = \arccos \langle x_i, x_j \rangle$.

The mean can be calculated iteratively as per Algorithm 2 (Peter et al., 2017; Pennec, 2006) where $\kappa$ is a *learning rate* parameter that regulates speed of convergence and $\epsilon$ is a desired threshold. The expressions $\mathrm{Exp}_\mu(\gamma)$ and $\mathrm{Log}_\mu(f)$ stand for the exponential and inverse-exponential maps on the manifold. The first is

[3] This is also named Fréchet mean or Riemannian centre of mass. In tandem with current literature, we will use Karcher mean, although Karcher himself preferred the term Riemannian centre of mass as, according to him, it better conveys its goal and origins(Karcher, 2014).

**Algorithm 2** Karcher mean

```
function KARCHERMEAN($S$)
    // $S = \{f_i\}$: set of $m = \#(S)$ images
    $\mu \leftarrow f_1$
    repeat
        $\gamma^t = \frac{\kappa}{m} \sum_{i=1}^{m} \mathrm{Log}_\mu(f_i)$
        $\mu = \mathrm{Exp}_\mu(\gamma^t)$
    until $\|\gamma_t - \gamma_{t-1}\| < \epsilon$
    return $\mu$
end function
```

the map that intuitively develops a geodesic along a given vector $\gamma$ in the tangent bundle at $\mu$ using the canonical affine connection determined by the metric. The $\mathrm{Log}_\mu(f)$ is the reverse operation in the sense that it gives the corresponding tangent vector along the geodesic connecting two points (Lee, 2013). In general, they are *local* operations available around a neighbourhood of $\mu$ and usually hard to compute. However, on the hypersphere they are available in closed form and defined almost everywhere[4]. Those closed forms are (Pennec, 2006; Lee, 2013; Peter et al., 2017)

$$\mathrm{Exp}_\mu(\gamma) = \cos(|\gamma|)\mu + \sin(|\gamma|)\frac{\gamma}{|\gamma|} \tag{5.2}$$

$$\mathrm{Log}_\mu(\iota) = \tilde{\rho}\frac{\cos^{-1}(\langle \mu, f \rangle)}{\sqrt{\langle \tilde{\rho}, \tilde{\rho} \rangle}}, \tag{5.3}$$

where $\tilde{\rho} = f - \langle \mu, f \rangle \mu$.

In the following section we present the results of the above algorithm and discuss future work.

## 5.3 Results

We ran several scenarios for the landmark $k$-NN on the hypershpere algorithm, for both the MNIST and Fashion-MNIST datasets. There are several free parameters in Algorithm 1 to pick landmarks for each class: the number of landmarks per class, the way to translate the distance metric into an affinity matrix and the number of projections for the spectral embedding of such matrix. For the number

[4] The cut locus of a point in the hypersphere is the point on the opposite side of the diameter, hence exponential and logarithmic maps are defined in the whole hypersphere except at that point.

of landmarks we picked 25, 50, 75, 125 and 175 as examples. Naturally, overall accuracy improves as the number of landmarks increases. On the flip side, the time spent per classification increases as it is linear in the number of landmarks and the number of classes. This means that the difference in speed of the classification, important when dealing with a chain of several algorithms in a data pipeline, between the $25$ (fastest) and $175$ (slowest) extremes is sevenfold and the correct choice is a the trade-off between performance and accuracy. See Table B.6.

| KM | Affinity | $R^p$ | $0$ | $1$ | $2$ | $3$ | $4$ | $5$ | $6$ | $7$ | $8$ | $9$ | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 25 | Linear | 3 | 0.962 | 0.981 | 0.856 | 0.869 | 0.744 | 0.710 | 0.941 | 0.837 | 0.759 | 0.834 | 0.852 |
| | | 8 | 0.970 | 0.984 | 0.826 | 0.875 | 0.747 | 0.667 | 0.916 | 0.802 | 0.806 | 0.838 | 0.846 |
| | | 21 | 0.975 | 0.982 | 0.842 | 0.854 | 0.728 | 0.709 | 0.915 | 0.813 | 0.824 | 0.846 | 0.852 |
| | 0.2 | 3 | 0.968 | 0.969 | 0.842 | 0.805 | 0.784 | 0.580 | 0.943 | 0.812 | 0.826 | 0.692 | 0.826 |
| | | 8 | 0.978 | 0.971 | 0.871 | 0.825 | 0.764 | 0.681 | 0.922 | 0.799 | 0.789 | 0.807 | 0.844 |
| | | 21 | 0.979 | 0.974 | 0.865 | 0.825 | 0.749 | 0.743 | 0.944 | 0.863 | 0.846 | 0.860 | 0.867 |
| | 0.4 | 3 | 0.976 | 0.984 | 0.872 | 0.848 | 0.733 | 0.742 | 0.951 | 0.847 | 0.832 | 0.857 | 0.867 |
| | | 8 | 0.988 | 0.982 | 0.876 | 0.863 | 0.753 | 0.747 | 0.942 | 0.846 | 0.837 | 0.878 | 0.874 |
| | | 21 | 0.988 | 0.986 | 0.879 | 0.867 | 0.722 | 0.702 | 0.945 | 0.846 | 0.843 | 0.887 | 0.870 |
| | 0.6 | 3 | 0.979 | 0.979 | 0.845 | 0.867 | 0.745 | 0.719 | 0.941 | 0.824 | 0.791 | 0.866 | 0.858 |
| | | 8 | 0.986 | 0.982 | 0.835 | 0.870 | 0.738 | 0.730 | 0.928 | 0.830 | 0.815 | 0.867 | 0.861 |
| | | 21 | 0.981 | 0.985 | 0.850 | 0.873 | 0.681 | 0.717 | 0.922 | 0.797 | 0.706 | 0.900 | 0.845 |
| | 0.8 | 3 | 0.964 | 0.984 | 0.872 | 0.866 | 0.736 | 0.695 | 0.947 | 0.810 | 0.765 | 0.856 | 0.853 |
| | | 8 | 0.977 | 0.980 | 0.837 | 0.857 | 0.714 | 0.701 | 0.925 | 0.804 | 0.835 | 0.872 | 0.853 |
| | | 21 | 0.978 | 0.983 | 0.827 | 0.866 | 0.720 | 0.685 | 0.927 | 0.830 | 0.773 | 0.872 | 0.849 |
| | 1.0 | 3 | 0.972 | 0.982 | 0.856 | 0.863 | 0.742 | 0.702 | 0.945 | 0.832 | 0.771 | 0.833 | 0.853 |
| | | 8 | 0.977 | 0.982 | 0.835 | 0.862 | 0.747 | 0.700 | 0.930 | 0.788 | 0.781 | 0.837 | 0.847 |
| | | 21 | 0.969 | 0.984 | 0.831 | 0.859 | 0.697 | 0.701 | 0.899 | 0.829 | 0.790 | 0.857 | 0.845 |
| 50 | Linear | 3 | 0.980 | 0.989 | 0.878 | 0.872 | 0.785 | 0.785 | 0.945 | 0.865 | 0.850 | 0.877 | 0.885 |
| | | 8 | 0.982 | 0.985 | 0.871 | 0.890 | 0.786 | 0.772 | 0.947 | 0.869 | 0.848 | 0.893 | 0.887 |
| | | 21 | 0.985 | 0.986 | 0.869 | 0.873 | 0.780 | 0.761 | 0.956 | 0.866 | 0.830 | 0.883 | 0.881 |
| | 0.2 | 3 | 0.979 | 0.974 | 0.864 | 0.850 | 0.822 | 0.772 | 0.952 | 0.838 | 0.839 | 0.859 | 0.877 |
| | | 8 | 0.986 | 0.981 | 0.906 | 0.861 | 0.826 | 0.736 | 0.947 | 0.811 | 0.860 | 0.859 | 0.880 |
| | | 21 | 0.989 | 0.982 | 0.898 | 0.859 | 0.803 | 0.818 | 0.950 | 0.881 | 0.887 | 0.900 | 0.898 |
| | 0.4 | 3 | 0.980 | 0.983 | 0.898 | 0.858 | 0.789 | 0.810 | 0.954 | 0.885 | 0.874 | 0.884 | 0.893 |
| | | 8 | 0.987 | 0.987 | 0.905 | 0.878 | 0.817 | 0.828 | 0.950 | 0.871 | 0.872 | 0.900 | 0.901 |
| | | 21 | 0.989 | 0.987 | 0.900 | 0.869 | 0.785 | 0.807 | 0.954 | 0.875 | 0.875 | 0.897 | 0.896 |
| | 0.6 | 3 | 0.980 | 0.989 | 0.873 | 0.873 | 0.764 | 0.787 | 0.962 | 0.871 | 0.857 | 0.901 | 0.888 |
| | | 8 | 0.984 | 0.987 | 0.890 | 0.884 | 0.788 | 0.796 | 0.949 | 0.863 | 0.863 | 0.896 | 0.892 |
| | | 21 | 0.990 | 0.986 | 0.888 | 0.884 | 0.782 | 0.793 | 0.959 | 0.863 | 0.838 | 0.897 | 0.890 |
| | 0.8 | 3 | 0.979 | 0.990 | 0.884 | 0.857 | 0.752 | 0.780 | 0.962 | 0.867 | 0.850 | 0.887 | 0.883 |
| | | 8 | 0.986 | 0.986 | 0.879 | 0.890 | 0.791 | 0.788 | 0.951 | 0.862 | 0.847 | 0.882 | 0.888 |
| | | 21 | 0.989 | 0.987 | 0.880 | 0.887 | 0.783 | 0.784 | 0.954 | 0.871 | 0.846 | 0.889 | 0.889 |
| | 1.0 | 3 | 0.981 | 0.988 | 0.883 | 0.871 | 0.774 | 0.764 | 0.963 | 0.871 | 0.850 | 0.892 | 0.886 |

Continued on next page

Table 5.1 – continued from previous page

| KM | Affinity | $R^p$ | $0$ | $1$ | $2$ | $3$ | $4$ | $5$ | $6$ | $7$ | $8$ | $9$ | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
|  |  | 8 | 0.983 | 0.986 | 0.890 | 0.891 | 0.780 | 0.783 | 0.949 | 0.858 | 0.850 | 0.887 | 0.888 |
|  |  | 21 | 0.985 | 0.986 | 0.873 | 0.878 | 0.791 | 0.783 | 0.954 | 0.869 | 0.840 | 0.885 | 0.887 |
| 75 | Linear | 3 | 0.983 | 0.990 | 0.895 | 0.880 | 0.821 | 0.802 | 0.957 | 0.878 | 0.871 | 0.893 | 0.899 |
|  |  | 8 | 0.986 | 0.988 | 0.888 | 0.902 | 0.836 | 0.813 | 0.954 | 0.877 | 0.864 | 0.899 | 0.903 |
|  |  | 21 | 0.987 | 0.987 | 0.880 | 0.882 | 0.829 | 0.800 | 0.968 | 0.879 | 0.867 | 0.891 | 0.899 |
|  | 0.2 | 3 | 0.982 | 0.979 | 0.872 | 0.870 | 0.843 | 0.780 | 0.958 | 0.881 | 0.840 | 0.898 | 0.892 |
|  |  | 8 | 0.988 | 0.982 | 0.914 | 0.877 | 0.870 | 0.801 | 0.954 | 0.837 | 0.878 | 0.897 | 0.901 |
|  |  | 21 | 0.989 | 0.983 | 0.912 | 0.869 | 0.843 | 0.821 | 0.960 | 0.884 | 0.908 | 0.913 | 0.910 |
|  | 0.4 | 3 | 0.981 | 0.990 | 0.913 | 0.882 | 0.829 | 0.849 | 0.958 | 0.884 | 0.887 | 0.908 | 0.910 |
|  |  | 8 | 0.989 | 0.987 | 0.918 | 0.879 | 0.836 | 0.862 | 0.961 | 0.875 | 0.891 | 0.909 | 0.912 |
|  |  | 21 | 0.988 | 0.989 | 0.915 | 0.879 | 0.824 | 0.846 | 0.966 | 0.897 | 0.880 | 0.902 | 0.910 |
|  | 0.6 | 3 | 0.985 | 0.991 | 0.911 | 0.873 | 0.804 | 0.819 | 0.963 | 0.882 | 0.899 | 0.899 | 0.904 |
|  |  | 8 | 0.987 | 0.990 | 0.903 | 0.896 | 0.821 | 0.825 | 0.964 | 0.888 | 0.868 | 0.902 | 0.906 |
|  |  | 21 | 0.988 | 0.988 | 0.902 | 0.892 | 0.821 | 0.826 | 0.966 | 0.895 | 0.865 | 0.895 | 0.906 |
|  | 0.8 | 3 | 0.983 | 0.992 | 0.897 | 0.878 | 0.807 | 0.806 | 0.966 | 0.886 | 0.874 | 0.894 | 0.900 |
|  |  | 8 | 0.984 | 0.989 | 0.899 | 0.902 | 0.832 | 0.824 | 0.963 | 0.877 | 0.861 | 0.896 | 0.904 |
|  |  | 21 | 0.987 | 0.989 | 0.892 | 0.897 | 0.822 | 0.823 | 0.971 | 0.889 | 0.862 | 0.883 | 0.903 |
|  | 1.0 | 3 | 0.983 | 0.990 | 0.896 | 0.887 | 0.800 | 0.799 | 0.962 | 0.882 | 0.859 | 0.903 | 0.898 |
|  |  | 8 | 0.986 | 0.988 | 0.908 | 0.896 | 0.817 | 0.817 | 0.958 | 0.879 | 0.859 | 0.906 | 0.903 |
|  |  | 21 | 0.986 | 0.990 | 0.883 | 0.895 | 0.831 | 0.825 | 0.960 | 0.885 | 0.862 | 0.904 | 0.904 |
| 125 | Linear | 3 | 0.988 | 0.992 | 0.905 | 0.890 | 0.849 | 0.853 | 0.967 | 0.891 | 0.871 | 0.913 | 0.913 |
|  |  | 8 | 0.988 | 0.990 | 0.912 | 0.906 | 0.854 | 0.852 | 0.973 | 0.900 | 0.881 | 0.917 | 0.919 |
|  |  | 21 | 0.986 | 0.989 | 0.902 | 0.901 | 0.861 | 0.850 | 0.970 | 0.901 | 0.887 | 0.907 | 0.917 |
|  | 0.2 | 3 | 0.984 | 0.984 | 0.909 | 0.877 | 0.869 | 0.853 | 0.968 | 0.904 | 0.876 | 0.921 | 0.916 |
|  |  | 8 | 0.988 | 0.986 | 0.933 | 0.902 | 0.892 | 0.838 | 0.956 | 0.865 | 0.869 | 0.918 | 0.916 |
|  |  | 21 | 0.988 | 0.984 | 0.922 | 0.889 | 0.864 | 0.849 | 0.967 | 0.888 | 0.923 | 0.925 | 0.921 |
|  | 0.4 | 3 | 0.988 | 0.992 | 0.932 | 0.887 | 0.867 | 0.863 | 0.962 | 0.894 | 0.911 | 0.917 | 0.923 |
|  |  | 8 | 0.990 | 0.991 | 0.932 | 0.895 | 0.875 | 0.882 | 0.971 | 0.896 | 0.905 | 0.908 | 0.926 |
|  |  | 21 | 0.989 | 0.991 | 0.930 | 0.902 | 0.876 | 0.866 | 0.971 | 0.904 | 0.895 | 0.913 | 0.925 |
|  | 0.6 | 3 | 0.987 | 0.991 | 0.923 | 0.879 | 0.852 | 0.861 | 0.970 | 0.891 | 0.889 | 0.910 | 0.917 |
|  |  | 8 | 0.989 | 0.992 | 0.919 | 0.897 | 0.860 | 0.862 | 0.974 | 0.902 | 0.892 | 0.906 | 0.921 |
|  |  | 21 | 0.988 | 0.990 | 0.921 | 0.900 | 0.860 | 0.869 | 0.975 | 0.911 | 0.878 | 0.912 | 0.922 |
|  | 0.8 | 3 | 0.987 | 0.993 | 0.911 | 0.894 | 0.857 | 0.853 | 0.964 | 0.886 | 0.881 | 0.904 | 0.914 |
|  |  | 8 | 0.987 | 0.990 | 0.916 | 0.904 | 0.855 | 0.853 | 0.970 | 0.897 | 0.883 | 0.902 | 0.917 |
|  |  | 21 | 0.987 | 0.991 | 0.907 | 0.901 | 0.865 | 0.859 | 0.976 | 0.906 | 0.885 | 0.907 | 0.920 |
|  | 1.0 | 3 | 0.986 | 0.992 | 0.912 | 0.890 | 0.857 | 0.859 | 0.960 | 0.898 | 0.874 | 0.913 | 0.916 |
|  |  | 8 | 0.987 | 0.990 | 0.916 | 0.911 | 0.852 | 0.845 | 0.971 | 0.898 | 0.863 | 0.904 | 0.915 |
|  |  | 21 | 0.987 | 0.990 | 0.910 | 0.900 | 0.854 | 0.843 | 0.973 | 0.910 | 0.880 | 0.904 | 0.917 |
| 175 | Linear | 3 | 0.988 | 0.992 | 0.917 | 0.900 | 0.867 | 0.876 | 0.972 | 0.896 | 0.906 | 0.920 | 0.924 |
|  |  | 8 | 0.988 | 0.992 | 0.926 | 0.909 | 0.874 | 0.867 | 0.976 | 0.901 | 0.896 | 0.916 | 0.926 |
|  |  | 21 | 0.988 | 0.992 | 0.922 | 0.906 | 0.888 | 0.885 | 0.975 | 0.912 | 0.887 | 0.914 | 0.928 |
|  | 0.2 | 3 | 0.987 | 0.988 | 0.918 | 0.883 | 0.893 | 0.868 | 0.972 | 0.913 | 0.893 | 0.924 | 0.925 |
|  |  | 8 | 0.985 | 0.989 | 0.939 | 0.908 | 0.883 | 0.854 | 0.954 | 0.897 | 0.908 | 0.919 | 0.925 |

Continued on next page

Table 5.1 – continued from previous page

| KM | Affinity | $R^p$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 21 | 0.989 | 0.985 | 0.933 | 0.902 | 0.875 | 0.867 | 0.969 | 0.890 | 0.924 | 0.938 | 0.928 |
| | 0.4 | 3 | 0.988 | 0.991 | 0.928 | 0.894 | 0.879 | 0.880 | 0.970 | 0.901 | 0.910 | 0.924 | 0.928 |
| | | 8 | 0.990 | 0.991 | 0.934 | 0.903 | 0.897 | 0.892 | 0.977 | 0.906 | 0.914 | 0.918 | **0.933** |
| | | 21 | 0.989 | 0.991 | 0.936 | 0.910 | 0.891 | 0.886 | 0.975 | 0.907 | 0.903 | 0.922 | 0.932 |
| | 0.6 | 3 | 0.986 | 0.992 | 0.927 | 0.892 | 0.881 | 0.879 | 0.973 | 0.898 | 0.900 | 0.917 | 0.926 |
| | | 8 | 0.987 | 0.992 | 0.926 | 0.902 | 0.885 | 0.880 | 0.979 | 0.902 | 0.896 | 0.913 | 0.927 |
| | | 21 | 0.988 | 0.990 | 0.928 | 0.908 | 0.880 | 0.886 | 0.977 | 0.907 | 0.889 | 0.911 | 0.927 |
| | 0.8 | 3 | 0.988 | 0.992 | 0.924 | 0.897 | 0.866 | 0.866 | 0.967 | 0.900 | 0.895 | 0.925 | 0.924 |
| | | 8 | 0.988 | 0.992 | 0.921 | 0.904 | 0.877 | 0.867 | 0.975 | 0.906 | 0.885 | 0.907 | 0.924 |
| | | 21 | 0.989 | 0.991 | 0.919 | 0.902 | 0.885 | 0.878 | 0.974 | 0.913 | 0.894 | 0.913 | 0.927 |
| | 1.0 | 3 | 0.986 | 0.992 | 0.925 | 0.894 | 0.869 | 0.866 | 0.969 | 0.887 | 0.894 | 0.914 | 0.921 |
| | | 8 | 0.990 | 0.992 | 0.928 | 0.907 | 0.876 | 0.865 | 0.977 | 0.904 | 0.893 | 0.918 | 0.926 |
| | | 21 | 0.986 | 0.991 | 0.919 | 0.905 | 0.888 | 0.869 | 0.976 | 0.911 | 0.896 | 0.916 | 0.927 |

Table 5.1. Accuracy of landmark $k$-NN over MNIST with different algorithmic choices. **KM** is the number of Karcher means used in each image class. **Affinity** is either *linear* or of quadratic exponential decay with given sigma. $R^p$ list the number of dimensions in the spectral embedding projection. Columns $0$ to $9$ list the corresponding accuracy, with the total classification accuracy across the test set in the column **total**.

To produce the affinity matrix, we used a linear transformation or a negative exponential transform. The first is defined by $a_{i,j} = 1 - d(f_i, f_j)/D$, where $d(f_i, f_j)$ is the geodesic distance between images $f_i$ and $f_j$ and $D$ is the maximum distance possible. As the affinity matrix is a generalisation of the adjacency matrix in a graph, this choice makes points further away from a give point less "connected" to it, i.e. with affinity close to $0$.

Another popular choice is the exponential[5], e.g. (Weiss, 1999), $a_{i,j} = e^{-d^2(f_i,f_j)/\sigma^2}$ where $\sigma$ is a free parameter. This choice again makes close objects having an affinity close to $1$, with values further away approaching zero rapidly depending on $\sigma$. Figure 5.6 shows a histogram of the geodesic distances between observations for the digit "9" in MNIST. From the histogram, one sees that a small value around $0.2$ will make the object graph very disconnected, whereas a value around $1.0$ or even bigger will make the distribution of values closer to $1$. Given a fixed choice

[5] Also known a Gaussian or heat kernel (Yan et al., 2009; Cai and Chen, 2015; Deng et al., 2016).

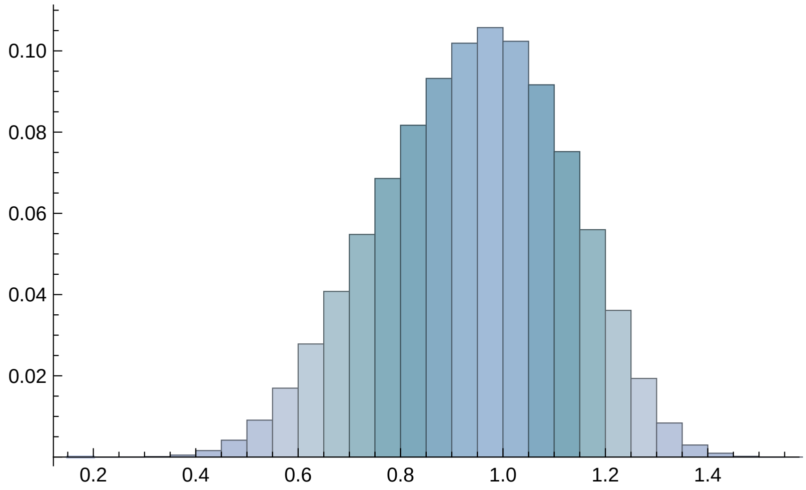


Figure 5.6. Histogram of distances in the hypersphere for the digit "9" in MNIST.

for number of means and projection dimension, there seems to be an optimum value which can be chosen by CV. However, it is worth adding, that the choice of the affinity function has no impact on the $k$-NN classifier as the classification is performed using the metric $d$ on the hypersphere. It only impacts on the quality of the embedding and the subsequent choice of the Karcher means.

Finally, another parameter is the number of dimensions in the embedding space, $R^p$. This has an overall impact on accuracy but it shows different behaviours depending on the choice of sigma. We picked $p = 3, 8, 21$ for this parameter to highlight contrasting behaviours. $p = 3$ gives a baseline and data cab be visualised as we did in Figure 5.4 and 5.5. On the other hand, the performance of the $k$-means algorithm that we use to pick the clusters degrades with increased number of dimensions. In these datasets, a value of $p = 8$ seemed to work best.

The $k$-NN classifier itself depends on the choice of $k$. We obtained best results with $k = 1$, i.e. by picking the class of the nearest neighbour among the selected Karcher means, a similar result to the foundational reports Fix and Hodges (1951, 1952) and as done in Yan et al. (2009).

When using the maximum number of 175 landmarks per class, an embedding into $R^p$ with $p = 8$ and $\sigma = 0.4$ the overall accuracy of this landmark-based classification was $93.3\%$. It is well-known that deep neural network methods outperform all other classifiers in this dataset. However, excluding this approach, Xiao et al. (2017) presents other 13 approaches based on several classifiers found in the Python machine learning library `scikit-learn` with different hyper-parameters for a total of 129 experiments. Our approach sits at the 67% percentile of that table, out performing several classifier implementations in the package[6]. However, the following algorithms, gradient boosting, $k$-nearest neighbours, multilayer perceptron classifier, random forest and support vectors[7], still have better performance than ours under certain configurations. For details on these algorithms, see Pedregosa et al. (2011) and the code repository referenced there. Our results are found in B.6 in the appendix.

On the other hand, results for Fashion-MNIST, shown in Table B.5 in the appendix, are less than bright. The average accuracy for the various `scikit-learn` classifiers reported in the same paper for Fashion-MNIST is $80.8\%$. The best configuration of our approach is near with $80.2\%$, better still than decision trees, decision trees, Gaussian naive Bayes, stochastic gradient descent classifier, logistic regression and passive-aggressive classifiers and percentron[8], but now sitting at an average performance. It is possible that improvements to our approach can improve this figure but it is worth mentioning that although the Fashion-MNIST dataset attempted to bring more realistic examples for an image classifier, it fell short due to the low resolution quality of the final pictures. It can be seen in one confusion matrix among the numerous runs, that there are two major clusters of images, one for dresses and bags, and another for shoes. It is likely that starting with better quality images, other dimensions within the data could have been discovered, either by the algorithm or by some pre-processing, which is not possible for such low quality images.

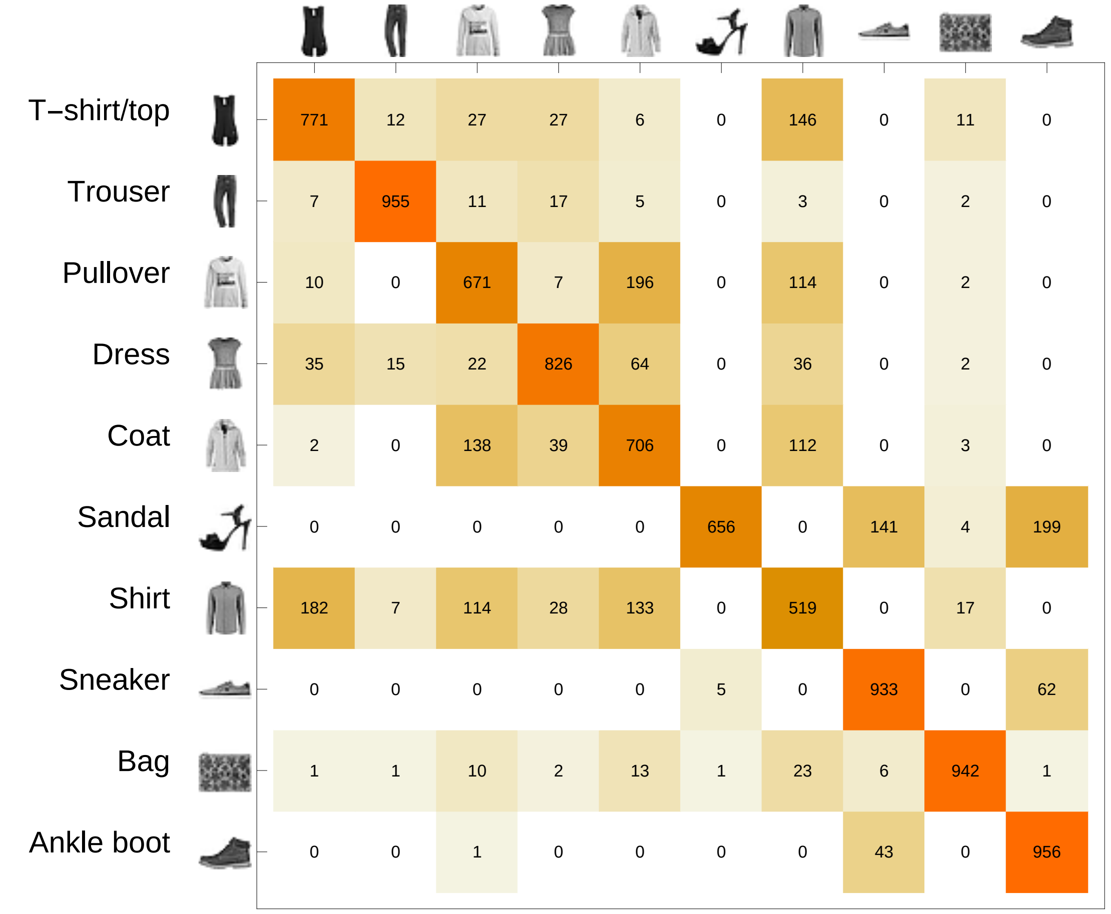
Increasing the number of Karcher means improves the accuracy to the average reported above for this dataset, but it also makes the classification run slower. One

[6] Namely `DecisionTreeClassifier`, `ExtraTreeClassifier`, `GaussianNB`, `LinearSVC`, `LogisticRegression`, `PassiveAggressiveClassifier`, `Perceptron` and `SGDClassifier`

[7] In code, identified by `GradientBoostingClassifier`, `KNeighborsClassifier`, `MLPClassifier`, `RandomForestClassifier` and `SVC`

[8] Specific names are: `DecisionTreeClassifier`, `ExtraTreeClassifier`, `GaussianNB`, `SGDClassifier`, `LogisticRegression` and `PassiveAggressiveClassifier` and `Perceptron`

way to alleviate this problem is to have different number of landmarks per class. For instance, in Figure 5.7, the accuracy for the classes "Trouser", "Sneaker", "Bag" and "Ankle boot" is around 95% . Therefore, one does not need to increase the number of landmarks for these classes and can instead add more landmarks for the harder to classify "Shirt", "Sandal" and "Pullover". In the case of Fashion-MNIST, starting with our best classifier at 175 landmarks per class, one can increase the number up to around 800 for some classes, reaching an average accuracy of 81%, slightly above the average reported by the creators of Fashion-MNIST as mentioned previously.



| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| T–shirt/top | 771 | 12 | 27 | 27 | 6 | 0 | 146 | 0 | 11 | 0 |
| Trouser | 7 | 955 | 11 | 17 | 5 | 0 | 3 | 0 | 2 | 0 |
| Pullover | 10 | 0 | 671 | 7 | 196 | 0 | 114 | 0 | 2 | 0 |
| Dress | 35 | 15 | 22 | 826 | 64 | 0 | 36 | 0 | 2 | 0 |
| Coat | 2 | 0 | 138 | 39 | 706 | 0 | 112 | 0 | 3 | 0 |
| Sandal | 0 | 0 | 0 | 0 | 0 | 656 | 0 | 141 | 4 | 199 |
| Shirt | 182 | 7 | 114 | 28 | 133 | 0 | 519 | 0 | 17 | 0 |
| Sneaker | 0 | 0 | 0 | 0 | 0 | 5 | 0 | 933 | 0 | 62 |
| Bag | 1 | 1 | 10 | 2 | 13 | 1 | 23 | 6 | 942 | 1 |
| Ankle boot | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 43 | 0 | 956 |

Figure 5.7. An example confusion matrix for the landmarks-based $k$-th NN classifier over the Fashion-MNIST dataset.

Another alternative is to preprocess the images. Preprocessing to elicit certain image features (e.g. Nixon and Aguado (2019); Parker (2010)) can improve classification. Of course, it can also decrease accuracy, but negative results are most likely not reported in the literature. The extend to which this is a science rather than tinkering is a subject of much debate in the machine learning community, specially in regards to methods derived from convolutional networks. Instead, what we would like to point out in here is that the wavelet expansion of the images in the hypersphere can also be used to alter the underlying metric and that it can also improve the performance of the classifier - of course, depending on the adjustment.

We illustrate that by the simple example of *blurring* the images by truncating the discrete wavelet transform up to a certain level. Recall that the distance in the hypersphere is defined as $d(x_i, x_j) = \arccos\langle x_i, x_j\rangle$. The reason behind this is quite simple, the cosine of the angle of two *vectors* in the embedding space is

$$\cos\alpha = \frac{\langle x_i, x_j\rangle}{\|x_i\|\,\|x_j\|}. \tag{5.4}$$

By being on the hypersphere, we are guaranteed that $\|x_i\| = \|x_j\| = 1$, hence the metric above. But if $x_i$ is a density, we can calculate the wavelet expansion of its square root and, therefore, its norm can be directly calculated by 3.10. In the case of images, this reduces to an operation over its discrete wavelet transform, namely the sum of the squares of all the coefficients.

So, what happens if we restrict the wavelet expansion to a certain level and use that in calculating the distance? In Figure 5.8, we have some examples of the Fashion-MNIST dataset before and after the wavelet expansion is truncated to two levels. As can be seen, it amounts to some sort of averaging or blurring of the images. Overall, with this particular transformation, the results (Table B.5) improved on average by 1% across the board.

In a very generic way, if we call the set of kept indexes $\mathcal{L}$, then the metric effectively used is

$$\langle x_i, x_j\rangle = \sum_{\lambda\in\mathcal{L}} \beta_\lambda^{(i)}\beta_\lambda^{(j)} \tag{5.5}$$

$$\|x_i\|^2 = \langle x_i, x_i\rangle \tag{5.6}$$

$$d(x_i, x_j) = \arccos\frac{\langle x_i, x_j\rangle}{\|x_i\|\,\|x_j\|}, \tag{5.7}$$

where $\beta^{(i)}$ and $\beta^{(j)}$ are the corresponding coefficients in the discrete wavelet ex-

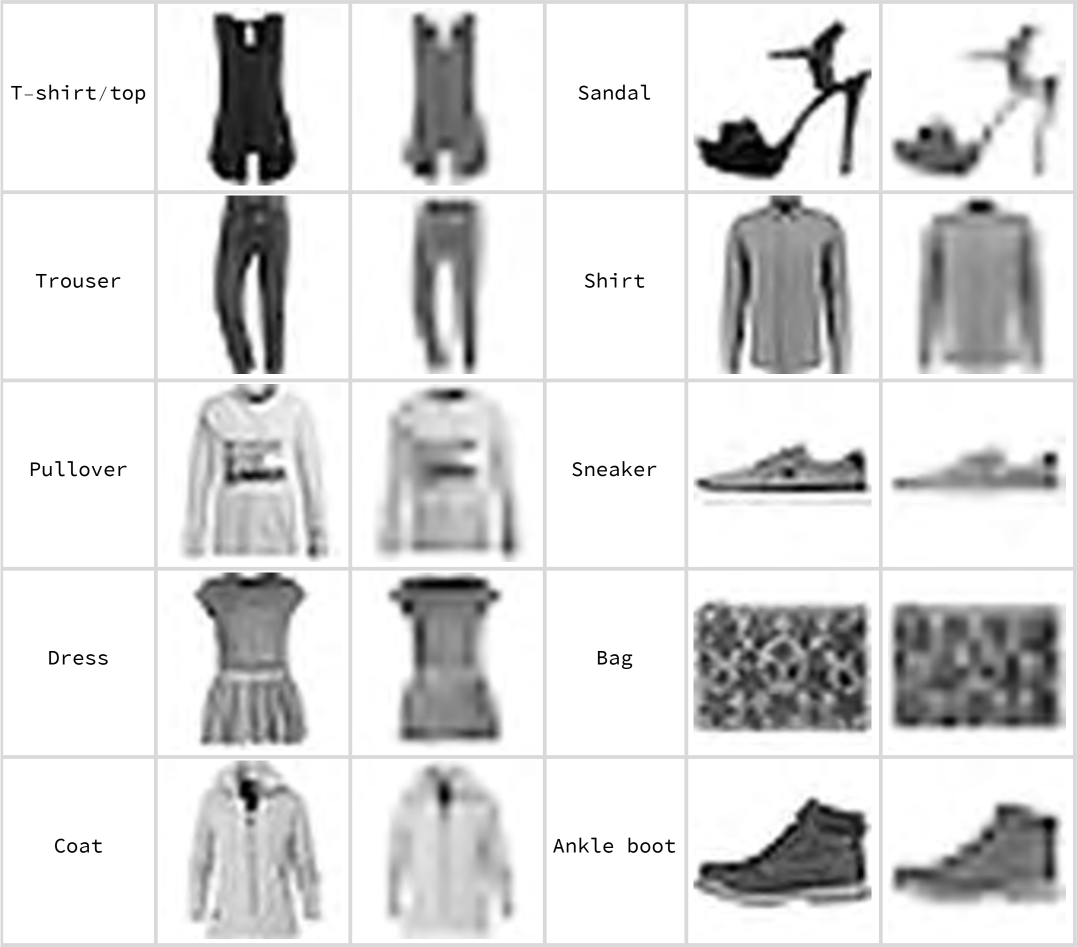


Figure 5.8. Effect of restricting the wavelet expansion of images in the Fashion-MNIST dataset. For each category, the original is on the left and the transformed image is on the right, here restricted to two levels in the wavelet expansion using the Symlet wavelet with 3 vanishing moments.

pansion for $x_i, x_j$. Undoubtedly, further refinements of the above and other alternative approaches that profit from the Riemannian manifold structure of the space of densities are possible but time precluded us from exploring more and, as we said, it is rather an area of future research. Potential research work is discussed in Chapter 6.

# Chapter 6

# Discussion

Penev and Dechevsky (1997) suggested an elegant construction of a wavelet estimator of the square-root of a univariate PDF in order to deal with negativity issues in an automatic way. However, as it was based on spacings, their idea could not be easily generalised beyond the univariate case. This thesis provides such an extension, essentially making use of nearest neighbour balls, the "*probabilistic counterpart to univariate spacings*" (Ranneby et al, 2005) in higher dimensions. The asymptotic properties of the estimator were obtained. It always attains the optimal rate of convergence in Mean Integrated Square Error in $d = 1$ and $d = 2$ dimensions, in dimensions up to $d = 4$ for reasonably smooth densities, and in all dimensions for 'rough' densities. In practice, the estimator was seen to be on par with the classical wavelet estimator, while automatically producing estimates which are always *bona fide* densities.

In addition, we presented a fully developed thresholding scheme of practical use. This was based on the Hellinger distance combined with LOO-CV. Because the shape-preserving estimator was based on the square root, the Hellinger distance was a natural metric to use for thresholding. In addition, we used again a particular link between nearest neighbour balls and the underlying distribution, making possible to calculate an empirical Bhattacharyya coefficient using LOO-CV. With this technique, we developed straightforward algorithms to select various hyperparameters in the wavelet construction.

First, a global resolution level can be easily determined. As the estimator's performance decays fairly quickly when under-smoothing occurs, this optimal resolution level turned out to be the recommended maximum resolution. This is interesting and worth exploring from a theoretical perspective in future research. As picking this resolution level is similar to finding the bandwidth in kernel density estimation, our implementation has the same shortcoming described in Hall and Penev (2001) of restricting this bandwidth to the values $\left(\frac{1}{2}\right)^j$. At the expense of more computation time, it is in principle possible to extend our method to the continuum by estimating an additional factor $p$ that scales each level by $p\,2^j$.

Another parameter is the initial resolution, which we found to have little impact, solely affecting the thresholding itself. As this is $O(log(n))$, a practical low value between 1 to 3 can be used. To threshold the beta coefficients, we again based our method on this empirical Bhattacharyya coefficient. We used hard thresholding to simplify our algorithms, although a soft-thresholding variant is also possible. Within hard thresholding, we presented three ways to calculate the threshold, two based on the traditional presentations of Donoho and Johnstone (1996) and Delyon and Juditsky (1996), and a novel approach involving the empirical variance of the beta coefficients. As the first two methods implicitly rely on a global variance or a level-by-level variance, we found our approach out-performing the former two in most simulations. Indeed, there is a vast literature on block thresholding, which we mentioned briefly in Subsection 2.3.2, that aims to find the right balance between the global and local view of a threshold (Chicken and Cai, 2005). This has already been applied to shape-preserving estimators based on the square root of the density, like (Brown et al., 2010; Shirazi and Chaubey, 2019). Hence, it is certainly worth exploring a block thresholding variant of our novel jackknife-based thresholding strategy.

Of particular interest for future research is the connection between our approach and MDL, already studied for HD in (Peter et al., 2017). Indeed, sparse representations in wavelet bases were popularised by D. Donoho under the name of "Compressed Sensing" in early 2000 (Donoho, 2006), and viewed as a model selection approach competing with MDL, the Bayesian information criterion, and other techniques, they are still an active area of research, e.g.Rissanen (2000); Roos et al. (2009); Adler et al. (2017); Dwork et al. (2020). Finally, we also tackled the difficult task of picking an appropriate wavelet basis, in particular its number of vanishing moments. This is a particularly difficult question when dealing with real-life data problems. We applied our methodology to the Old Faithful geyser dataset and were able to suggest a wavelet basis based on analysis of the optimisation curve and the Bhattacharyya coefficient results. We believe the methodology we sketched can be further formalised as part of the scientific method as advocated by authors like Box (1976); Blei (2014) and references therein.

As seen in numerical experiments on Section 4.4, we also implemented an extension of the algorithm using biorthogonal wavelets by a slight variation of the initial formulation. The use of biorthogonal filters to estimate the square root of a density has been previously demonstrated in the DSP literature, e.g. Yoon and Vaidyanathan (2004); Kaushik et al. (2014), but using histograms as first approximation. In our case, the adaptation to the biorthogonal setting is straightforward. Equations (3.6) and (3.7) are defined now using the dual basis

$$\hat{\alpha}_{j,z} \doteq \frac{\Gamma(k)}{\Gamma(k+1/2)} \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \tilde{\phi}_{j,z}(X_i) \sqrt{V_{(k);i}} \tag{6.1}$$

$$\hat{\beta}^{(q)}_{j,z} \doteq \frac{\Gamma(k)}{\Gamma(k+1/2)} \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \tilde{\psi}^{(q)}_{j,z}(X_i) \sqrt{V_{(k);i}}, \tag{6.2}$$

with dual coefficients $\hat{\tilde{\alpha}}_{j,z}$ and $\hat{\tilde{\beta}}_{j,z}$ using $\tilde{\tilde{\varphi}} = \varphi$ and $\tilde{\tilde{\psi}} = \psi$ respectively. The estimator $\mathring{g}_{J_0,J}(x)$ is defined using the standard basis as before. Finally, the norm can also be computed easily (Casazza and Kutyniok (2012), Prop 1.15(iii) with $x = y$) using coefficients and their duals

$$\|\mathring{g}_{J_0,J}\|^2 = \sum_{z \in Z^d} \hat{\alpha}_{J_0,z} \hat{\tilde{\alpha}}_{J_0,z} + \sum_{j=J_0}^{J} \sum_{z \in Z} \sum_{q \in Q_d} \hat{\beta}^{(q)}_{j,z} \hat{\tilde{\beta}}_{j,z}, \tag{6.3}$$

making the extension with norm equal to $1$ straightforward.

Last but no least, we established the asymptotic optimality of the maximum resolution and hard thresholding using the tools of Hall (1983a); Marron and Härdle (1986) and Marron (1987). Here, we adapted their quite generic results on delta estimators to the resolution level and thresholding problems, proving the suitability of the algorithms under somewhat general assumptions. In this regard, it is well-known that thresholding wavelet coefficients in the classical case gives better estimates in general Besov spaces (Donoho et al., 1996; Donoho and Johnstone, 1998). Although we limited ourselves here to Sobolev spaces, it is possible that our results can be extended there.

In all simulations and experiments, we employed multidimensional wavelets based on the tensor product. This performed relatively well, showing remarkable sparsity in our simulations as seen in supplementary tables (Appendix B.2). However, it is well-known that the number of coefficients required by the tensor product is $O(n)$ in the general case, lacking the ability to adapt to anisotropies in the data (Starck et al., 2010)[1]. Therefore, of particular interest is the topic of the construction of wavelet basis in a multivariate setting beyond the limitations of the isotropic tensor product we used in this work. Although classic literature on wavelet density estimation focuses on the orthogonal case, e.g. Kerkyacharian and Picard (1993); Donoho and Johnstone (1996); Donoho et al (1995); Donoho et al. (1996); Vannucci (1995); Vannucci and Vidakovic (1997); Penev and Dechevsky (1997); Härdle et al (1998), the biorthogonal setting is the most general wavelet construction (Meyer, 1992; Daubechies, 1992; Sweldens, 1996), and the one advanced in the current literature on wavelet-based regression and DSP - specially in the context of multivariate data, e.g. ridgelets (Candès, 1998), curvelets (Candès and Donoho, 2005), shearlets (Labate et al., 2005) and $\alpha$-molecules (Grohs et al., 2013). As mentioned above, we made a first step by extending our construction to the biorthogonal case - again using a tensor product of biorthogonal wavelets to go into multiple dimensions. Thus, it is natural to postulate that a framework based on the Hellinger distance and the Bhattacharyya coefficient, as the one advanced in this thesis for the shape-preserving density estimator based on orthogonal, evenly spaced wavelets, could have natural extensions in modern

[1] It is worth adding, that we extended our algorithm a little in the multi-level case by splitting the betas into different levels, allowing the *sorting* to pick up different thresholds for different levels. This gives the method more degrees of freedom as one has in a cross-validated bandwidth selection in KDE. For lack of space and time, these results are not presented here. In addition, this construction is still within the limited setting of a tensor product and probably not as relevant as future research as the multi-variate, anisotropic methods mentioned in here.

frameworks like $\alpha$-molecules (Grohs et al., 2013) and irregular grids and scales, e.g. (Vanraes et al., 2001; Aldroubi et al., 2004; Kittipoom et al., 2011). Indeed, as we noted in Chapter 3 and Chapter 4, discretisation of scale produces jumps between levels that, one wishes, could be dealt with using arbitrary scale factors or initial resolutions (Hall and Penev, 2001). More over, irregular grids, a natural setting for sampled data, pose the problem of scale-mixing due to interactions between uneven grids and discrete scaling (Jansen, 2003). Certainly, this is an immediate area of research where we believe our construction could be exploited in more generality. On this, it is worth mentioning the progress made on kernel-based methods beyond (2.1) and it would only be fair to compare any improved estimator against other state-of-the-art approaches. Weighted KDE, where each kernel term has an arbitrary weight $\omega_i$, has a long history, even with links to nearest neighbours (Breiman et al., 1977; Abramson, 1982). More recently, motivated by the demands of big data, those weighted formulations have led naturally to $\ell_1$ regularised estimators (Bunea et al., 2007) and interest in sparse KDE (Girolami and He, 2003). So, comparing anisotropic variants would only make sense against algorithms like those of (Deng et al., 2008; Hong et al., 2008; Kristan et al., 2011; Doosti and Hall, 2016).

Apart from above, there is one challenge remaining that can be motivated from an algorithmic perspective and that could potentially establish deeper links between the selected metric, the Hellinger distance, and thresholding itself. In our algorithm, a thresholding formula, being either the simple threshold, a level-adjusted version or our data-driven variance method, becomes embedded into a particular sorting of the beta coefficients (Figure 4.1) done in advance over a set of selected betas. Here, the selection of a sorting algorithm and hence the corresponding threshold formula was independent of the optimisation target, the Bhattacharyya coefficient (equivalently, the Hellinger distance). If instead of sorting in advance, those remaining betas are sorted based on the optimisation target itself, this would be equivalent to a greedy optimisation. The not so obvious reason on why this fails is that, on close inspection, many of the optimisation curves plotted in sub section 4.4.3, Real data: Old Faithful geyser, have multiple local maxima. For instance, although Figure 4.16 has a global maximum, one can notice several tiny valleys appearing in the way to the top as one travels from right to left. An extreme case of this is the failed wavelet basis choice of Figure 4.18, (c) and (d). Modified "greedy" methods like those of Starck et al. (2010) could potentially be applied here and are another possible focus of research that, in addition, could establish a deeper link between the thresholding approach and the optimisation target we used. It is our hope that taking these constructions into modern frameworks like $\alpha$-molecules (Labate et al. (2013)) can push even further the applicability of orthogonal and biorthogonal series decompositions in a wider range of statistical problems.

In regards to this, we ventured into applying some of the techniques developed above to image analysis. Similar to Peter and Rangarajan (2008); Peter et al. (2017), we treated images as densities using the square root representation and attempted a couple of well-known image classification problems, MNIST and Fashion-MNIST, adapting the use of $k$-NN to this setting. The state of the art on these problems is the use of connectionist methods based on the neural networks framework, a nonparametric regression technique[2]. As a full set of nearest neighbours is impractical for this kind of "big data" problems, we developed a technique to pick representatives to calculate those based on spectral embedding and Karcher means. The reason for this is that we treated grey scale images as square root densities and in doing so we made them part of the hypersphere, a well-known Riemannian manifold. The advantage of using the hypersphere is that its geometry is well known and a number of constructions from Riemannian geometry have closed form formulae.

Although the performance obtained was slightly better than average, we believe there is room to extend the proposed construction. As mentioned in Chapter 5, there are a number of options to use a nonparametric method like wavelets while exploiting at the same time the simplicity of the hypersphere, thus providing a level of explain-ability. For instance, an easy generalisation of (5.5) is to consider a weighting function $w_\lambda$ associated with the index $\lambda \in \mathcal{L}$ in the calculation of the

[2] The practitioner is essentially free to chose the *architecture*, number of nodes and their relationships, and in this sense is essentially nonparametric.

inner product

$$\langle x_i, x_j \rangle = \sum_{\lambda \in \mathcal{L}} w_\lambda \beta_\lambda^{(i)} \beta_\lambda^{(j)}. \tag{6.4}$$

This can be seen as favouring a certain direction in the tangent space, effectively making the neighbourhoods in the hypersphere like ellipses instead of circles (Figure 6.1). Thus, assigning more weight $w_\lambda$ in (6.4) to coefficients corresponding to components $\varphi_{j,z}(x)\psi_{j,z}(y)$ in the tensor product will favour the coordinate $x$ over $y$ in calculating the distance, i.e. will make horizontal lines more important to calculate differences[3]

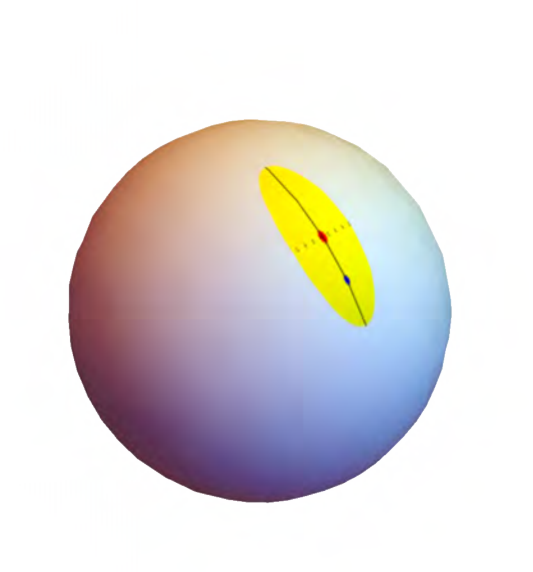


Figure 6.1. A *elliptical* neighbourhood in the hypersphere by modifying the distance metric by weights

Today, there is a huge variety of semi-structured data, images, shapes, trees, graphs, etc. that require sophisticated statistical tools beyond the methods developed in the last part of the twentieth century. The most popular algorithms nowadays are connectionist approaches based on neural networks, e.g. methods grouped under the so called deep learning, convolutional networks and similar. As intrinsically nonparametric methods, they have been very successful in applications to these kind of data but are considered "back boxes" with little room for

[3] Although not the infinite dimensional hypersphere of densities, slightly related work can be found in the now rich literature of wavelets on the sphere $\mathbb{S}^2$ and more generally on $\mathbb{S}^n$, .e.,g (Antoine et al., 2002; Starck et al., 2006).

interpretability. More importantly perhaps, there has been criticism about the nature of advancements in the field, with some authors arguing that there have not been real improvements in the last 10-15 years, at least in certain problems, when scrutinised under the same metrics and experimental setups (Sculley et al., 2018; Yang et al., 2019; Lin, 2019; Blalock et al., 2020).

On the other hand, much less press have been devoted to the nascent work of statistics in Riemannian manifolds and "manifold data" or "object oriented statistics", e.g. Pennec (2006); Wang and Marron (2007); Grohs and Wallner (2009); Charon and Trouvé (2013); Marron and Alonso (2014). There is also an extensive literature on wavelets in Riemannian manifolds, e.g. (Dahlke, 1994; Geller and Mayeli, 2009; Pesenson, 2015) and references therein. We hope that the methods and algorithms presented in this thesis, extending the nonparametric method of density estimation using wavelets combined with the geometry of data, nearest neighbours and Riemannian manifolds, find a novel point of convergence in these recent areas of research.

# Appendix A

# Proofs

## Preliminaries

First some preliminary concepts and technical results are presented.

For any convex and compact $C \subset \mathbb{R}^d$, let $\partial C$ denote its boundary. For $\eta > 0$, define the $\eta$-belt of $C$ as

$$C^{(<\eta)} = \left\{ x \in C \colon \inf_{y \in \partial C} \| y - x \| < \eta \right\},$$

the set of points in $C$ within Euclidean distance $\eta$ or less from $\partial C$. Also, we call $C^{(>\eta)} = C \backslash C^{(<\eta)}$ the $\eta$-interior of $C$.

Fix $x \in C$, call $B_x(r)$ the ball of radius $r$ centered at $x$ and $\mu(B_x(r)) = c_0 r^d$ its volume ($\mu$ is the Lebesgue measure on $\mathbb{R}^d$, $c_0 = \pi^{d/2}/\Gamma(d/2+1)$). Results in Percus and Martin (1998) and Evans et al. (2002) show that the following two properties hold for any compact and convex set $C \subset \mathbb{R}^d$:

**C1**. There exists $c_2 > 0$, independent of $x \in C$, such that for $r < \sup_{x,y \in C} \| x - y \|$, $\mu \left( B_x (r) \cap C \right) \geq c_2 r^d$ ;

**C2**. There exist constants $\lambda > 0$ and $c_3 > 0$ such that for all $0 < \eta < \lambda$, $\mu \left\{ C^{(<\eta)} \right\} < c_3 \eta$.

The following technical lemma will be used repeatedly in the proofs below.

**Lemma 1.** *Let $\mathcal{X} = \{X_1, \ldots, X_n\}$ be a random sample from a distribution $F$ admit-*

*ting a density $f$ supported on $C \subset \mathbb{R}^d$ satisfying Assumption 3.2.3. Let $R_{(k);i}$ be the distance between $X_i$ and its $k$th nearest neighbor in the sample, as defined in Section 3.1.1. Let $\phi : \mathbb{R}^d \to \mathbb{R}$ be bounded on $C$ and $a > 0$ such that $\int_{\mathbb{R}^d} \phi(x) f(x)^{1-a} dx < \infty$. Then, for all $i \in \{1, \ldots, n\}$, as $n \to \infty$,*

$$\mathbb{E}\left\{\phi(X_i) R^{ad}_{(k);i}\right\} = \mathbb{E}\left\{\phi(X_i) \mathbb{E}\left(R^{ad}_{(k);i} | X_i\right)\right\}$$
$$= \frac{1}{n^a} \frac{\Gamma(k+a)}{\Gamma(k)} \frac{1}{c_0^a} \left\{\int_{\mathbb{R}^d} \phi(x) f(x)^{1-a} \,\mathrm{d}x + O\left(n^{-1/d}\right)\right\}.$$

*Proof.* Call

$$\omega_x(r) = \int_{B_x(r)} f(z) \,\mathrm{d}z,$$

the probability that the random variable $X \sim F$ falls in $B_x(r)$, and set $\omega_i(r) = \omega_{X_i}(r)$ when referring to the ball centered at one particular observation $X_i$ from the sample. Let $F_{(k);i}$ be the distribution function of $R_{(k);i}$ for fixed $X_i$, that is, $F_{(k);i}(r) = \Pr(R_{(k);i} \leq r | X_i)$. With $X_i$ fixed, Lemma 4.1 in Evans et al. (2002) writes

$$\mathrm{d}F_{(k);i}(r) = k\binom{n-1}{k}\omega_i(r)^{k-1}(1-\omega_i(r))^{n-k-1} \,\mathrm{d}\omega_i(r).$$

Hence

$$\mathbb{E}\left(R^{ad}_{(k);i} | X_i\right) = k\binom{n-1}{k} \int_0^{c_1} r^{ad}\omega_i(r)^{k-1}(1-\omega_i(r))^{n-k-1} \,\mathrm{d}\omega_i(r).$$

Since $f$ is positive on $C$ and $C$ is convex, $\omega_i(r)$ is strictly increasing for $r \in [0, r_0]$ for some $r_0$, and $\omega_i(r) \equiv 1$ for $r_0 \leq r$. Writing $h_i(\omega)$ for the inverse function $\omega_i^{-1}$ (where it exists), a change of variable yields

$$\mathbb{E}\left(R^{ad}_{(k);i} | X_i\right) = k\binom{n-1}{k} \int_0^1 h_i(\omega)^{ad}\omega^{k-1}(1-\omega)^{n-k-1} \,\mathrm{d}\omega.$$

Define $\delta_n = n^{-1/d}$, and break this expectation down into

$$\mathbb{E}\left(R^{ad}_{(k);i} | X_i\right) = k\binom{n-1}{k} \int_0^{\omega_i(\delta_n)} h_i(\omega)^{ad}\omega^{k-1}(1-\omega)^{n-k-1} \,\mathrm{d}\omega$$
$$+ k\binom{n-1}{k} \int_{\omega_i(\delta_n)}^1 h_i(\omega)^{ad}\omega^{k-1}(1-\omega)^{n-k-1} \,\mathrm{d}\omega$$
$$= k\binom{n-1}{k} \int_0^{\omega_i(\delta_n)} h_i(\omega)^{ad}\omega^{k-1}(1-\omega)^{n-k-1} \,\mathrm{d}\omega + O(n^{-b})$$

for all $b > 0$, uniformly in $X_i$, as per Lemma 5.3 of Evans et al. (2002).

Now, with $h_x = \omega_x^{-1}$, see that

$$\begin{aligned}
&\mathbb{E}\left\{\phi\left(X_i\right)\mathbb{E}\left(R^{ad}_{(k);i}|X_i\right)\right\} \\
&= \int_C \phi(x)\left\{k\binom{n-1}{k}\int_0^1 h_x(\omega)^{ad}\omega^{k-1}(1-\omega)^{n-k-1}\,\mathrm{d}\omega\right\} f(x)\,\mathrm{d}x \qquad \text{(A.1)}\\
&= \int_C \phi(x)\left\{k\binom{n-1}{k}\int_0^{\omega_x(\delta_n)} h_x(\omega)^{ad}\omega^{k-1}(1-\omega)^{n-k-1}\,\mathrm{d}\omega\right\} f(x)\,\mathrm{d}x + O(n^{-b}),
\end{aligned}$$

as $\phi$ and $f$ are bounded on the compact $C$. As $b$ can be taken arbitrarily large, the remainder term can be neglected in front of any term tending to 0 polynomially fast. Hence, (asymptotically) all contribution to the inner integral in (A.1) comes from the set $\omega \in (0, \omega_x(\delta_n))$, that is, when $R_{(k);i}$ is smaller than $\delta_n$.

Now, write (A.1) as

$$\int_C \cdots \mathrm{d}x = \int_{C^{(>\delta_n)}} \cdots \mathrm{d}x + \int_{C^{(<\delta_n)}} \cdots \mathrm{d}x \doteq (I) + (II)$$

with $C^{(>\delta_n)}$ and $C^{(<\delta_n)}$ the $\delta_n$-interior and $\delta_n$-belt of $C$ as defined above.

**Integral** $(I)$: $\int_{C^{(>\delta_n)}} \cdots \mathrm{d}x$, hence $x \in \delta_n$-interior and the distance from $x$ to $\partial C$ is at least $\delta_n$. Hence for all $r \leq \delta_n$, $B_x(r) \cap C = B_x(r)$. The first mean value theorem for definite integrals establishes the existence of $\xi_1 \in B_x\left(r\right) \subset C$ such that

$$\omega_x\left(r\right) = \int_{B_x(r)} f\left(z\right)\mathrm{d}z = f\left(\xi_1\right)\mu\left(B_x\left(r\right)\right) = f(\xi_1)c_0 r^d. \qquad \text{(A.2)}$$

By the mean value theorem, there is a $\xi_2$ between $x$ and $\xi_1$, hence $\xi_2 \in B_x\left(r\right) \subset C$, such that $f\left(\xi_1\right) = f\left(x\right) + \nabla f\left(\xi_2\right)'\left(x - \xi_1\right)$. Because $\xi_1 \in B_x(\delta_n)$ and $\|\nabla f\left(\xi_2\right)\| < M$ for an absolute constant $M$ (the partial derivatives of $f$ are uniformly bounded on $C$ by Assumption 3.2.3), we have $|f\left(\xi_1\right) - f\left(x\right)| < \delta_n M$ and hence $f\left(\xi_1\right) = f\left(x\right) + O\left(\delta_n\right)$. Substitution in (A.2) gives $\omega_x\left(r\right) = \left\{f\left(x\right) + O\left(\delta_n\right)\right\}c_0 r^d$. As $f$ is bounded from below, this means that, as $n \to \infty$,

$$h_x(\omega) = \left\{\frac{\omega}{c_0 f(x)}\right\}^{1/d}\{1 + O(\delta_n)\},$$

where the $O(\delta_n)$-term holds uniformly in $x$ and $\omega$. This can be substituted in the inner integral of (A.1), and we obtain

$$\begin{aligned}\int_{C^{(>\delta_n)}} \phi(x)\left\{k\binom{n-1}{k}\int_0^1 h_x(\omega)^{ad}\omega^{k-1}(1-\omega)^{n-k-1}\,\mathrm{d}\omega\right\} f(x)\,\mathrm{d}x\\ = \frac{\Gamma(n)}{\Gamma(k)\Gamma(n-k)}(1+O(\delta_n))\int_{C^{(>\delta_n)}} \frac{\phi(x)f^{1-a}(x)}{c_0^a}\,\mathrm{d}x\\ \int_0^1 \omega^{a+k-1}(1-\omega)^{n-k-1}\,\mathrm{d}\omega\\ = \frac{\Gamma(n)}{\Gamma(k)\Gamma(n-k)}\frac{\Gamma(k+a)\Gamma(n-k)}{\Gamma(n+a)}(1+O(\delta_n))\int_{C^{(>\delta_n)}} \frac{\phi(x)f^{1-a}(x)}{c_0^a}\,\mathrm{d}x\\ = \frac{\Gamma(n)}{\Gamma(n+a)}\frac{\Gamma(k+a)}{\Gamma(k)}\frac{1}{c_0^a}(1+O(\delta_n))\int_{C^{(>\delta_n)}} \phi(x)f^{1-a}(x)\,\mathrm{d}x.\end{aligned}$$

Now, given that $f$ is bounded from below and above on $C$, $f(x)^{1-a} \leq a_3$, with $a_3 \equiv \max\{(1/a_1)^{1-a}, a_2^{1-a}\}$, and by **C2** above, $\mu(C^{(<\delta_n)}) < c_3\delta_n$ for $n$ large enough. So,

$$\left|\int_{C^{(<\delta_n)}} \phi(x)f^{1-a}(x)\,\mathrm{d}x\right| \leq \sup_{x\in C^{(<\delta_n)}} |\phi(x)|a_3c_3\delta_n = O(\delta_n),$$

as $n \to \infty$. Therefore,

$$\int_{C^{(>\delta_n)}} \phi(x)f^{1-a}(x)\,\mathrm{d}x = \int_C \phi(x)f^{1-a}(x)\,\mathrm{d}x + O(\delta_n) = \int_{\mathbb{R}^d} \phi(x)f^{1-a}(x)\,\mathrm{d}x + O(\delta_n).$$

Noting that $\Gamma(n)/\Gamma(n+a) = n^{-a}\{1+O(n^{-1})\} = n^{-a}\{1+O(\delta_n)\}$, we finally get

$$\begin{aligned}\int_{C^{(>\delta_n)}} \phi(x)\left\{k\binom{n-1}{k}\int_0^1 h_x(\omega)^{ad}\omega^{k-1}(1-\omega)^{n-k-1}\,\mathrm{d}\omega\right\} f(x)\,\mathrm{d}x\\ = \frac{1}{n^a}\frac{\Gamma(k+a)}{\Gamma(k)}\frac{1}{c_0^a}\left\{\int_{\mathbb{R}^d} \phi(x)\,f(x)^{1-a}\,\mathrm{d}x + O\left(n^{-1/d}\right)\right\}.\end{aligned} \quad \text{(A.3)}$$

**Integral** $(II)$: $\int_{C^{(<\delta_n)}}\cdots\mathrm{d}x$, hence we can no more assume that $B_x(r) \subset C$. However, as $\sup_{x\in C^{(<\delta_n)}} f(x) \leq \sup_{x\in C} f(x) \leq a_2$ and $\mu(B_x(r)\cap C) < \mu(B_x(r) = c_0r^d$, it holds $\omega_x(r) < a_2c_0r^d$. An upper bound for its inverse is thus $h_x(\omega) \leq (a_2c_0)^{-1/d}\omega^{1/d}$.

Hence,

$$\begin{aligned}
|(II)| &\le \int_{C^{(<\delta_n)}} |\phi(x)| \left\{ k\binom{n-1}{k} \int_0^1 h_x(\omega)^{ad} \omega^{k-1}(1-\omega)^{n-k-1}\,\mathrm{d}\omega \right\} f(x)\,\mathrm{d}x \\
&\le \int_{C^{(<\delta_n)}} |\phi(x)| \left\{ k\binom{n-1}{k} \int_0^1 (a_2 c_0)^{-a} \omega^{a+k-1}(1-\omega)^{n-k-1}\,\mathrm{d}\omega \right\} f(x)\,\mathrm{d}x \\
&= \frac{\Gamma(a+k)}{\Gamma(k)} \frac{\Gamma(n)}{\Gamma(n+a)} (a_2 c_0)^{-a} \int_{C^{(<\delta_n)}} |\phi(x)| f(x)\,\mathrm{d}x \\
&\le \frac{\Gamma(a+k)}{\Gamma(k)} \frac{\Gamma(n)}{\Gamma(n+a)} (a_2 c_0)^{-a} \sup_{x\in C} |\phi(x)| a_2 c_3 \delta_n,
\end{aligned}$$

by C2 above. Thus, as $n \to \infty$,

$$|(II)| \le \frac{\Gamma(k+a)}{\Gamma(k)} O(n^{-a}\delta_n) = \frac{\Gamma(k+a)}{\Gamma(k)} O(n^{-a-1/d}). \tag{A.4}$$

Putting together (A.3) and (A.4) in (A.1), it follows that

$$\mathbb{E}\left(\phi(X_i) R^{ad}_{(k);i}\right) = \frac{1}{n^a} \frac{\Gamma(k+a)}{\Gamma(k)} \frac{1}{c_0^a} \left( \int_{\mathbb{R}^d} \phi(x) f(x)^{1-a}\,\mathrm{d}x + O\left(n^{-1/d}\right) \right),$$

as announced. □

## Proof of Proposition 3.2.1

The proof is given for the coefficients $\hat{\alpha}_{j,z}$. The proof for the coefficients $\hat{\beta}^{(q)}_{j,z}$ is identical.

*Bias:* From (3.6), we have with (3.1),

$$\begin{aligned}
\mathbb{E}(\hat{\alpha}_{j,z}) &= \mathbb{E}\left\{ \frac{\Gamma(k)}{\Gamma(k+1/2)} \frac{1}{\sqrt{n}} \sum_{i=1}^n \varphi_{j,z}(X_i) \sqrt{V_{(k);i}} \right\} \\
&= n^{1/2} \frac{\Gamma(k)}{\Gamma(k+1/2)} \sqrt{c_0}\, \mathbb{E}\left\{ \varphi_{j,z}(X_1) R^{d/2}_{(k);1} \right\}.
\end{aligned}$$

Applying Lemma 1 with $\phi = \varphi_{j,z}$ and $a = 1/2$ yields

$$\mathbb{E}\left\{ \varphi_{j,z}(X_1) R^{d/2}_{(k);1} \right\} = n^{-1/2} \frac{\Gamma(k+1/2)}{\Gamma(k)} \frac{1}{\sqrt{c_0}} \left\{ \int_{\mathbb{R}^d} \varphi_{j,z}(x) \sqrt{f}(x)\,\mathrm{d}x + O(n^{-1/d}) \right\},$$

which gives

$$\mathbb{E}(\hat{\alpha}_{j,z}) = \int_{\mathbb{R}^d} \varphi_{j,z}(x)\sqrt{f}(x)\,\mathrm{d}x + O(n^{-1/d}) = \alpha_{j,z} + O(n^{-1/d}).$$

*Variance:* Lemma 4.6(ii) of Evans (2008) gives an upper bound on the variance of statistics of type $S_n = \sum_{i=1}^n h_{i,n}(\mathcal{X})$, where $h_{i,n}(\mathcal{X})$ is an arbitrary (measurable) function of the sample point $X_i$ and its $k$-nearest neighbors among the sample $\mathcal{X}$. Take here

$$h_{i,n}(\mathcal{X}) = \varphi_{j,z}(X_i)\sqrt{V_{(k);i}}$$

and see that $\hat{\alpha}_{j,z} = \Gamma(k)S_n/\{\Gamma(k+1/2)\sqrt{n}\}$. Lemma 4.6(ii) of Evans (2008) reads

$$\mathbb{V}\mathrm{ar}(S_n) \le 2(n+1)(3+8k^2dc_0)\mathbb{E}\left\{h_{i,n}^2(\mathcal{X})\right\}, \tag{A.5}$$

for $n \ge 16k$. Here,

$$\begin{aligned}\mathbb{E}\left\{h_{i,n}^2(\mathcal{X})\right\} &= \mathbb{E}\left\{\varphi_{j,z}^2(X_i)\,V_{(k);i}\right\}\\ &= c_0\mathbb{E}\left\{\varphi_{j,z}^2(X_i)\,R_{(k);i}^d\right\} = \frac{k}{n}\left\{\int_{\mathbb{R}^d}\varphi_{j,z}^2(x)\,\mathrm{d}x + O(n^{-1/d})\right\},\end{aligned}$$

from Lemma 1 with $\phi = \varphi_{j,z}^2$ and $a = 1$. By definition, $\int_{\mathbb{R}^d}\varphi_{j,z}^2(x)\,\mathrm{d}x = 1$ (orthonormal wavelet basis, Assumption 3.2.2), hence $\mathbb{E}\left\{h_{i,n}^2(\mathcal{X})\right\} = k\{1+O(n^{-1/d})\}/n$. From this and (A.5), we conclude that, as $n \to \infty$,

$$\begin{aligned}\mathbb{V}\mathrm{ar}(\hat{\alpha}_{j,z}) &\le \left\{\frac{\Gamma(k)}{\Gamma(k+1/2)}\right\}^2 \frac{1}{n}2(n+1)(3+8k^2dc_0)\frac{k}{n}\{1+O(n^{-1/d})\}\\ &= k^3\left\{\frac{\Gamma(k)}{\Gamma(k+1/2)}\right\}^2 O(n^{-1}).\end{aligned}$$

□

## Proof of Proposition 3.3.1

From (3.9) we have

$$\begin{aligned}\hat{g}_J(x) &= \sum_{z\in\mathbb{Z}^d} \hat{\alpha}_{J+1,z}\varphi_{J+1,z}(x) = \sum_{z\in\mathbb{Z}^d} \frac{\Gamma(k)}{\Gamma(k+1/2)} \frac{1}{\sqrt{n}} \sum_{i=1}^n \varphi_{J+1,z}(X_i)\sqrt{V_{(k);i}}\,\varphi_{J+1,z}(x) \\ &= \frac{\Gamma(k)}{\Gamma(k+1/2)} \frac{\sqrt{c_0}}{\sqrt{n}} \sum_{i=1}^n R_{(k);i}^{d/2} \sum_{z\in\mathbb{Z}^d} \varphi_{J+1,z}(X_i)\,\varphi_{J+1,z}(x) \\ &= \frac{\Gamma(k)}{\Gamma(k+1/2)} \frac{\sqrt{c_0}}{\sqrt{n}} \sum_{i=1}^n R_{(k);i}^{d/2} K_{J+1}(x, X_i), \qquad \text{(A.6)}\end{aligned}$$

hence

$$\mathbb{E}\{\hat{g}_J(x)\} = \frac{\Gamma(k)}{\Gamma(k+1/2)}\sqrt{n}\sqrt{c_0}\,\mathbb{E}\left\{K_{J+1}(x,X_1)R_{(k);1}^{d/2}\right\}.$$

Lemma 1 with $\phi = K_{J+1}(x,\cdot)$ and $a = 1/2$ establishes that

$$\mathbb{E}\left\{K_{J+1}(x,X_1)R_{(k);1}^{d/2}\right\} = \frac{1}{\sqrt{n}}\frac{\Gamma(k+1/2)}{\Gamma(k)}\frac{1}{\sqrt{c_0}}\left\{\int_{\mathbb{R}^d} K_{J+1}(x,y)\sqrt{f}(y)\,\mathrm{d}y + O(n^{-1/d})\right\},$$

and inspection of the proof of Lemma 1 reveals that the $O(n^{-1/d})$ term holds uniformly in $x \in C$. This means that

$$\mathbb{E}\{\hat{g}_J(x)\} = \int_{\mathbb{R}^d} K_{J+1}(x,y)\sqrt{f}(y)\,\mathrm{d}y + O(n^{-1/d}) = K_{J+1}\sqrt{f}(x) + O(n^{-1/d}),$$

as $n \to \infty$, uniformly in $x \in C$, proving $(i)$. It follows from (A.6) as well that

$$\mathbb{V}\mathrm{ar}\left\{\frac{\Gamma(k+1/2)}{\Gamma(k)}\sqrt{\frac{n}{k^3}}\,\hat{g}_J(x)\right\} = \frac{c_0}{k^3}\,\mathbb{V}\mathrm{ar}\left\{\sum_{i=1}^n h_{i,n}(\mathcal{X})\right\}$$

where here $h_{i,n}(\mathcal{X}) = K_{J+1}(x,X_i)R_{(k);i}^{d/2}$. Lemma 1 with $a = 1$ and $\phi = K_{J+1}^2(x,\cdot)$ yields

$$\mathbb{E}\left\{h_{i,n}^2(\mathcal{X})\right\} = \frac{k}{c_0 n}\left\{\int_{\mathbb{R}^d} K_{J+1}^2(x,y)\,\mathrm{d}y + O(n^{-1/d})\right\}$$

(with again the $O(n^{-1/d})$-term holding uniformly in $x \in C$). Hence, for $n \geq 16k$, Lemma 4.6(ii) of Evans (2008) gives

$$\mathbb{V}\mathrm{ar}\left\{\sum_{i=1}^n h_{i,n}(\mathcal{X})\right\} \leq 2(n+1)(3+8k^2dc_0)\frac{k}{c_0 n}\left\{\int_{\mathbb{R}^d} K_{J+1}^2(x,y)\,\mathrm{d}y + O(n^{-1/d})\right\},$$

whereby

$$\mathbb{V}\mathrm{ar}\left\{\frac{\Gamma(k+1/2)}{\Gamma(k)}\sqrt{\frac{n}{k^3}}\,\hat{g}_J(x)\right\} \leq \text{constant} \times \int_{\mathbb{R}^d} K_{J+1}^2(x,y)\,\mathrm{d}y + O(n^{-1/d}).$$

This establishes $(ii)$. □

## Proof of Theorem 6

The MISE $\mathbb{E}(\|\hat{g}_J - \sqrt{f}\|_2^2)$ can classically be decomposed into the integrated squared bias and the integrated variance:

$$\mathbb{E}\left(\|\hat{g}_J - \sqrt{f}\|_2^2\right) = \|\mathbb{E}\,(\hat{g}_J) - \sqrt{f}\|_2^2 + \mathbb{E}\left(\|\hat{g}_J - \mathbb{E}\,(\hat{g}_J)\,\|_2^2\right). \tag{A.7}$$

For the bias term, it follows from Proposition 3.3.1$(i)$ that

$$\|\mathbb{E}\,(\hat{g}_J) - \sqrt{f}\|_2 \leq \|K_{J+1}\sqrt{f} - \sqrt{f}\|_2 + O(n^{-1/d}).$$

As $f \in B^{m,2}(L)$ implies $\sqrt{f} \in B^{m,2}(L')$ for some $0 \leq L' < \infty$, one can call on (multivariate versions of) Theorem 8.1(ii) and Corollary 10.1 of Härdle et al (1998) to obtain

$$\sup_{f \in B^{m,2}(L)} \|K_{J+1}\sqrt{f} - \sqrt{f}\|_2 \leq \kappa_1 2^{-Jm},$$

for some constant $\kappa_1$. Hence, for $n$ large enough,

$$\sup_{f \in B^{m,2}(L)} \|\mathbb{E}\,(\hat{g}_J) - \sqrt{f}\|_2 \leq \kappa_1 2^{-Jm} + \kappa_2 n^{-1/d}, \tag{A.8}$$

for constants $\kappa_1, \kappa_2 < \infty$.

To evaluate $\int_{\mathbb{R}^d} K_{J+1}^2(x,y)\,\mathrm{d}y$ in the right-hand side of Proposition 3.3.1(ii) we use that

$$\begin{aligned}\int_{\mathbb{R}^d} K_{J+1}^2(x,y)\,\mathrm{d}y &= \int_{\mathbb{R}^d} 2^{2d(J+1)} K^2(2^{J+1}x, 2^{J+1}y)\,\mathrm{d}y \\ &\leq 2^{2d(J+1)} \int_{\mathbb{R}^d} F^2(2^{J+1}(x-y))\,\mathrm{d}y = 2^{(d+1)J} \int_{\mathbb{R}^d} F^2(v)\,\mathrm{d}v,\end{aligned}$$

where Assumption 3.3.1 justifies the inequality. It follows

$$\mathbb{V}\mathrm{ar}\left\{\hat{g}_J(x)\right\} \le \text{constant} \times n^{-1}k^3 \left\{\frac{\Gamma(k)}{\Gamma(k+1/2)}\right\}^2 \left\{2^{dJ}\int_{\mathbb{R}^d} F^2(v)\,\mathrm{d}v + O(n^{-1/d})\right\},$$

which can be integrated over the compact $C$:

$$\begin{aligned}\mathbb{E}\left\{\|\hat{g}_J - \mathbb{E}(\hat{g}_J)\|_2^2\right\} &= \int_{\mathbb{R}^d} \mathbb{V}\mathrm{ar}\left(\hat{g}_J(x)\right)\mathrm{d}x \\ &\le \text{constant} \times n^{-1}k^3 \left\{\frac{\Gamma(k)}{\Gamma(k+1/2)}\right\}^2 \left\{2^{dJ}\int_{\mathbb{R}^d} F^2(v)\,\mathrm{d}v + O(n^{-1/d})\right\}.\end{aligned}$$

Hence, for $n$ large enough, there exists a constant $\kappa_3' < \infty$ such that

$$\mathbb{E}\left\{\|\hat{g}_J - \mathbb{E}(\hat{g}_J)\|_2^2\right\} < \kappa_3' n^{-1}k^3 \left\{\frac{\Gamma(k)}{\Gamma(k+1/2)}\right\}^2 2^{dJ}. \tag{A.9}$$

Plugging (A.8) and (A.9) in (A.7) yields the result. □

## Proof of Theorem 8

Our proofs here follow the same structure as in Marron (1987) but in the setting of HD and our estimator. Naturally, they rely on different intermediate results but the spirit of the argument can be made remarkably similar. From there, we are interested in Theorem 2, which has as corollary what we will recast as our Theorem 8 and Theorem 9. The results there are presented in such generality, that they can be applied easily to both. In Theorem 8, we will make use of the results in Marron (1987) as applied to choice of bandwidth in kernel estimators. In Theorem 9, we will use the same in the context of orthogonal series estimators. We will adopt here the notation $\Lambda_n$ to refer to the parameter space for both cases, resolution level selection and hard thresholding.

Let $\Lambda_n$ be the set of possible resolution levels $J$ of the estimator $\mathring{g}_J$ from a sample of size $n$. Thus, $\lambda$ represents the kernel-equivalent bandwidth parameter for a particular resolution level $J$, i.e. $\lambda = 2^{-j}$. Recall (4.13)

$$\mathring{g}_J(x) = \sum_i^n \boldsymbol{\delta}_J(X_i, x) F_{i,n},$$

where $F_{i,n} = \frac{2}{\sqrt{\pi n}}\sqrt{V_{(1);i}}$. Note that the factor $F_{i,n}$ for delta estimators discussed in Marron (1987) is instead $\frac{1}{n}$ and represents the measure at $X_i$. We align our notation by rewriting $F_{i,n} = \frac{1}{n}\frac{2}{\sqrt{\pi}}\sqrt{V_{(1);i}} = \frac{1}{n}w_{i,n}$, where $w_{i,n} = \frac{2}{\sqrt{\pi}}\sqrt{n\,V_{(1);i}}$. Recall that $\sqrt{n\,V_{(1);i}}$ is the Rayleigh distributed random variable we introduced in (3.3).

Assume our parameter $\lambda$ ranges over a finite set whose cardinality grows at most algebraically fast, i.e. such that $\#(\Lambda_n) \leqslant \mathcal{C}n^{\rho}$ for some $\mathcal{C}, c > 0$. As the resolution level is $O(\log_2 n)$ (Donoho et al., 1996), this is in general satisfied. Also, the assumption on the bias $B(x)$ in Marron (1987), (4.9) corresponds to Proposition 3.3.1 (i).

Let's take $\widehat{\mathcal{B}}_j^{(u)}$ in (4.12) and do

$$\begin{aligned}\widehat{\mathcal{B}}_j^{(u)} &= \sum_i \left|\mathring{g}_j^{(-i)}(X_i)\right| F_{i,n} - \frac{1}{2}\,\|\mathring{g}_j\|^2 \\ &= \sum_i \left|\mathring{g}_j^{(-i)}(X_i)\right| F_{i,n} - \int \mathring{g}_j(x)\sqrt{f(x)}\,\mathrm{d}x + \\ &\qquad \frac{1}{2} - \frac{1}{2}\int \left(\mathring{g}_j(x) - \sqrt{f(x)}\right)^2 \mathrm{d}x.\end{aligned} \tag{A.10}$$

Note the last term is just $HD(\mathring{g}_j^2, f)^2$, which, for succinctness, we shall call $HD_j^2$. Define $d_j^M$ as

$$\begin{aligned}d_j^M &= \mathbb{E}\left[\frac{1}{2}\int \left(\mathring{g}_j(x) - \sqrt{f(x)}\right)^2 \mathrm{d}x\right] \\ &= \mathbb{E}\left[HD_j^2\right].\end{aligned}$$

So, if we had the following

$$\sum_{i=1}^{n} \left|\mathring{g}_j^{(-i)}(X_i)\right| F_{i,n} - \int \mathring{g}_j(x)\sqrt{f(x)}\,\mathrm{d}x - \sum_{i=1}^{n}\sqrt{f(X_i)}F_{i,n} + 1 = o\left(d_j^M\right), \tag{A.11}$$

then by using A.10

$$\widehat{\mathcal{B}}_j^{(u)} - \sum_{i=1}^{n}\sqrt{f}(X_i)F_{i,n} + \frac{1}{2} + HD_j^2 = o\left(d_j^M\right)$$

and we have a result similar to Marron (1987), Theorem 2, in the sense that

$$\lim_{n\to\infty}\sup_{\lambda\in\Lambda_n}\left|\frac{\widehat{\mathcal{B}}_j^{(u)} + HD_j^2 - T}{d_j^M}\right| = 0 \quad a.s \tag{A.12}$$

where $T$, in our case much simpler, is

$$T = \sum_{i=1}^{n}\sqrt{f}(X_i)F_{i,n} - \frac{1}{2}. \tag{A.13}$$

From (A.12), Theorem 8 follows as a corollary (Marron (1987), Corollary 2). Expression (A.11), is a rewrite of Lemma 2 in Marron (1987), which is at the core of the results there. Again, as $HD_j^2 = 1 - BC_j$, where $BC_j$ is the corresponding Bhattacharyya coefficient (4.2), then (A.11) becomes

$$\sum_{i=1}^{n}\left|\mathring{g}_j^{(-i)}(X_i)\right| F_{i,n} - \sum_{i=1}^{n}\sqrt{f(X_i)}F_{i,n} + HD_j{}^2 = o\left(\mathbb{E}\left[HD_j^2\right]\right). \tag{A.14}$$

A quick remark. We have

$$\mathring{g}_J^{(-i)}(x) = \sum_{i'\neq i}^{n}\boldsymbol{\delta}_J(X_{i'},x)F_{i',n}^{(-i)};$$

where the factor $F_{i',n}^{(-i)}$ depends on $S^{(-i)}$ but $\boldsymbol{\delta}_j$ does not. In general, if $X_i$ appears among the $k$ nearest neighbours to $X_{i'}$ in the sum, we can look at the $(k+1)$-th nearest, i.e.

$$V_{(k);i'}^{(-i)} = V_{(k);i'}, \qquad \text{if } i \notin NN_{i'}^{(k)} \tag{A.15}$$

$$V_{(k);i'}^{(-i)} = V_{(k+1);i'}, \qquad \text{if } i \in NN_{i'}^{(k)} \tag{A.16}$$

where $NN_{i'}^{(k)}$ is the set of $k$ nearest neighbours to $X_{i'}$. For $k = 1$ this means that if $X_i$ is the nearest neighbour to $X_{i'}$, then in calculating $\mathring{g}_j^{(-i)}$ we have to use $V_{(2);i'}$ instead. So, $\mathring{g}_J^{(-i)}(x)$ and $\mathring{g}_J(x)$ differ on the points $i'$ that have $i$ as neighbour in $S$ and note that by Lemma 4.2 in Evans (2008), this number is bounded independently of $S$.

Following the argument in Marron (1987), (A.11) is rewritten

$$\sup_{\lambda\in\Lambda_n} n^{-1}(n-1)^{-1}\left|\sum_{i\neq i'} U_{i,i'}\right|\left(d_j^M\right)^{-1} \to 0 \quad a.s. \tag{A.17}$$

where

$$U_{i,i'} = \boldsymbol{\delta}_\lambda(X_{i'}, X_i)w_i w_{i'} - \int \boldsymbol{\delta}(x, X_i)\sqrt{f(x)}\,\mathrm{d}x - \sqrt{f(X_{i'})} + 1. \tag{A.18}$$

Note $W_{i'} = \mathbb{E}[U_{i,i'}|X_{i'}]$ is related to the point-wise bias we studied in Subsection 3.3.1, $B(x) = \mathring{g}_j(x) - K_j\sqrt{f}(x)$,

$$W_{i'} = B(X_{i'})w_{i'} - \int B(x)\sqrt{f(x)}\,\mathrm{d}x. \tag{A.19}$$

Therefore, the expression there, namely (7.4),

$$\sum_{n=1}^{\infty} \#\left(\Lambda_n\right) \sup_{\lambda\in A_n} P\left[\left|n^{-1}\sum_{j=1}^{n} W_j\right| > \epsilon d_j^M\right] < \infty \tag{A.20}$$

can be derived in a similar fashiion in virtue of Proposition 3.3.1 and stated assumptions.

Equation (7.2) in Marron (1987) involves the $k$-th order cumulants $\mathrm{cum}_k$

$$\left|n^{-2k}\left(d_j^M\right)^{-k}\sum \mathrm{cum}_k\left(V_{i_1,i_1'}, \ldots, V_{i_k,i_k'}\right)\right| \leq C_k n^{-\gamma k}, \tag{A.21}$$

where $V_{i,i'} = U_{i,i'} - W_{i'}$ and the summation ranges over all distinct pairs. As we remarked above, the number of affected neighbours $i'$ such that $i \in NN_{i'}^{(k)}$ is bounded independently of $S$ and we can apply a similar argument to Marron (1987) to prove that

$$\sup_{\lambda\in\Lambda_n} n^{-2}\left|\sum_{i\neq i'} V_{i,i'}\right|\left(d_j^M\right)^{-1} \to 0 \quad a.s. \tag{A.22}$$

These two assert (A.17) and therefore (A.11), which is what we wanted to achieve.

□

## Proof of Theorem 9

Let $\Lambda_n$ be the set of possible thresholds $t$ of the estimator $\mathring{g}_{[t]}$ from a sample of size $n$. Thus, $\lambda$ represents the smoothness parameter for orthogonal series estimators. Recall that the threshold-dependent definition of the SPWDE estimator (4.16):

$$\mathring{g}_{[\tau]}(x) = \sum_{t=1}^{\tau} \hat{\beta}_{[t]}\, \psi_{[t]}(x).$$

can be writen as a delta sequence (4.17)

$$\mathring{g}_{[\tau]} = \sum_{i}^{n} \boldsymbol{\delta}_{[\tau]}\left(x, X_i\right) F_{i,n},$$

where $\boldsymbol{\delta}_{[\tau]}$ is defined by (4.18)

$$\boldsymbol{\delta}_{[\tau]}\left(x, X_i\right) = \sum_{t=1}^{\tau} \psi_{[t]}(x) \psi_{[t]}\left(X_i\right).$$

Thus, the assumptions we used to derive Theorem 8 in the context of $J$ need to be revised here and we look at them below.

As we mentioned in Section 4.3, the number of threshold choices come directly from the number of observations in the sample, hence $\#\left(\Lambda_n\right) \leq n^c$ holds for some $c \geq 1$. Also, if not for the factor $F_{i,n}$, the unnormalised $\mathring{g}_{[\tau]}$ estimator of $\sqrt{f}$ has the shape of delta estimators discussed in (Marron, 1987), under the scope of orthogonal series estimator. The only remaining assumption we add is that there are two constants $C$ and $\epsilon > 0$ such that

$$C^{-1} n^{\epsilon} \leq \lambda \leq C n^{1-\epsilon}. \tag{A.23}$$

Apart from these observations, the proof follows the same argument as for Theorem 8. □

# Appendix B

# Supplementary results

## B.1 Example distributions

- Mixtures for simulation study in Subsection 3.4.1

$$
\begin{aligned}
f_{(a)} &= 0.16667\mathcal{N}\left((0.20000, 0.30000), \begin{pmatrix} 0.00167 & 0.00000 \\ 0.00000 & 0.00167 \end{pmatrix}\right) \\
&+ 0.83333\mathcal{N}\left((0.70000, 0.70000), \begin{pmatrix} 0.01500 & 0.00016 \\ 0.00016 & 0.01500 \end{pmatrix}\right)
\end{aligned}
$$

$$
\begin{aligned}
f_{(b)} &= 0.50000\mathcal{N}\left((0.40000, 0.30000), \begin{pmatrix} 0.00833 & 0.00000 \\ 0.00000 & 0.00625 \end{pmatrix}\right) \\
&+ 0.50000\mathcal{N}\left((0.70000, 0.70000), \begin{pmatrix} 0.00827 & 0.00036 \\ 0.00036 & 0.00631 \end{pmatrix}\right)
\end{aligned}
$$

$$
\begin{aligned}
f_{(c)} &= 0.36364\mathcal{N}\left((0.20000, 0.30000), \begin{pmatrix} 0.00500 & 0.00000 \\ 0.00000 & 0.00375 \end{pmatrix}\right) \\
&+ 0.27273\mathcal{N}\left((0.50000, 0.50000), \begin{pmatrix} 0.00331 & 0.00014 \\ 0.00014 & 0.00253 \end{pmatrix}\right) \\
&+ 0.22727\mathcal{N}\left((0.65000, 0.70000), \begin{pmatrix} 0.00250 & 0.00000 \\ 0.00000 & 0.00187 \end{pmatrix}\right) \\
&+ 0.13636\mathcal{N}\left((0.82000, 0.85000), \begin{pmatrix} 0.00165 & 0.00007 \\ 0.00007 & 0.00126 \end{pmatrix}\right)
\end{aligned}
$$

$$
\begin{aligned}
f_{(d)} = 0.40000\mathcal{N}&\left((0.30000, 0.40000, 0.35000), \begin{pmatrix} 0.02000 & 0.01000 & 0.00000 \\ 0.01000 & 0.02000 & 0.00000 \\ 0.00000 & 0.00000 & 0.02000 \end{pmatrix}\right) \\
+ 0.30000\mathcal{N}&\left((0.70000, 0.70000, 0.60000), \begin{pmatrix} 0.01333 & 0.00000 & 0.00000 \\ 0.00000 & 0.01333 & 0.00000 \\ 0.00000 & 0.00000 & 0.01333 \end{pmatrix}\right) \\
+ 0.30000\mathcal{N}&\left((0.70000, 0.60000, 0.35000), \begin{pmatrix} 0.02500 & 0.00000 & 0.00000 \\ 0.00000 & 0.02500 & 0.01000 \\ 0.00000 & 0.01000 & 0.02500 \end{pmatrix}\right)
\end{aligned}
$$

- Mixtures for simulation study in Subsection 4.4.1

$$
\begin{aligned}
f_{(a)} = 0.36364\mathcal{N}&\left((0.20000, 0.30000), \begin{pmatrix} 0.00500 & 0.00000 \\ 0.00000 & 0.00375 \end{pmatrix}\right) \\
+ 0.27273\mathcal{N}&\left((0.50000, 0.50000), \begin{pmatrix} 0.00331 & 0.00014 \\ 0.00014 & 0.00253 \end{pmatrix}\right) \\
+ 0.22727\mathcal{N}&\left((0.65000, 0.70000), \begin{pmatrix} 0.00250 & 0.00000 \\ 0.00000 & 0.00187 \end{pmatrix}\right) \\
+ 0.13636\mathcal{N}&\left((0.82000, 0.85000), \begin{pmatrix} 0.00165 & 0.00007 \\ 0.00007 & 0.00126 \end{pmatrix}\right)
\end{aligned}
$$

$$
\begin{aligned}
f_{(b)}(x, y) = 19.2 &\left(\begin{cases} 8(x - 0.125) & x \geq 0.125, x \leq 0.25 \\ 1 - 2(x - 0.25) & x \geq 0.25, x \leq 0.75 \\ 0 & \textit{otherwise} \end{cases}\right) \times \\
&\left(\begin{cases} 16(y - 0.625) & y \geq 0.625, y \leq 0.6875 \\ 1 - 16(y - 0.6875) & y \geq 0.6875, y \leq 0.75 \\ 0 & \textit{otherwise} \end{cases}\right) + \\
16 &\left(\begin{cases} 16(x - 0.5) & x \geq 0.5, x \leq 0.5625 \\ 1 - 16(x - 0.5625) & x \geq 0.5625, x \leq 0.625 \\ 0 & \textit{otherwise} \end{cases}\right) \times \\
&\left(\begin{cases} 8(y - 0.125) & y \geq 0.125, y \leq 0.25 \\ 1 - 1.6(y - 0.25) & y \geq 0.25, y \leq 0.875 \\ 0 & \textit{otherwise} \end{cases}\right)
\end{aligned}
$$

$$\begin{aligned}
f_{(c)} &= 0.94737\mathcal{N}\left((0.50000, 0.50000), \begin{pmatrix} 0.05000 & 0.00000 \\ 0.00000 & 0.05000 \end{pmatrix}\right) \\
&+ 0.03158\mathcal{N}\left((0.65000, 0.70000), \begin{pmatrix} 0.00033 & 0.00001 \\ 0.00001 & 0.00025 \end{pmatrix}\right) \\
&+ 0.02105\mathcal{N}\left((0.69000, 0.75000), \begin{pmatrix} 0.00033 & 0.00000 \\ 0.00000 & 0.00025 \end{pmatrix}\right)
\end{aligned}$$

$$\begin{aligned}
f_{(d)} &= 0.50000\mathcal{N}\left((0.30000, 0.50000), \begin{pmatrix} 0.01562 & 0.01437 \\ 0.01437 & 0.01562 \end{pmatrix}\right) \\
&+ 0.50000\mathcal{N}\left((0.70000, 0.50000), \begin{pmatrix} 0.01562 & -0.01437 \\ -0.01437 & 0.01562 \end{pmatrix}\right)
\end{aligned}$$

- Mixtures for simulation study in Subsection 4.4.2

$$\begin{aligned}
f_{(a)} &= 0.74074\mathcal{N}\left((0.45000, 0.45000), \begin{pmatrix} 0.00992 & 0.00043 \\ 0.00043 & 0.00758 \end{pmatrix}\right) \\
&+ 0.18519\mathcal{N}\left((0.70000, 0.70000), \begin{pmatrix} 0.00050 & 0.00000 \\ 0.00000 & 0.00038 \end{pmatrix}\right) \\
&+ 0.07407\mathcal{N}\left((0.45000, 0.45000), \begin{pmatrix} 0.00033 & 0.00000 \\ 0.00000 & 0.00025 \end{pmatrix}\right)
\end{aligned}$$

$$\begin{aligned}
f_{(b)} &= 0.50000\mathcal{N}\left((0.45000, 0.45000), \begin{pmatrix} 0.00496 & 0.00021 \\ 0.00021 & 0.00379 \end{pmatrix}\right) \\
&+ 0.50000\mathcal{N}\left((0.70000, 0.70000), \begin{pmatrix} 0.00062 & 0.00000 \\ 0.00000 & 0.00047 \end{pmatrix}\right)
\end{aligned}$$

$$
\begin{aligned}
f_{(c)} = {} & 0.50000\mathcal{N}\left((0.66000, 0.33000), \begin{pmatrix} 0.00992 & 0.00043 \\ 0.00043 & 0.00758 \end{pmatrix}\right) \\
& + 0.10000\mathcal{N}\left((0.25000, 0.25000), \begin{pmatrix} 0.00062 & 0.00000 \\ 0.00000 & 0.00047 \end{pmatrix}\right) \\
& + 0.10000\mathcal{N}\left((0.25000, 0.50000), \begin{pmatrix} 0.00062 & 0.00000 \\ 0.00000 & 0.00047 \end{pmatrix}\right) \\
& + 0.10000\mathcal{N}\left((0.25000, 0.75000), \begin{pmatrix} 0.00062 & 0.00000 \\ 0.00000 & 0.00047 \end{pmatrix}\right) \\
& + 0.10000\mathcal{N}\left((0.50000, 0.75000), \begin{pmatrix} 0.00062 & 0.00000 \\ 0.00000 & 0.00047 \end{pmatrix}\right) \\
& + 0.10000\mathcal{N}\left((0.75000, 0.75000), \begin{pmatrix} 0.00062 & 0.00000 \\ 0.00000 & 0.00047 \end{pmatrix}\right)
\end{aligned}
$$

$$
\begin{aligned}
f_{(d)} = {} & 0.27273\mathcal{N}\left((0.25000, 0.25000), \begin{pmatrix} 0.00625 & 0.00000 \\ 0.00000 & 0.00469 \end{pmatrix}\right) \\
& + 0.23636\mathcal{N}\left((0.25000, 0.50000), \begin{pmatrix} 0.00312 & 0.00000 \\ 0.00000 & 0.00234 \end{pmatrix}\right) \\
& + 0.20000\mathcal{N}\left((0.25000, 0.75000), \begin{pmatrix} 0.00208 & 0.00000 \\ 0.00000 & 0.00156 \end{pmatrix}\right) \\
& + 0.16364\mathcal{N}\left((0.50000, 0.75000), \begin{pmatrix} 0.00156 & 0.00000 \\ 0.00000 & 0.00117 \end{pmatrix}\right) \\
& + 0.12727\mathcal{N}\left((0.75000, 0.75000), \begin{pmatrix} 0.00125 & 0.00000 \\ 0.00000 & 0.00094 \end{pmatrix}\right)
\end{aligned}
$$

## B.2 Thresholding - Number of coefficients

Number of wavelet coefficients for the various non linear estimator algorithms of Section 4.3 using different wavelets for the densities in Figure 4.7. In each table and for each algorithm, *Med* is the median number of coefficients required across all samples, whereas *Q1* and *Q3* represent the lower and upper quantile ranges. Values in bold highlight the same entries as in corresponding tables in the main text.

| **n** | ${}_p\widehat{\mathcal{B}}_J$ | $\Delta J$ | $\lambda$ | | | $\lambda\sqrt{\Delta J}$ | | | $\lambda\,\sigma^B$ | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | $Q_1$ | $Med$ | $Q_3$ | $Q_1$ | $Med$ | $Q_3$ | $Q_1$ | $Med$ | $Q_3$ |
| 250 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 101.0 | 108.0 | 118.2 | 101.0 | 108.0 | 118.2 | 151.5 | 167.0 | 181.2 |
| | | 2 | 101.8 | 111.5 | 119.0 | 100.0 | 109.0 | 124.0 | 222.0 | 241.5 | 265.0 |
| | | 3 | 98.5 | 107.5 | 122.0 | 102.0 | 115.0 | 136.2 | 301.5 | 331.5 | 362.2 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 101.0 | 107.0 | 112.2 | 101.0 | 107.0 | 112.2 | 148.0 | 160.0 | 175.0 |
| | | 2 | 102.0 | 110.0 | 120.0 | 100.0 | 106.5 | 124.0 | 223.5 | 240.5 | 266.0 |
| | | 3 | 98.5 | 107.5 | 123.0 | 101.0 | 111.0 | 127.2 | 301.5 | 330.5 | 365.0 |
| 500 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 118.0 | 136.0 | 146.0 | 118.0 | 136.0 | 146.0 | 166.8 | 194.5 | 211.0 |
| | | 2 | 117.0 | 127.0 | 135.0 | 119.0 | 128.0 | 140.2 | 229.5 | 244.5 | 260.5 |
| | | 3 | 113.0 | 121.5 | 132.2 | 122.0 | 129.0 | 139.0 | 309.0 | 329.5 | 350.8 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 113.0 | 122.0 | 137.2 | 113.0 | 122.0 | 137.2 | 160.8 | 173.5 | 197.0 |
| | | 2 | 114.8 | 124.5 | 134.0 | 119.0 | 126.5 | 136.0 | 226.0 | 238.5 | 260.0 |
| | | 3 | 112.0 | 123.0 | 137.2 | 118.8 | 128.0 | 142.2 | 309.0 | 326.0 | 350.8 |
| 1000 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 147.0 | 153.0 | 159.0 | 147.0 | 153.0 | 159.0 | 203.0 | 213.0 | 225.0 |
| | | 2 | 132.0 | 137.0 | 142.0 | 137.0 | 142.0 | 149.0 | 236.0 | 255.0 | 270.0 |
| | | 3 | 125.0 | 134.0 | 144.0 | 127.0 | 134.0 | 138.0 | 323.0 | 345.0 | 367.0 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 148.0 | 152.0 | 159.0 | 148.0 | 152.0 | 159.0 | 201.0 | 212.0 | 223.0 |
| | | 2 | 131.0 | 136.0 | 140.0 | 134.0 | 141.0 | 148.0 | 236.0 | 251.0 | 267.0 |
| | | 3 | 125.0 | 133.0 | 142.0 | 127.0 | 134.0 | 138.0 | 319.0 | 342.0 | 362.0 |
| 1500 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 155.0 | 158.0 | 164.0 | 155.0 | 158.0 | 164.0 | 202.0 | **217.0** | 231.2 |
| | | 2 | 137.0 | 143.0 | 150.8 | 140.0 | 146.5 | 151.0 | 247.2 | 266.5 | 281.8 |
| | | 3 | 138.2 | 145.5 | 152.0 | 130.0 | 135.0 | 151.2 | 327.0 | 361.0 | 385.0 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 155.0 | 158.0 | 164.0 | 155.0 | 158.0 | 164.0 | 202.0 | **213.0** | 225.0 |
| | | 2 | 137.0 | 141.0 | 149.8 | 140.0 | 146.5 | 151.0 | 242.5 | 265.5 | 280.8 |
| | | 3 | 137.0 | 144.0 | 151.0 | 129.0 | 134.0 | 148.0 | 325.2 | 358.5 | 380.8 |
| 2000 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 161.0 | 166.0 | 171.0 | 161.0 | 166.0 | 171.0 | 214.0 | **226.0** | 239.0 |
| | | 2 | 142.0 | 148.0 | 159.0 | 142.0 | 148.0 | 155.0 | 250.0 | **267.0** | 290.5 |
| | | 3 | 142.0 | 151.0 | 163.0 | 143.8 | 163.5 | 175.2 | 337.8 | **355.0** | 385.0 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 161.0 | 165.0 | 170.2 | 161.0 | 165.0 | 170.2 | 214.0 | **226.0** | 238.2 |
| | | 2 | 141.0 | 147.5 | 157.0 | 142.0 | 147.5 | 154.0 | 249.0 | **267.0** | 290.5 |
| | | 3 | 142.0 | 150.0 | 161.2 | 140.8 | 162.0 | 173.2 | 337.8 | **354.5** | 385.0 |
| 3000 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 170.0 | 174.0 | 180.0 | 170.0 | 174.0 | 180.0 | 222.8 | **234.0** | 247.0 |
| | | 2 | 157.0 | 164.0 | 170.2 | 151.8 | 160.0 | 168.0 | 266.0 | **286.0** | 307.0 |
| | | 3 | 160.0 | 168.0 | 175.0 | 166.8 | 173.5 | 183.0 | 360.0 | **383.0** | 405.5 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 169.0 | 173.0 | 179.0 | 169.0 | 173.0 | 179.0 | 222.8 | **233.5** | 247.0 |
| | | 2 | 157.0 | 164.0 | 170.0 | 150.5 | 159.0 | 166.2 | 265.8 | **283.0** | 305.2 |
| | | 3 | 159.8 | 167.0 | 175.0 | 165.5 | 173.0 | 182.0 | 359.0 | **381.5** | 405.5 |
| 4000 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 172.0 | **178.0** | 183.0 | 172.0 | **178.0** | 183.0 | 234.0 | **242.0** | 257.0 |
| | | 2 | 162.0 | 167.0 | 172.0 | 162.0 | 170.0 | 180.0 | 283.0 | **299.0** | 320.0 |
| | | 3 | 164.0 | 172.0 | 177.0 | 170.0 | 175.0 | 184.0 | 376.0 | **397.0** | 419.0 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 174.0 | **179.0** | 183.0 | 174.0 | **179.0** | 183.0 | 223.0 | **238.0** | 253.0 |
| | | 2 | 161.0 | 165.0 | 170.0 | 158.0 | 172.0 | 179.0 | 274.0 | **288.0** | 310.0 |
| | | 3 | 163.0 | 171.0 | 178.0 | 169.0 | 174.0 | 184.0 | 370.0 | **386.0** | 411.0 |
| 6000 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 185.0 | 189.0 | 193.0 | 185.0 | 189.0 | 193.0 | 225.0 | **241.0** | 257.0 |
| | | 2 | 170.0 | 174.0 | 178.0 | 173.0 | 180.0 | 182.0 | 271.0 | **295.0** | 318.0 |
| | | 3 | 173.0 | 176.0 | 184.0 | 178.0 | 186.0 | 190.0 | 363.0 | **395.0** | 422.0 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 183.0 | 186.0 | 192.2 | 183.0 | 186.0 | 192.2 | 228.0 | **244.5** | 258.2 |
| | | 2 | 171.0 | 175.0 | 179.0 | 173.0 | 176.0 | 181.0 | 272.8 | **299.0** | 321.5 |
| | | 3 | 172.8 | 179.0 | 183.0 | 179.0 | 185.0 | 192.0 | 369.5 | **397.0** | 427.5 |

Table B.1. Number of wavelet coefficients using the Daubechies 4 wavelet for the density Kurtotic Mixture 1 (Figure 4.7 (a)). Corresponding table for Hellinger distance is 4.1

| **n** | ${}_p\widehat{\mathcal{B}}_J$ | $\Delta J$ | $\lambda$ | | | $\lambda\sqrt{\Delta J}$ | | | $\lambda\,\sigma^B$ | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | $Q_1$ | $Med$ | $Q_3$ | $Q_1$ | $Med$ | $Q_3$ | $Q_1$ | $Med$ | $Q_3$ |
| 250 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 120.5 | 126.0 | 132.0 | 120.5 | 126.0 | 132.0 | 175.0 | **187.0** | 200.0 |
| | | 2 | 112.0 | 117.0 | 130.5 | 114.5 | 121.0 | 128.5 | 225.0 | **241.0** | 265.0 |
| | | 3 | 115.0 | 120.0 | 132.0 | 118.0 | 126.0 | 139.5 | 313.0 | **335.0** | 362.0 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 120.0 | 124.0 | 130.0 | 120.0 | 124.0 | 130.0 | 169.5 | **183.0** | 198.5 |
| | | 2 | 109.0 | 115.0 | 124.0 | 113.0 | 118.0 | 127.5 | 218.5 | **238.0** | 264.5 |
| | | 3 | 111.5 | 119.0 | 128.0 | 115.5 | 122.0 | 136.5 | 302.0 | **330.0** | 362.0 |
| 500 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 129.0 | 135.0 | 141.2 | 129.0 | 135.0 | 141.2 | 188.0 | **199.5** | 211.5 |
| | | 2 | 125.0 | 134.5 | 147.0 | 124.8 | 134.5 | 148.0 | 245.0 | **270.0** | 292.2 |
| | | 3 | 127.0 | 137.0 | 148.2 | 137.0 | 146.5 | 152.2 | 339.5 | **360.5** | 391.2 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 128.0 | 134.5 | 141.0 | 128.0 | 134.5 | 141.0 | 186.2 | **198.0** | 211.0 |
| | | 2 | 123.0 | 132.0 | 143.0 | 123.0 | 132.5 | 147.2 | 242.5 | **266.5** | 289.0 |
| | | 3 | 125.0 | 135.0 | 146.0 | 131.0 | 143.5 | 149.2 | 336.2 | **357.5** | 387.2 |
| 1000 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 142.8 | **146.0** | 152.0 | 142.8 | **146.0** | 152.0 | 204.8 | **219.5** | 228.8 |
| | | 2 | 147.0 | 151.5 | 156.0 | 148.0 | 153.5 | 160.0 | 278.5 | **299.5** | 324.8 |
| | | 3 | 144.0 | 149.0 | 156.2 | 144.8 | 150.5 | 157.0 | 372.8 | **399.0** | 432.5 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 142.0 | **146.0** | 152.0 | 142.0 | **146.0** | 152.0 | 203.0 | **216.0** | 226.2 |
| | | 2 | 147.0 | 151.0 | 155.2 | 147.0 | 152.0 | 159.0 | 273.8 | **297.5** | 319.5 |
| | | 3 | 144.0 | 149.0 | 156.0 | 144.0 | 150.0 | 156.2 | 371.8 | **397.5** | 431.2 |
| 1500 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 149.0 | **153.0** | 158.0 | 149.0 | **153.0** | 158.0 | 209.8 | **220.0** | 228.2 |
| | | 2 | 149.8 | 154.0 | 158.2 | 152.0 | 158.0 | 163.0 | 283.0 | **301.5** | 324.2 |
| | | 3 | 150.0 | 158.0 | 170.5 | 148.0 | 155.5 | 182.0 | 382.5 | **402.0** | 428.5 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 148.0 | **153.0** | 158.0 | 148.0 | **153.0** | 158.0 | 209.0 | **219.5** | 228.0 |
| | | 2 | 149.8 | 154.0 | 158.2 | 152.0 | 157.0 | 163.0 | 283.0 | **301.5** | 324.2 |
| | | 3 | 148.8 | 157.0 | 170.0 | 147.8 | 155.0 | 179.5 | 382.5 | **402.5** | 426.2 |
| 2000 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 153.0 | **157.0** | 160.0 | 153.0 | **157.0** | 160.0 | 219.0 | **228.0** | 236.2 |
| | | 2 | 153.0 | 158.0 | 163.2 | 156.0 | 162.5 | 169.2 | 301.0 | **318.0** | 339.2 |
| | | 3 | 164.8 | 176.0 | 191.2 | 155.0 | 182.5 | 196.2 | 399.8 | **416.0** | 442.0 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 153.0 | **157.0** | 160.0 | 153.0 | **157.0** | 160.0 | 218.8 | **227.5** | 236.0 |
| | | 2 | 153.0 | 158.0 | 162.2 | 156.0 | 162.0 | 169.0 | 301.0 | **318.0** | 334.2 |
| | | 3 | 163.8 | 175.0 | 191.0 | 154.0 | 180.0 | 194.0 | 399.0 | **416.0** | 438.5 |
| 3000 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 155.0 | **159.0** | 163.2 | 155.0 | **159.0** | 163.2 | 226.0 | **235.5** | 250.2 |
| | | 2 | 159.0 | 167.0 | 176.0 | 163.0 | 169.0 | 179.0 | 311.8 | **332.5** | 352.5 |
| | | 3 | 176.8 | 190.0 | 203.2 | 187.0 | 202.0 | 220.0 | 412.8 | **433.0** | 456.0 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 155.0 | **159.0** | 163.0 | 155.0 | **159.0** | 163.0 | 225.8 | **235.5** | 250.2 |
| | | 2 | 159.0 | 167.0 | 176.0 | 162.0 | 169.0 | 178.2 | 309.0 | **329.5** | 352.5 |
| | | 3 | 176.8 | 189.5 | 202.2 | 187.0 | 202.0 | 220.0 | 409.8 | **433.0** | 454.5 |
| 4000 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 161.0 | **166.0** | 170.0 | 161.0 | **166.0** | 170.0 | 231.8 | **241.0** | 253.0 |
| | | 2 | 173.0 | 181.0 | 187.0 | 169.0 | 180.0 | 191.0 | 328.0 | **344.5** | 361.2 |
| | | 3 | 190.0 | **198.0** | 207.0 | 195.0 | 211.5 | 225.2 | 429.5 | **445.5** | 466.0 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 161.0 | **165.0** | 169.0 | 161.0 | **165.0** | 169.0 | 231.0 | **241.0** | 252.2 |
| | | 2 | 172.8 | 179.5 | 187.0 | 169.0 | 179.5 | 189.2 | 326.8 | **342.5** | 357.8 |
| | | 3 | 190.0 | 198.0 | 206.0 | 193.8 | 209.0 | 225.2 | 425.0 | **444.5** | 463.0 |
| 6000 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 169.0 | 174.0 | 178.0 | 169.0 | 174.0 | 178.0 | 236.0 | 248.0 | 257.2 |
| | | 2 | 181.0 | 189.0 | 197.0 | 184.0 | 190.0 | 197.0 | 333.0 | 351.0 | 368.2 |
| | | 3 | 196.0 | 204.5 | 212.0 | 213.8 | 220.0 | 228.0 | 433.5 | 455.5 | 475.0 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 169.0 | 174.0 | 178.0 | 169.0 | 174.0 | 178.0 | 236.0 | 248.0 | 257.2 |
| | | 2 | 181.0 | 188.5 | 196.0 | 184.0 | 190.0 | 197.0 | 333.0 | 351.0 | 368.0 |
| | | 3 | 196.0 | 204.0 | 212.0 | 212.8 | 220.0 | 227.2 | 433.5 | 454.5 | 475.0 |

Table B.2. Number of wavelet coefficients using the Daubechies 4 wavelet for the density Mixture 2 (Figure 4.7 (b)). Corresponding table for Hellinger distance is 4.2

| **n** | $_p\widehat{\mathcal{B}}_J$ | $\Delta J$ | $\lambda$ | | | $\lambda\sqrt{\Delta J}$ | | | $\lambda\,\sigma^B$ | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | $Q_1$ | $Med$ | $Q_3$ | $Q_1$ | $Med$ | $Q_3$ | $Q_1$ | $Med$ | $Q_3$ |
| 250 | $_1\widehat{\mathcal{B}}_J$ | 1 | 95.0 | 98.0 | 102.0 | 95.0 | 98.0 | 102.0 | 168.0 | 178.0 | 187.0 |
| | | 2 | 78.5 | 84.0 | 92.0 | 77.5 | 84.0 | 92.5 | 207.0 | 217.0 | 229.0 |
| | | 3 | 84.5 | 99.0 | 108.0 | 91.5 | 108.0 | 128.5 | 258.0 | 267.0 | 283.0 |
| | $_2\widehat{\mathcal{B}}_J$ | 1 | 94.5 | 98.0 | 101.5 | 94.5 | 98.0 | 101.5 | 167.5 | 176.0 | 184.5 |
| | | 2 | 77.0 | 82.0 | 89.0 | 76.5 | 83.0 | 90.0 | 203.5 | 215.0 | 226.0 |
| | | 3 | 84.0 | 97.0 | 106.0 | 90.5 | 105.0 | 127.5 | 253.0 | 265.0 | 280.0 |
| 500 | $_1\widehat{\mathcal{B}}_J$ | 1 | 99.0 | 103.0 | 118.5 | 99.0 | 103.0 | 118.5 | 178.0 | 186.0 | 197.2 |
| | | 2 | 90.0 | 100.5 | 114.0 | 97.8 | 106.5 | 119.0 | 227.8 | 240.5 | 255.0 |
| | | 3 | 106.8 | 120.0 | 135.0 | 116.0 | 125.5 | 138.0 | 279.0 | 291.0 | 308.0 |
| | $_2\widehat{\mathcal{B}}_J$ | 1 | 99.0 | 102.0 | 115.0 | 99.0 | 102.0 | 115.0 | 175.0 | 185.0 | 195.2 |
| | | 2 | 88.0 | 97.0 | 112.0 | 92.8 | 105.0 | 114.0 | 225.0 | 237.5 | 250.0 |
| | | 3 | 106.8 | 120.0 | 137.2 | 116.0 | 126.5 | 138.5 | 277.8 | 290.5 | 304.2 |
| 1000 | $_1\widehat{\mathcal{B}}_J$ | 1 | 131.0 | 136.5 | 141.0 | 131.0 | 136.5 | 141.0 | 196.0 | **208.5** | 223.0 |
| | | 2 | 121.0 | 126.5 | 132.0 | 122.0 | 127.5 | 137.0 | 263.8 | 283.0 | 297.0 |
| | | 3 | 109.8 | 116.0 | 132.0 | 112.0 | 120.0 | 132.0 | 317.5 | 342.0 | 355.2 |
| | $_2\widehat{\mathcal{B}}_J$ | 1 | 123.0 | 135.0 | 139.0 | 123.0 | 135.0 | 139.0 | 191.8 | 204.5 | 222.0 |
| | | 2 | 119.5 | 126.0 | 132.0 | 121.0 | 127.5 | 136.2 | 256.0 | 277.5 | 296.0 |
| | | 3 | 110.8 | 118.5 | 137.0 | 113.8 | 124.0 | 139.0 | 303.8 | 334.0 | 354.2 |
| 1500 | $_1\widehat{\mathcal{B}}_J$ | 1 | 136.0 | **139.0** | 143.2 | 136.0 | **139.0** | 143.2 | 205.5 | **217.0** | 235.0 |
| | | 2 | 124.0 | 130.0 | 139.0 | 127.8 | 134.0 | 139.0 | 273.5 | **291.5** | 314.0 |
| | | 3 | 117.0 | 125.0 | 134.0 | 114.0 | 125.0 | 140.0 | 326.2 | **355.0** | 377.2 |
| | $_2\widehat{\mathcal{B}}_J$ | 1 | 135.0 | **138.5** | 142.2 | 135.0 | **138.5** | 142.2 | 200.8 | **214.0** | 232.2 |
| | | 2 | 123.0 | 129.0 | 138.0 | 127.0 | 133.0 | 139.0 | 268.0 | **289.0** | 308.2 |
| | | 3 | 115.0 | 125.0 | 134.2 | 114.0 | 124.0 | 139.0 | 324.0 | **354.0** | 369.8 |
| 2000 | $_1\widehat{\mathcal{B}}_J$ | 1 | 137.0 | **141.5** | 146.0 | 137.0 | **141.5** | 146.0 | 209.0 | **222.5** | 234.0 |
| | | 2 | 129.0 | 133.0 | 139.0 | 130.0 | 136.0 | 144.0 | 278.5 | **296.0** | 314.0 |
| | | 3 | 124.8 | 133.0 | 141.0 | 123.8 | 137.0 | 150.2 | 339.0 | **357.0** | 373.0 |
| | $_2\widehat{\mathcal{B}}_J$ | 1 | 137.0 | **141.0** | 146.0 | 137.0 | **141.0** | 146.0 | 207.8 | **219.5** | 230.5 |
| | | 2 | 129.0 | 132.0 | 137.0 | 130.0 | 135.5 | 142.0 | 277.0 | **294.5** | 312.0 |
| | | 3 | 123.0 | 131.5 | 140.0 | 121.0 | 132.0 | 147.2 | 336.8 | **356.0** | 371.5 |
| 3000 | $_1\widehat{\mathcal{B}}_J$ | 1 | 141.0 | **145.5** | 151.0 | 141.0 | **145.5** | 151.0 | 220.0 | **230.0** | 243.0 |
| | | 2 | 135.8 | **143.0** | 152.0 | 137.8 | 143.0 | 152.0 | 293.0 | **308.5** | 326.2 |
| | | 3 | 133.8 | 144.0 | 156.0 | 150.0 | 161.5 | 169.2 | 351.5 | **372.0** | 389.0 |
| | $_2\widehat{\mathcal{B}}_J$ | 1 | 140.8 | **145.0** | 149.2 | 140.8 | **145.0** | 149.2 | 218.8 | **228.0** | 241.2 |
| | | 2 | 134.0 | **141.0** | 149.2 | 136.0 | 140.0 | 151.0 | 291.8 | **306.0** | 324.2 |
| | | 3 | 133.0 | 142.5 | 155.2 | 150.0 | 160.5 | 168.0 | 351.5 | **369.5** | 389.0 |
| 4000 | $_1\widehat{\mathcal{B}}_J$ | 1 | 144.0 | **148.0** | 154.0 | 144.0 | **148.0** | 154.0 | 225.0 | **239.0** | 253.0 |
| | | 2 | 145.8 | 155.0 | 164.0 | 141.8 | 153.0 | 166.0 | 297.0 | **316.5** | 338.2 |
| | | 3 | 145.0 | 155.5 | 163.5 | 160.0 | 169.0 | 177.2 | 362.0 | **381.5** | 403.0 |
| | $_2\widehat{\mathcal{B}}_J$ | 1 | 144.0 | **147.5** | 154.0 | 144.0 | **147.5** | 154.0 | 222.0 | **238.5** | 253.0 |
| | | 2 | 143.5 | 154.0 | 163.0 | 141.8 | 153.0 | 164.2 | 297.0 | **316.0** | 338.0 |
| | | 3 | 143.8 | 155.0 | 163.0 | 158.0 | 168.0 | 177.0 | 362.0 | **381.5** | 403.0 |
| 6000 | $_1\widehat{\mathcal{B}}_J$ | 1 | 152.8 | 159.0 | 165.0 | 152.8 | 159.0 | 165.0 | 235.8 | **247.0** | 257.2 |
| | | 2 | 158.8 | 165.5 | 171.0 | 158.0 | 167.0 | 181.0 | 312.8 | **328.0** | 343.8 |
| | | 3 | 159.8 | 165.0 | 175.0 | 167.8 | 177.0 | 184.0 | 375.8 | **395.5** | 411.2 |
| | $_2\widehat{\mathcal{B}}_J$ | 1 | 152.8 | 159.0 | 163.2 | 152.8 | 159.0 | 163.2 | 235.8 | **247.0** | 254.8 |
| | | 2 | 160.8 | 165.0 | 171.0 | 159.5 | 170.5 | 181.2 | 315.2 | **327.0** | 340.8 |
| | | 3 | 160.0 | 166.5 | 175.2 | 163.8 | 174.0 | 182.5 | 379.0 | **395.5** | 412.0 |

Table B.3. Number of wavelet coefficients using the Symlet 3 wavelet for the density 2D Comb 1 (claw) (Figure 4.7 (c)). Corresponding table for Hellinger distance is 4.3

| n | ${}_p\widehat{\mathcal{B}}_J$ | $\Delta J$ | $\lambda$ | | | $\lambda\sqrt{\Delta J}$ | | | $\lambda\,\sigma^B$ | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | $Q_1$ | $Med$ | $Q_3$ | $Q_1$ | $Med$ | $Q_3$ | $Q_1$ | $Med$ | $Q_3$ |
| 250 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 76.0 | **78.0** | 80.0 | 76.0 | **78.0** | 80.0 | 132.0 | **142.0** | 150.0 |
| | | 2 | 64.0 | **68.0** | 74.0 | 65.0 | **68.0** | 76.0 | 164.0 | **177.0** | 193.0 |
| | | 3 | 70.0 | **76.0** | 97.0 | 76.0 | **81.0** | 105.0 | 209.0 | **224.0** | 244.0 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 76.0 | **78.0** | 80.0 | 76.0 | **78.0** | 80.0 | 129.0 | **138.0** | 147.0 |
| | | 2 | 64.0 | **68.0** | 73.0 | 64.0 | **68.0** | 72.5 | 161.8 | **174.0** | 191.5 |
| | | 3 | 69.0 | **74.0** | 91.5 | 73.0 | **80.0** | 101.8 | 206.0 | **223.0** | 243.0 |
| 500 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 78.0 | **81.0** | 85.0 | 78.0 | **81.0** | 85.0 | 134.8 | **144.5** | 154.0 |
| | | 2 | 69.0 | **75.0** | 88.0 | 70.8 | **80.0** | 90.2 | 169.0 | **183.0** | 196.2 |
| | | 3 | 90.0 | **102.0** | 113.5 | 97.0 | **117.5** | 138.2 | 216.2 | **231.5** | 248.0 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 77.8 | **81.0** | 85.0 | 77.8 | **81.0** | 85.0 | 134.0 | **144.5** | 153.2 |
| | | 2 | 69.0 | **74.5** | 87.0 | 69.0 | **77.5** | 88.2 | 168.0 | **182.0** | 195.0 |
| | | 3 | 86.0 | **102.0** | 111.0 | 94.0 | **113.0** | 137.2 | 213.0 | **230.5** | 246.2 |
| 1000 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 87.0 | **90.0** | 95.0 | 87.0 | **90.0** | 95.0 | 141.0 | **150.0** | 156.0 |
| | | 2 | 87.0 | **93.5** | 105.2 | 87.0 | **96.5** | 112.2 | 179.0 | **191.5** | 201.2 |
| | | 3 | 104.0 | **116.5** | 136.0 | 123.0 | **134.5** | 151.2 | 227.0 | **241.0** | 253.5 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 87.0 | **90.0** | 94.2 | 87.0 | **90.0** | 94.2 | 140.8 | **150.0** | 155.2 |
| | | 2 | 87.0 | **93.0** | 105.0 | 87.0 | **96.5** | 112.0 | 179.0 | **191.0** | 201.0 |
| | | 3 | 104.0 | **116.5** | 136.0 | 123.0 | **133.5** | 151.0 | 227.0 | **240.5** | 253.0 |
| 1500 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 91.0 | 93.0 | 98.2 | 91.0 | 93.0 | 98.2 | 147.0 | 152.5 | 161.2 |
| | | 2 | 92.0 | 106.5 | 114.0 | 97.0 | 109.5 | 120.0 | 184.8 | 197.0 | 209.0 |
| | | 3 | 123.0 | 135.5 | 144.2 | 125.8 | 139.0 | 154.2 | 232.0 | 245.5 | 261.5 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 90.8 | 93.0 | 98.0 | 90.8 | 93.0 | 98.0 | 147.0 | 152.0 | 160.2 |
| | | 2 | 92.0 | 106.5 | 114.0 | 97.0 | 109.0 | 120.0 | 184.8 | 196.5 | 208.2 |
| | | 3 | 122.8 | 135.0 | 144.0 | 125.8 | 137.0 | 152.5 | 232.0 | 245.5 | 261.5 |
| 2000 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 93.0 | 96.0 | 100.0 | 93.0 | 96.0 | 100.0 | 151.0 | 156.0 | 162.0 |
| | | 2 | 102.8 | 110.5 | 117.2 | 102.8 | 114.0 | 123.2 | 191.0 | 200.0 | 214.2 |
| | | 3 | 128.0 | 138.0 | 147.0 | 134.0 | 145.0 | 161.2 | 240.8 | 250.0 | 264.2 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 93.0 | 96.0 | 100.0 | 93.0 | 96.0 | 100.0 | 150.5 | 156.0 | 161.2 |
| | | 2 | 102.8 | 110.5 | 117.2 | 102.0 | 113.5 | 123.0 | 191.0 | 200.0 | 214.2 |
| | | 3 | 128.0 | 138.0 | 145.2 | 133.8 | 145.0 | 161.0 | 240.8 | 250.0 | 264.2 |
| 3000 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 96.0 | 101.0 | 106.0 | 96.0 | 101.0 | 106.0 | 150.8 | 157.0 | 164.0 |
| | | 2 | 113.0 | 117.0 | 123.0 | 113.8 | 120.0 | 126.0 | 192.8 | 208.0 | 217.2 |
| | | 3 | 135.0 | 143.0 | 155.2 | 140.0 | 151.0 | 168.0 | 240.0 | 258.0 | 272.0 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 96.0 | 101.0 | 106.0 | 96.0 | 101.0 | 106.0 | 150.8 | 156.5 | 164.0 |
| | | 2 | 112.8 | 117.0 | 122.2 | 113.8 | 120.0 | 126.0 | 191.8 | 208.0 | 217.2 |
| | | 3 | 135.0 | 143.0 | 155.2 | 140.0 | 151.0 | 168.0 | 240.0 | 258.0 | 272.0 |
| 4000 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 101.5 | 106.0 | 110.0 | 101.5 | 106.0 | 110.0 | 150.0 | 157.0 | 165.5 |
| | | 2 | 115.0 | 118.0 | 127.5 | 119.0 | 124.0 | 136.5 | 190.5 | 205.0 | 215.0 |
| | | 3 | 140.0 | 145.0 | 162.0 | 148.0 | 164.0 | 179.0 | 238.0 | 256.0 | 267.0 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 101.8 | 106.0 | 111.0 | 101.8 | 106.0 | 111.0 | 151.0 | 158.0 | 166.0 |
| | | 2 | 115.8 | 119.0 | 129.0 | 119.0 | 124.0 | 136.2 | 190.8 | 205.0 | 215.0 |
| | | 3 | 140.0 | 148.0 | 164.2 | 148.8 | 166.5 | 179.0 | 238.0 | 256.0 | 267.0 |
| 6000 | ${}_1\widehat{\mathcal{B}}_J$ | 1 | 105.0 | 109.0 | 115.0 | 105.0 | 109.0 | 115.0 | 155.0 | 160.5 | 168.2 |
| | | 2 | 116.0 | 123.0 | 141.0 | 120.0 | 128.5 | 145.0 | 200.0 | 210.0 | 221.0 |
| | | 3 | 144.8 | 158.5 | 171.0 | 157.8 | 170.5 | 179.0 | 249.0 | 260.0 | 273.2 |
| | ${}_2\widehat{\mathcal{B}}_J$ | 1 | 104.2 | 107.5 | 113.8 | 104.2 | 107.5 | 113.8 | 157.0 | 163.0 | 169.5 |
| | | 2 | 116.2 | 123.0 | 137.5 | 119.0 | 127.5 | 142.2 | 198.0 | 211.0 | 222.5 |
| | | 3 | 142.5 | 157.5 | 167.8 | 158.8 | 169.0 | 178.8 | 246.2 | 258.5 | 274.8 |

Table B.4. Number of wavelet coefficients using the Symlet 3 wavelet for the density 2D Smooth comb (Figure 4.7 (d)). Corresponding table for Hellinger distance is 4.4

## B.3 Image analysis results

| **KM** | **Affinity** | $R^p$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | **Total** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Linear | 3 | 0.727 | 0.915 | 0.568 | 0.756 | 0.590 | 0.499 | 0.433 | 0.896 | 0.930 | 0.917 | 0.723 |
| | | 8 | 0.728 | 0.915 | 0.525 | 0.753 | 0.645 | 0.513 | 0.477 | 0.908 | 0.941 | 0.930 | 0.734 |

Continued on next page

Table B.5 – continued from previous page

| KM | Affinity | $R^p$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 21 | 0.723 | 0.917 | 0.541 | 0.764 | 0.615 | 0.498 | 0.477 | 0.901 | 0.932 | 0.949 | 0.732 |
| | 0.2 | 3 | 0.575 | 0.906 | 0.607 | 0.696 | 0.608 | 0.456 | 0.525 | 0.915 | 0.874 | 0.928 | 0.709 |
| | | 8 | 0.616 | 0.927 | 0.605 | 0.705 | 0.653 | 0.428 | 0.558 | 0.905 | 0.880 | 0.920 | 0.720 |
| | | 21 | 0.705 | 0.923 | 0.555 | 0.693 | 0.661 | 0.408 | 0.520 | 0.911 | 0.938 | 0.942 | 0.726 |
| | 0.4 | 3 | 0.719 | 0.920 | 0.576 | 0.745 | 0.648 | 0.529 | 0.477 | 0.888 | 0.932 | 0.954 | 0.739 |
| | | 8 | 0.748 | 0.930 | 0.575 | 0.766 | 0.606 | 0.556 | 0.474 | 0.904 | 0.941 | 0.902 | 0.740 |
| | | 21 | 0.743 | 0.904 | 0.540 | 0.726 | 0.617 | 0.541 | 0.519 | 0.907 | 0.942 | 0.946 | 0.739 |
| | 0.6 | 3 | 0.738 | 0.919 | 0.553 | 0.787 | 0.601 | 0.488 | 0.429 | 0.889 | 0.931 | 0.946 | 0.728 |
| | | 8 | 0.741 | 0.921 | 0.543 | 0.756 | 0.624 | 0.528 | 0.490 | 0.902 | 0.947 | 0.943 | 0.739 |
| | | 21 | 0.722 | 0.926 | 0.568 | 0.671 | 0.630 | 0.483 | 0.493 | 0.911 | 0.923 | 0.941 | 0.727 |
| | 0.8 | 3 | 0.740 | 0.923 | 0.572 | 0.790 | 0.615 | 0.482 | 0.415 | 0.893 | 0.927 | 0.931 | 0.729 |
| | | 8 | 0.750 | 0.913 | 0.565 | 0.763 | 0.643 | 0.512 | 0.488 | 0.907 | 0.944 | 0.924 | 0.741 |
| | | 21 | 0.712 | 0.926 | 0.556 | 0.731 | 0.597 | 0.495 | 0.507 | 0.896 | 0.937 | 0.945 | 0.730 |
| | 1.0 | 3 | 0.745 | 0.915 | 0.568 | 0.769 | 0.627 | 0.510 | 0.414 | 0.882 | 0.932 | 0.929 | 0.729 |
| | | 8 | 0.721 | 0.919 | 0.567 | 0.733 | 0.640 | 0.509 | 0.508 | 0.912 | 0.943 | 0.925 | 0.738 |
| | | 21 | 0.714 | 0.919 | 0.548 | 0.776 | 0.556 | 0.496 | 0.515 | 0.899 | 0.937 | 0.947 | 0.731 |
| 50 | Linear | 3 | 0.760 | 0.926 | 0.562 | 0.734 | 0.680 | 0.580 | 0.456 | 0.900 | 0.931 | 0.945 | 0.747 |
| | | 8 | 0.762 | 0.923 | 0.599 | 0.769 | 0.686 | 0.583 | 0.471 | 0.912 | 0.947 | 0.950 | 0.760 |
| | | 21 | 0.757 | 0.944 | 0.568 | 0.754 | 0.669 | 0.589 | 0.489 | 0.909 | 0.927 | 0.949 | 0.755 |
| | 0.2 | 3 | 0.655 | 0.936 | 0.582 | 0.738 | 0.613 | 0.600 | 0.538 | 0.907 | 0.907 | 0.956 | 0.743 |
| | | 8 | 0.654 | 0.944 | 0.614 | 0.718 | 0.693 | 0.588 | 0.564 | 0.930 | 0.895 | 0.940 | 0.754 |
| | | 21 | 0.733 | 0.940 | 0.616 | 0.744 | 0.667 | 0.560 | 0.544 | 0.935 | 0.952 | 0.930 | 0.762 |
| | 0.4 | 3 | 0.748 | 0.939 | 0.587 | 0.761 | 0.640 | 0.625 | 0.472 | 0.917 | 0.939 | 0.941 | 0.757 |
| | | 8 | 0.737 | 0.941 | 0.587 | 0.766 | 0.664 | 0.643 | 0.489 | 0.910 | 0.949 | 0.942 | 0.763 |
| | | 21 | 0.777 | 0.939 | 0.581 | 0.752 | 0.669 | 0.628 | 0.514 | 0.932 | 0.936 | 0.941 | 0.767 |
| | 0.6 | 3 | 0.734 | 0.933 | 0.578 | 0.763 | 0.665 | 0.593 | 0.515 | 0.916 | 0.930 | 0.939 | 0.757 |
| | | 8 | 0.764 | 0.935 | 0.605 | 0.752 | 0.675 | 0.613 | 0.475 | 0.916 | 0.942 | 0.942 | 0.762 |
| | | 21 | 0.737 | 0.945 | 0.577 | 0.736 | 0.664 | 0.595 | 0.527 | 0.921 | 0.933 | 0.937 | 0.757 |
| | 0.8 | 3 | 0.741 | 0.930 | 0.586 | 0.758 | 0.667 | 0.562 | 0.460 | 0.897 | 0.925 | 0.956 | 0.748 |
| | | 8 | 0.766 | 0.931 | 0.584 | 0.756 | 0.672 | 0.608 | 0.496 | 0.916 | 0.946 | 0.938 | 0.761 |
| | | 21 | 0.760 | 0.944 | 0.573 | 0.753 | 0.659 | 0.592 | 0.484 | 0.915 | 0.933 | 0.948 | 0.756 |
| | 1.0 | 3 | 0.761 | 0.926 | 0.601 | 0.773 | 0.632 | 0.573 | 0.455 | 0.907 | 0.932 | 0.950 | 0.751 |
| | | 8 | 0.760 | 0.927 | 0.588 | 0.751 | 0.676 | 0.590 | 0.481 | 0.911 | 0.937 | 0.952 | 0.757 |
| | | 21 | 0.736 | 0.952 | 0.573 | 0.761 | 0.671 | 0.599 | 0.484 | 0.913 | 0.929 | 0.947 | 0.756 |
| 75 | Linear | 3 | 0.746 | 0.939 | 0.591 | 0.754 | 0.678 | 0.627 | 0.497 | 0.915 | 0.934 | 0.947 | 0.763 |
| | | 8 | 0.753 | 0.940 | 0.598 | 0.774 | 0.686 | 0.614 | 0.545 | 0.913 | 0.948 | 0.953 | 0.772 |
| | | 21 | 0.751 | 0.951 | 0.595 | 0.783 | 0.680 | 0.620 | 0.528 | 0.926 | 0.942 | 0.955 | 0.773 |
| | 0.2 | 3 | 0.679 | 0.946 | 0.602 | 0.770 | 0.690 | 0.681 | 0.529 | 0.919 | 0.918 | 0.960 | 0.769 |
| | | 8 | 0.719 | 0.947 | 0.636 | 0.737 | 0.698 | 0.657 | 0.508 | 0.922 | 0.902 | 0.947 | 0.767 |
| | | 21 | 0.752 | 0.950 | 0.633 | 0.749 | 0.668 | 0.629 | 0.552 | 0.937 | 0.947 | 0.939 | 0.776 |
| | 0.4 | 3 | 0.746 | 0.951 | 0.605 | 0.784 | 0.657 | 0.664 | 0.510 | 0.919 | 0.946 | 0.948 | 0.773 |
| | | 8 | 0.744 | 0.945 | 0.615 | 0.795 | 0.680 | 0.685 | 0.516 | 0.922 | 0.950 | 0.946 | 0.780 |
| | | 21 | 0.762 | 0.942 | 0.597 | 0.769 | 0.666 | 0.662 | 0.536 | 0.938 | 0.941 | 0.947 | 0.776 |

Continued on next page

**Table B.5 – continued from previous page**

| KM | Affinity | $R^p$ | $0$ | $1$ | $2$ | $3$ | $4$ | $5$ | $6$ | $7$ | $8$ | $9$ | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 0.6 | 3 | 0.754 | 0.951 | 0.587 | 0.755 | 0.675 | 0.635 | 0.516 | 0.919 | 0.936 | 0.944 | 0.767 |
| | | 8 | 0.752 | 0.948 | 0.603 | 0.783 | 0.676 | 0.637 | 0.505 | 0.920 | 0.942 | 0.948 | 0.771 |
| | | 21 | 0.755 | 0.948 | 0.589 | 0.769 | 0.689 | 0.644 | 0.516 | 0.931 | 0.938 | 0.945 | 0.772 |
| | 0.8 | 3 | 0.740 | 0.947 | 0.569 | 0.763 | 0.693 | 0.618 | 0.477 | 0.908 | 0.936 | 0.956 | 0.761 |
| | | 8 | 0.775 | 0.942 | 0.601 | 0.771 | 0.692 | 0.657 | 0.514 | 0.911 | 0.947 | 0.951 | 0.776 |
| | | 21 | 0.744 | 0.955 | 0.590 | 0.761 | 0.694 | 0.644 | 0.510 | 0.936 | 0.943 | 0.945 | 0.772 |
| | 1.0 | 3 | 0.746 | 0.942 | 0.576 | 0.740 | 0.695 | 0.618 | 0.466 | 0.925 | 0.926 | 0.937 | 0.757 |
| | | 8 | 0.750 | 0.943 | 0.594 | 0.771 | 0.697 | 0.625 | 0.515 | 0.913 | 0.946 | 0.957 | 0.771 |
| | | 21 | 0.740 | 0.957 | 0.587 | 0.769 | 0.689 | 0.634 | 0.491 | 0.916 | 0.939 | 0.954 | 0.767 |
| 125 | Linear | 3 | 0.731 | 0.944 | 0.611 | 0.784 | 0.691 | 0.677 | 0.519 | 0.927 | 0.945 | 0.956 | 0.779 |
| | | 8 | 0.772 | 0.955 | 0.612 | 0.776 | 0.708 | 0.672 | 0.520 | 0.921 | 0.950 | 0.965 | 0.785 |
| | | 21 | 0.753 | 0.954 | 0.640 | 0.806 | 0.704 | 0.688 | 0.518 | 0.937 | 0.945 | 0.960 | 0.790 |
| | 0.2 | 3 | 0.702 | 0.958 | 0.635 | 0.783 | 0.679 | 0.701 | 0.534 | 0.931 | 0.934 | 0.958 | 0.781 |
| | | 8 | 0.755 | 0.953 | 0.670 | 0.760 | 0.685 | 0.702 | 0.524 | 0.940 | 0.933 | 0.957 | 0.788 |
| | | 21 | 0.763 | 0.957 | 0.662 | 0.790 | 0.668 | 0.703 | 0.547 | 0.940 | 0.952 | 0.943 | 0.793 |
| | 0.4 | 3 | 0.745 | 0.953 | 0.618 | 0.800 | 0.666 | 0.714 | 0.536 | 0.925 | 0.943 | 0.962 | 0.786 |
| | | 8 | 0.760 | 0.952 | 0.640 | 0.790 | 0.664 | 0.713 | 0.531 | 0.936 | 0.945 | 0.954 | 0.788 |
| | | 21 | 0.764 | 0.953 | 0.602 | 0.788 | 0.715 | 0.710 | 0.532 | 0.943 | 0.938 | 0.952 | 0.790 |
| | 0.6 | 3 | 0.739 | 0.953 | 0.600 | 0.780 | 0.676 | 0.679 | 0.543 | 0.934 | 0.944 | 0.951 | 0.780 |
| | | 8 | 0.769 | 0.953 | 0.610 | 0.789 | 0.705 | 0.678 | 0.507 | 0.939 | 0.945 | 0.950 | 0.784 |
| | | 21 | 0.765 | 0.951 | 0.612 | 0.782 | 0.689 | 0.681 | 0.522 | 0.939 | 0.942 | 0.953 | 0.784 |
| | 0.8 | 3 | 0.749 | 0.952 | 0.614 | 0.778 | 0.690 | 0.667 | 0.499 | 0.930 | 0.945 | 0.952 | 0.778 |
| | | 8 | 0.757 | 0.953 | 0.615 | 0.786 | 0.711 | 0.688 | 0.521 | 0.922 | 0.950 | 0.960 | 0.786 |
| | | 21 | 0.748 | 0.957 | 0.630 | 0.778 | 0.711 | 0.694 | 0.518 | 0.942 | 0.940 | 0.957 | 0.788 |
| | 1.0 | 3 | 0.741 | 0.951 | 0.606 | 0.786 | 0.688 | 0.661 | 0.498 | 0.928 | 0.946 | 0.957 | 0.776 |
| | | 8 | 0.780 | 0.952 | 0.610 | 0.776 | 0.690 | 0.684 | 0.517 | 0.920 | 0.946 | 0.964 | 0.784 |
| | | 21 | 0.758 | 0.955 | 0.622 | 0.786 | 0.697 | 0.679 | 0.508 | 0.942 | 0.939 | 0.956 | 0.784 |
| 175 | Linear | 3 | 0.746 | 0.950 | 0.621 | 0.808 | 0.691 | 0.706 | 0.539 | 0.935 | 0.944 | 0.961 | 0.790 |
| | | 8 | 0.769 | 0.951 | 0.640 | 0.794 | 0.714 | 0.705 | 0.535 | 0.935 | 0.950 | 0.961 | 0.795 |
| | | 21 | 0.763 | 0.958 | 0.643 | 0.804 | 0.718 | 0.699 | 0.506 | 0.945 | 0.949 | 0.956 | 0.794 |
| | 0.2 | 3 | 0.728 | 0.958 | 0.630 | 0.798 | 0.700 | 0.709 | 0.546 | 0.936 | 0.941 | 0.961 | 0.791 |
| | | 8 | 0.753 | 0.956 | 0.668 | 0.782 | 0.698 | 0.708 | 0.540 | 0.946 | 0.944 | 0.951 | 0.795 |
| | | 21 | 0.781 | 0.957 | 0.687 | 0.803 | 0.684 | 0.714 | 0.553 | 0.952 | 0.949 | 0.941 | **0.802** |
| | 0.4 | 3 | 0.748 | 0.953 | 0.655 | 0.813 | 0.679 | 0.728 | 0.545 | 0.930 | 0.948 | 0.964 | 0.796 |
| | | 8 | 0.771 | 0.954 | 0.640 | 0.806 | 0.699 | 0.732 | 0.523 | 0.938 | 0.944 | 0.962 | 0.797 |
| | | 21 | 0.766 | 0.956 | 0.636 | 0.804 | 0.721 | 0.740 | 0.545 | 0.930 | 0.936 | 0.960 | 0.799 |
| | 0.6 | 3 | 0.735 | 0.953 | 0.633 | 0.797 | 0.689 | 0.705 | 0.548 | 0.942 | 0.943 | 0.953 | 0.790 |
| | | 8 | 0.777 | 0.957 | 0.652 | 0.781 | 0.708 | 0.710 | 0.512 | 0.930 | 0.948 | 0.960 | 0.793 |
| | | 21 | 0.762 | 0.955 | 0.646 | 0.800 | 0.712 | 0.712 | 0.528 | 0.939 | 0.946 | 0.957 | 0.796 |
| | 0.8 | 3 | 0.725 | 0.955 | 0.636 | 0.794 | 0.687 | 0.706 | 0.538 | 0.934 | 0.945 | 0.960 | 0.788 |
| | | 8 | 0.768 | 0.954 | 0.659 | 0.792 | 0.699 | 0.713 | 0.518 | 0.932 | 0.952 | 0.961 | 0.795 |
| | | 21 | 0.769 | 0.958 | 0.645 | 0.799 | 0.695 | 0.712 | 0.533 | 0.947 | 0.945 | 0.956 | 0.796 |
| | 1.0 | 3 | 0.749 | 0.952 | 0.653 | 0.799 | 0.679 | 0.692 | 0.528 | 0.931 | 0.947 | 0.957 | 0.789 |

Continued on next page

Table B.5 – continued from previous page

| KM | Affinity | $R^p$ | $0$ | $1$ | $2$ | $3$ | $4$ | $5$ | $6$ | $7$ | $8$ | $9$ | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 8 | 0.768 | 0.956 | 0.638 | 0.789 | 0.717 | 0.708 | 0.522 | 0.922 | 0.948 | 0.965 | 0.793 |
| | | 21 | 0.763 | 0.958 | 0.652 | 0.794 | 0.713 | 0.704 | 0.512 | 0.940 | 0.943 | 0.956 | 0.794 |

Table B.5. Accuracy of landmark $k$-NN over Fashion-MNIST with different algorithmic choices and a modified Riemannian metric. **KM** is the number of Karcher means used in each image class. **Affinity** is either *linear* or of quadratic exponential decay with given sigma. $R^p$ lists the number of dimensions in the spectral embedding projection. Columns $0$ to $9$ the corresponding accuracy, with the overall accuracy in the column **total**

| KM | Affinity | $R^p$ | $0$ | $1$ | $2$ | $3$ | $4$ | $5$ | $6$ | $7$ | $8$ | $9$ | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 25 | Linear | 3 | 0.962 | 0.981 | 0.856 | 0.869 | 0.744 | 0.710 | 0.941 | 0.837 | 0.759 | 0.834 | 0.852 |
| | | 8 | 0.970 | 0.984 | 0.826 | 0.875 | 0.747 | 0.667 | 0.916 | 0.802 | 0.806 | 0.838 | 0.846 |
| | | 21 | 0.975 | 0.982 | 0.842 | 0.854 | 0.728 | 0.709 | 0.915 | 0.813 | 0.824 | 0.846 | 0.852 |
| | 0.2 | 3 | 0.968 | 0.969 | 0.842 | 0.805 | 0.784 | 0.580 | 0.943 | 0.812 | 0.826 | 0.692 | 0.826 |
| | | 8 | 0.978 | 0.971 | 0.871 | 0.825 | 0.764 | 0.681 | 0.922 | 0.799 | 0.789 | 0.807 | 0.844 |
| | | 21 | 0.979 | 0.974 | 0.865 | 0.825 | 0.749 | 0.743 | 0.944 | 0.863 | 0.846 | 0.860 | 0.867 |
| | 0.4 | 3 | 0.976 | 0.984 | 0.872 | 0.848 | 0.733 | 0.742 | 0.951 | 0.847 | 0.832 | 0.857 | 0.867 |
| | | 8 | 0.988 | 0.982 | 0.876 | 0.863 | 0.753 | 0.747 | 0.942 | 0.846 | 0.837 | 0.878 | 0.874 |
| | | 21 | 0.988 | 0.986 | 0.879 | 0.867 | 0.722 | 0.702 | 0.945 | 0.846 | 0.843 | 0.887 | 0.870 |
| | 0.6 | 3 | 0.979 | 0.979 | 0.845 | 0.867 | 0.745 | 0.719 | 0.941 | 0.824 | 0.791 | 0.866 | 0.858 |
| | | 8 | 0.986 | 0.982 | 0.835 | 0.870 | 0.738 | 0.730 | 0.928 | 0.830 | 0.815 | 0.867 | 0.861 |
| | | 21 | 0.981 | 0.985 | 0.850 | 0.873 | 0.681 | 0.717 | 0.922 | 0.797 | 0.706 | 0.900 | 0.845 |
| | 0.8 | 3 | 0.964 | 0.984 | 0.872 | 0.866 | 0.736 | 0.695 | 0.947 | 0.810 | 0.765 | 0.856 | 0.853 |
| | | 8 | 0.977 | 0.980 | 0.837 | 0.857 | 0.714 | 0.701 | 0.925 | 0.804 | 0.835 | 0.872 | 0.853 |
| | | 21 | 0.978 | 0.983 | 0.827 | 0.866 | 0.720 | 0.685 | 0.927 | 0.830 | 0.773 | 0.872 | 0.849 |
| | 1.0 | 3 | 0.972 | 0.982 | 0.856 | 0.863 | 0.742 | 0.702 | 0.945 | 0.832 | 0.771 | 0.833 | 0.853 |
| | | 8 | 0.977 | 0.982 | 0.835 | 0.862 | 0.747 | 0.700 | 0.930 | 0.788 | 0.781 | 0.837 | 0.847 |
| | | 21 | 0.969 | 0.984 | 0.831 | 0.859 | 0.697 | 0.701 | 0.899 | 0.829 | 0.790 | 0.857 | 0.845 |
| 50 | Linear | 3 | 0.980 | 0.989 | 0.878 | 0.872 | 0.785 | 0.785 | 0.945 | 0.865 | 0.850 | 0.877 | 0.885 |
| | | 8 | 0.982 | 0.985 | 0.871 | 0.890 | 0.786 | 0.772 | 0.947 | 0.869 | 0.848 | 0.893 | 0.887 |
| | | 21 | 0.985 | 0.986 | 0.869 | 0.873 | 0.780 | 0.761 | 0.956 | 0.866 | 0.830 | 0.883 | 0.881 |
| | 0.2 | 3 | 0.979 | 0.974 | 0.864 | 0.850 | 0.822 | 0.772 | 0.952 | 0.838 | 0.839 | 0.859 | 0.877 |
| | | 8 | 0.986 | 0.981 | 0.906 | 0.861 | 0.826 | 0.736 | 0.947 | 0.811 | 0.860 | 0.859 | 0.880 |
| | | 21 | 0.989 | 0.982 | 0.898 | 0.859 | 0.803 | 0.818 | 0.950 | 0.881 | 0.887 | 0.900 | 0.898 |
| | 0.4 | 3 | 0.980 | 0.983 | 0.898 | 0.858 | 0.789 | 0.810 | 0.954 | 0.885 | 0.874 | 0.884 | 0.893 |
| | | 8 | 0.987 | 0.987 | 0.905 | 0.878 | 0.817 | 0.828 | 0.950 | 0.871 | 0.872 | 0.900 | 0.901 |
| | | 21 | 0.989 | 0.987 | 0.900 | 0.869 | 0.785 | 0.807 | 0.954 | 0.875 | 0.875 | 0.897 | 0.896 |
| | 0.6 | 3 | 0.980 | 0.989 | 0.873 | 0.873 | 0.764 | 0.787 | 0.962 | 0.871 | 0.857 | 0.901 | 0.888 |
| | | 8 | 0.984 | 0.987 | 0.890 | 0.884 | 0.788 | 0.796 | 0.949 | 0.863 | 0.863 | 0.896 | 0.892 |
| | | 21 | 0.990 | 0.986 | 0.888 | 0.884 | 0.782 | 0.793 | 0.959 | 0.863 | 0.838 | 0.897 | 0.890 |
| | 0.8 | 3 | 0.979 | 0.990 | 0.884 | 0.857 | 0.752 | 0.780 | 0.962 | 0.867 | 0.850 | 0.887 | 0.883 |

Continued on next page

Table B.6 – continued from previous page

| KM | Affinity | $R^p$ | $0$ | $1$ | $2$ | $3$ | $4$ | $5$ | $6$ | $7$ | $8$ | $9$ | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 8 | 0.986 | 0.986 | 0.879 | 0.890 | 0.791 | 0.788 | 0.951 | 0.862 | 0.847 | 0.882 | 0.888 |
| | | 21 | 0.989 | 0.987 | 0.880 | 0.887 | 0.783 | 0.784 | 0.954 | 0.871 | 0.846 | 0.889 | 0.889 |
| | 1.0 | 3 | 0.981 | 0.988 | 0.883 | 0.871 | 0.774 | 0.764 | 0.963 | 0.871 | 0.850 | 0.892 | 0.886 |
| | | 8 | 0.983 | 0.986 | 0.890 | 0.891 | 0.780 | 0.783 | 0.949 | 0.858 | 0.850 | 0.887 | 0.888 |
| | | 21 | 0.985 | 0.986 | 0.873 | 0.878 | 0.791 | 0.783 | 0.954 | 0.869 | 0.840 | 0.885 | 0.887 |
| 75 | Linear | 3 | 0.983 | 0.990 | 0.895 | 0.880 | 0.821 | 0.802 | 0.957 | 0.878 | 0.871 | 0.893 | 0.899 |
| | | 8 | 0.986 | 0.988 | 0.888 | 0.902 | 0.836 | 0.813 | 0.954 | 0.877 | 0.864 | 0.899 | 0.903 |
| | | 21 | 0.987 | 0.987 | 0.880 | 0.882 | 0.829 | 0.800 | 0.968 | 0.879 | 0.867 | 0.891 | 0.899 |
| | 0.2 | 3 | 0.982 | 0.979 | 0.872 | 0.870 | 0.843 | 0.780 | 0.958 | 0.881 | 0.840 | 0.898 | 0.892 |
| | | 8 | 0.988 | 0.982 | 0.914 | 0.877 | 0.870 | 0.801 | 0.954 | 0.837 | 0.878 | 0.897 | 0.901 |
| | | 21 | 0.989 | 0.983 | 0.912 | 0.869 | 0.843 | 0.821 | 0.960 | 0.884 | 0.908 | 0.913 | 0.910 |
| | 0.4 | 3 | 0.981 | 0.990 | 0.913 | 0.882 | 0.829 | 0.849 | 0.958 | 0.884 | 0.887 | 0.908 | 0.910 |
| | | 8 | 0.989 | 0.987 | 0.918 | 0.879 | 0.836 | 0.862 | 0.961 | 0.875 | 0.891 | 0.909 | 0.912 |
| | | 21 | 0.988 | 0.989 | 0.915 | 0.879 | 0.824 | 0.846 | 0.966 | 0.897 | 0.880 | 0.902 | 0.910 |
| | 0.6 | 3 | 0.985 | 0.991 | 0.911 | 0.873 | 0.804 | 0.819 | 0.963 | 0.882 | 0.899 | 0.899 | 0.904 |
| | | 8 | 0.987 | 0.990 | 0.903 | 0.896 | 0.821 | 0.825 | 0.964 | 0.888 | 0.868 | 0.902 | 0.906 |
| | | 21 | 0.988 | 0.988 | 0.902 | 0.892 | 0.821 | 0.826 | 0.966 | 0.895 | 0.865 | 0.895 | 0.906 |
| | 0.8 | 3 | 0.983 | 0.992 | 0.897 | 0.878 | 0.807 | 0.806 | 0.966 | 0.886 | 0.874 | 0.894 | 0.900 |
| | | 8 | 0.984 | 0.989 | 0.899 | 0.902 | 0.832 | 0.824 | 0.963 | 0.877 | 0.861 | 0.896 | 0.904 |
| | | 21 | 0.987 | 0.989 | 0.892 | 0.897 | 0.822 | 0.823 | 0.971 | 0.889 | 0.862 | 0.883 | 0.903 |
| | 1.0 | 3 | 0.983 | 0.990 | 0.896 | 0.887 | 0.800 | 0.799 | 0.962 | 0.882 | 0.859 | 0.903 | 0.898 |
| | | 8 | 0.986 | 0.988 | 0.908 | 0.896 | 0.817 | 0.817 | 0.958 | 0.879 | 0.859 | 0.906 | 0.903 |
| | | 21 | 0.986 | 0.990 | 0.883 | 0.895 | 0.831 | 0.825 | 0.960 | 0.885 | 0.862 | 0.904 | 0.904 |
| 125 | Linear | 3 | 0.988 | 0.992 | 0.905 | 0.890 | 0.849 | 0.853 | 0.967 | 0.891 | 0.871 | 0.913 | 0.913 |
| | | 8 | 0.988 | 0.990 | 0.912 | 0.906 | 0.854 | 0.852 | 0.973 | 0.900 | 0.881 | 0.917 | 0.919 |
| | | 21 | 0.986 | 0.989 | 0.902 | 0.901 | 0.861 | 0.850 | 0.970 | 0.901 | 0.887 | 0.907 | 0.917 |
| | 0.2 | 3 | 0.984 | 0.984 | 0.909 | 0.877 | 0.869 | 0.853 | 0.968 | 0.904 | 0.876 | 0.921 | 0.916 |
| | | 8 | 0.988 | 0.986 | 0.933 | 0.902 | 0.892 | 0.838 | 0.956 | 0.865 | 0.869 | 0.918 | 0.916 |
| | | 21 | 0.988 | 0.984 | 0.922 | 0.889 | 0.864 | 0.849 | 0.967 | 0.888 | 0.923 | 0.925 | 0.921 |
| | 0.4 | 3 | 0.988 | 0.992 | 0.932 | 0.887 | 0.867 | 0.863 | 0.962 | 0.894 | 0.911 | 0.917 | 0.923 |
| | | 8 | 0.990 | 0.991 | 0.932 | 0.895 | 0.875 | 0.882 | 0.971 | 0.896 | 0.905 | 0.908 | 0.926 |
| | | 21 | 0.989 | 0.991 | 0.930 | 0.902 | 0.876 | 0.866 | 0.971 | 0.904 | 0.895 | 0.913 | 0.925 |
| | 0.6 | 3 | 0.987 | 0.991 | 0.923 | 0.879 | 0.852 | 0.861 | 0.970 | 0.891 | 0.889 | 0.910 | 0.917 |
| | | 8 | 0.989 | 0.992 | 0.919 | 0.897 | 0.860 | 0.862 | 0.974 | 0.902 | 0.892 | 0.906 | 0.921 |
| | | 21 | 0.988 | 0.990 | 0.921 | 0.900 | 0.860 | 0.869 | 0.975 | 0.911 | 0.878 | 0.912 | 0.922 |
| | 0.8 | 3 | 0.987 | 0.993 | 0.911 | 0.894 | 0.857 | 0.853 | 0.964 | 0.886 | 0.881 | 0.904 | 0.914 |
| | | 8 | 0.987 | 0.990 | 0.916 | 0.904 | 0.855 | 0.853 | 0.970 | 0.897 | 0.883 | 0.902 | 0.917 |
| | | 21 | 0.987 | 0.991 | 0.907 | 0.901 | 0.865 | 0.859 | 0.976 | 0.906 | 0.885 | 0.907 | 0.920 |
| | 1.0 | 3 | 0.986 | 0.992 | 0.912 | 0.890 | 0.857 | 0.859 | 0.960 | 0.898 | 0.874 | 0.913 | 0.916 |
| | | 8 | 0.987 | 0.990 | 0.916 | 0.911 | 0.852 | 0.845 | 0.971 | 0.898 | 0.863 | 0.904 | 0.915 |
| | | 21 | 0.987 | 0.990 | 0.910 | 0.900 | 0.854 | 0.843 | 0.973 | 0.910 | 0.880 | 0.904 | 0.917 |
| | Linear | 3 | 0.988 | 0.992 | 0.917 | 0.900 | 0.867 | 0.876 | 0.972 | 0.896 | 0.906 | 0.920 | 0.924 |
| | | 8 | 0.988 | 0.992 | 0.926 | 0.909 | 0.874 | 0.867 | 0.976 | 0.901 | 0.896 | 0.916 | 0.926 |

Continued on next page

Table B.6 – continued from previous page

| KM | Affinity | $R^p$ | $0$ | $1$ | $2$ | $3$ | $4$ | $5$ | $6$ | $7$ | $8$ | $9$ | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 21 | 0.988 | 0.992 | 0.922 | 0.906 | 0.888 | 0.885 | 0.975 | 0.912 | 0.887 | 0.914 | 0.928 |
| | 0.2 | 3 | 0.987 | 0.988 | 0.918 | 0.883 | 0.893 | 0.868 | 0.972 | 0.913 | 0.893 | 0.924 | 0.925 |
| | | 8 | 0.985 | 0.989 | 0.939 | 0.908 | 0.883 | 0.854 | 0.954 | 0.897 | 0.908 | 0.919 | 0.925 |
| | | 21 | 0.989 | 0.985 | 0.933 | 0.902 | 0.875 | 0.867 | 0.969 | 0.890 | 0.924 | 0.938 | 0.928 |
| | 0.4 | 3 | 0.988 | 0.991 | 0.928 | 0.894 | 0.879 | 0.880 | 0.970 | 0.901 | 0.910 | 0.924 | 0.928 |
| | | 8 | 0.990 | 0.991 | 0.934 | 0.903 | 0.897 | 0.892 | 0.977 | 0.906 | 0.914 | 0.918 | **0.933** |
| | | 21 | 0.989 | 0.991 | 0.936 | 0.910 | 0.891 | 0.886 | 0.975 | 0.907 | 0.903 | 0.922 | 0.932 |
| | 0.6 | 3 | 0.986 | 0.992 | 0.927 | 0.892 | 0.881 | 0.879 | 0.973 | 0.898 | 0.900 | 0.917 | 0.926 |
| | | 8 | 0.987 | 0.992 | 0.926 | 0.902 | 0.885 | 0.880 | 0.979 | 0.902 | 0.896 | 0.913 | 0.927 |
| | | 21 | 0.988 | 0.990 | 0.928 | 0.908 | 0.880 | 0.886 | 0.977 | 0.907 | 0.889 | 0.911 | 0.927 |
| | 0.8 | 3 | 0.988 | 0.992 | 0.924 | 0.897 | 0.866 | 0.866 | 0.967 | 0.900 | 0.895 | 0.925 | 0.924 |
| | | 8 | 0.988 | 0.992 | 0.921 | 0.904 | 0.877 | 0.867 | 0.975 | 0.906 | 0.885 | 0.907 | 0.924 |
| | | 21 | 0.989 | 0.991 | 0.919 | 0.902 | 0.885 | 0.878 | 0.974 | 0.913 | 0.894 | 0.913 | 0.927 |
| | 1.0 | 3 | 0.986 | 0.992 | 0.925 | 0.894 | 0.869 | 0.866 | 0.969 | 0.887 | 0.894 | 0.914 | 0.921 |
| | | 8 | 0.990 | 0.992 | 0.928 | 0.907 | 0.876 | 0.865 | 0.977 | 0.904 | 0.893 | 0.918 | 0.926 |
| | | 21 | 0.986 | 0.991 | 0.919 | 0.905 | 0.888 | 0.869 | 0.976 | 0.911 | 0.896 | 0.916 | 0.927 |

Table B.6. Accuracy of landmark $k$-NN over MNIST with different algorithmic choices. **KM** is the number of Karcher means used in each image class. **Affinity** is either *linear* or of quadratic exponential decay with given sigma. $R^p$ list the number of dimensions in the spectral embedding projection. Columns $0$ to $9$ list the corresponding accuracy, with the total classification accuracy across the test set in the column **total**.

# Appendix C

# Software implementations

Implementation of SPWDE estimator:

https://github.com/carlosayam/pywde.

Of particular interest:

- pywde/pywde_ext.py contains an extension to PyWavelets for multivariate-tensor products.
- pywde/simple_estimator.py is an implementation of the classic wavelet-based estimator of (2.25).
- pywde/square_root_estimator.py has the implementation used for the linear SPWDE.
- pywde/spwde.py contains implementations for the best resolution level and hard-thresholding algorithms, the non-linear SPWDE.

Commands to generate simulations, results, tables and plots are in the following repository:

https://github.com/carlosayam/pywde-run.

# Bibliography

Abramson, I. S. (1982). On bandwidth variation in kernel estimates-a square root law. The Annals of Statistics, 1217-1223.

Adler, A., Boublil, D., & Zibulevsky, M. (2017). Block-based compressed sensing of images via deep learning. In 2017 IEEE 19th International Workshop on Multimedia Signal Processing (MMSP) (pp. 1-6). IEEE.

Aldroubi, A., Cabrelli, C., & Molter, U. M. (2004). Wavelets on irregular grids with arbitrary dilation matrices and frame atoms for $L^2(\mathbb{R}^d)$. Applied and Computational Harmonic Analysis, 17(2), 119-140.

Antoine, J. P., Demanet, L., Jacques, L., & Vandergheynst, P. (2002). Wavelets on the sphere: Implementation and approximations. Applied and Computational Harmonic Analysis, 13(3), 177-200.

Antoniadis, A. (2007). Wavelet methods in statistics: Some recent developments and their applications. Statistics Surveys, 1, 16-55.

Aya-Moreno, C., Geenens, G., & Penev, S. (2018). Shape-preserving wavelet-based multivariate density estimation. Journal of Multivariate Analysis, 168, 30-47.

Baryshnikov, Y., Penrose, M., & Yukich, J. (2009). Gaussian limits for generalized spacings. The Annals of Applied Probability, 19, 158-185.

Bhattacharyya, A. (1946). On a measure of divergence between two multinomial populations. Sankhyā: the indian journal of statistics, 401-406.

Biau, G., & Devroye, L., Lectures on the Nearest Neighbor Method. Springer Series in the Data Sciences, Springer, 2015.

Blalock, D., Ortiz, J. J. G., Frankle, J., & Guttag, J. (2020). What is the state of neural network pruning?. arXiv preprint arXiv:2003.03033.

Blei, D. M. (2014). Build, compute, critique, repeat: Data analysis with latent variable models. Annual Review of Statistics and Its Application.

Bochner, S. (1955). Harmonic analysis and the theory of probability. University of California Press.

Bowman, A. W. (1984). An alternative method of cross-validation for the smoothing of density estimates. Biometrika, 71(2), 353-360.

Box, G. E. (1976). Science and statistics. Journal of the American Statistical Association, 71(356), 791-799.

Breiman, L., Meisel, W., & Purcell, E. (1977). Variable kernel estimates of multivariate densities. Technometrics, 19(2), 135-144.

Brown, L., Cai, T., Zhang, R., Zhao, L., & Zhou, H. (2010). The root–unroot algorithm for density estimation as implemented via wavelet block thresholding. Probability theory and related fields, 146(3-4), 401.

Bunea, F., Tsybakov, A. B., & Wegkamp, M. H. (2007). Sparse density estimation with $\ell_1$ penalties. In International Conference on Computational Learning Theory (pp. 530-543). Springer, Berlin, Heidelberg.

Burman, P. (1985). A data dependent approach to density estimation. Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete, 69(4), 609-628.

Cai, D., & Chen, X. (2015). Large Scale Spectral Clustering Via Landmark-Based Sparse Representation. IEEE Transactions on Cybernetics, 45(8), 1669–1680.

Calderón, A. P. (1965). Intermediate spaces and interpolation, the complex method. Matematika, 9(3), 56-129.

Candès, E. J. (1998). Ridgelets: theory and applications (Doctoral dissertation, Stanford University).

Candès, E. J., & Donoho, D. L. (2004). New tight frames of curvelets and optimal representations of objects with piecewise $C^2$ singularities. Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences, 57(2), 219-266.

Candès, E. J., & Donoho, D. L. (2005). Continuous curvelet transform: II. Discretization and frames. Applied and Computational Harmonic Analysis, 19(2), 198-222.

Cao, R., Cuevas, A., & Manteiga, W. G. (1994). A comparative study of several smoothing methods in density estimation. Computational Statistics & Data Analysis, 17(2), 153-176.

Casazza, P. G., & Kutyniok, G. (Eds.). (2012). Finite frames: Theory and applications. Springer Science & Business Media.

Charon, N., & Trouvé, A. (2013). The varifold representation of nonoriented shapes for diffeomorphic registration. SIAM Journal on Imaging Sciences, 6(4), 2547-2580.

Chicken, E., & Cai, T. T. (2005). Block thresholding for density estimation: local and global adaptivity. Journal of Multivariate Analysis, 95(1), 76-106.

Chung, F. R., & Graham, F. C. (1997). Spectral graph theory (No. 92). American Mathematical Soc..

Cohen, A., Daubechies, I., & Feauveau, J. C. (1992). Biorthogonal bases of compactly supported wavelets. Communications on pure and applied mathematics, 45(5), 485-560.

Cosma, A., Scaillet, O., & von Sachs, R. (2007). Multivariate wavelet-based shape-preserving estimation for dependent observations. Bernoulli, 13, 301-329.

Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. IEEE transactions on information theory, 13(1), 21-27.

Craven, B. D. (1985). Generalized functions for applications. The ANZIAM Journal, 26(3), 362-374.

Dahlke, S. (1994). Multiresolution analysis, Haar bases and wavelets on Riemannian manifolds. In Wavelet Analysis and its Applications (Vol. 5, pp. 33-52). Academic Press.

Daubechies, I. (1988). Orthonormal bases of compactly supported wavelets. Communications on pure and applied mathematics, 41(7), 909-996.

Daubechies, I. (1992). Ten Lectures on Wavelets. Society for Industrial and Applied Mathematics.

Daubechies, I. (1993). Review: Ondelettes. Science, 262(5139), 1589-1591.

De Vries, A. (2006). Wavelets. FH Südwestfalen University of Applied Sciences, Haldener Straße, 182.

Dechevsky, L., & Penev, S. (1997). On shape preserving probabilistic wavelet approximators, Stochastic Anal. Appl., 15, 187-215.

Dechevsky, L., & Penev, S. (1998). On shape preserving wavelet estimators of cumulative distribution functions and densities, Stochastic Anal. Appl., 16, 423-462.

Delattre, S., & Fournier, N. (2017). On the Kozachenko-Leonenko entropy estimator. Journal of Statistical Planning and Inference, 185, 69-93.

Delyon, B., & Juditsky, A. (1996). On minimax wavelet estimators. Applied and Computational Harmonic Analysis, 3(3), 215-228.

Deng, Z., Chung, F. L., & Wang, S. (2008). FRSDE: Fast reduced set density estimator using minimal enclosing ball approximation. Pattern Recognition, 41(4), 1363-1372.

Deng, Z., Zhu, X., Cheng, D., Zong, M., & Zhang, S. (2016). Efficient $k-NN$ classification algorithm for big data. Neurocomputing, 195(C), 143–148.

Devroye, L., & Wagner, T. (1977). The strong uniform consistency of nearest-neighbor density estimates. The Annals of Statistics, 536-540.

Donoho, D.L., & Johnstone, I.M. (1994). Ideal spatial adaptation by wavelet shrinkage. Biometrika, 81, 425-455.

Donoho, D.L., & Johnstone, I.M. (1995). Adapting to unknown smoothness via wavelet shrinkage. Journal of the American Statistical Association, 90(432), 1200-1224.

Donoho, D.L., & Johnstone, I.M. (1996). Neo-classical minimax problems, thresholding, and adaptive function estimation. Bernoulli, 2, 39-62.

Donoho, D.L., & Johnstone, I.M. (1998). Minimax estimation via wavelet shrinkage. The Annals of Statistics, 26(3), 879-921.

Donoho, D.L., Johnstone, I.M., Kerkyacharian, G., & Picard, D. (1995). Wavelet shrinkage: Asymptopia?. Journal of the Royal Statistical Society: Series B (Methodological), 57(2), 301-337.

Donoho, D.L., Johnstone, I.M., Kerkyacharian, G., & Picard, D. (1996). Density estimation by wavelet thresholding. The Annals of Statistics, 24, 508-539.

Donoho, D. L. (2002). Emerging applications of geometric multiscale analysis. Proceedings of the ICM, Beijing 2002, vol. 1, 209-233.

Donoho, D. L. (2006). Compressed sensing. IEEE Transactions on information theory, 52(4), 1289-1306.

Doosti, H., & Hall, P. (2016). Making a non-parametric density estimator more attractive, and more accurate, by data perturbation. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 78(2), 445-462.

Duffin, R. J., & Schaeffer, A. C. (1952). A class of nonharmonic Fourier series. Transactions of the American Mathematical Society, 72(2), 341-366.

Dwork, N., O'Connor, D., Baron, C. A., Johnson, E. M., Kerr, A. B., Pauly, J. M., & Larson, P. E. (2020). Utilizing the Wavelet Transform's Structure in Compressed Sensing. arXiv preprint arXiv:2002.04150.

Ebner, B., Henze, N., & Yukich, J.E. (2018). Multivariate goodness-of-fit on flat and curved spaces via nearest neighbor distances. Journal of Multivariate Analysis, 165, 231-242.

Efron, B., & Hastie, T. (2016). Computer age statistical inference (Vol. 5). Cambridge University Press.

Efron, B., & Stein, C. (1981). The jackknife estimate of variance. The Annals of Statistics, 586-596.

Epanechnikov, V. A. (1969). Non-parametric estimation of a multivariate probability density. Theory of Probability & its Applications, 14(1), 153-158.

Evans, D. (2008). A law of large numbers for nearest neighbour statistics. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, 464(2100), 3175-3192.

Evans, D., Jones, A.J., & Schmidt, W.M. (2002). Asymptotic moments of near-neighbour distance distributions. Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences, 458(2028), 2839-2849.

Fan, J., Hall, P., Martin, M., & Patil, P. (1996). On the local smoothing of non-parametric curve estimators. Journal of the American Statistical Association, 91(433), 258-266.

Fix, E., & Hodges Jr, J. L. (1951). Discriminatory analysis: nonparametric discrimination, consistency properties. USAF School of Aviation Medicine.

Fix, E., & Hodges Jr, J. L. (1952). Discriminatory analysis-nonparametric discrimination: Small sample performance (No. UCB-11). California Univ Berkeley.

Gadat, S., Klein, T., & Marteau, C. (2016). Classification in general finite dimensional spaces with the $k$-nearest neighbor rule. The Annals of Statistics, 44(3), 982-1009.

Geenens, G. (2011). Curse of Dimensionality and related issues in nonparametric functional regression. Statistics Surveys, 5, 30-43.

Geenens, G., & Lafaye de Micheaux, P. (2020). The Hellinger correlation. Published online: 17 Aug 2020, Journal of the American Statistical Association, https://doi.org/10.1080/01621459.2020.1791132.

Geier, C. (2019). Training on test data: Removing near duplicates in Fashion-MNIST. arXiv preprint arXiv:1906.08255.

Geller, D., & Mayeli, A. (2009). Continuous wavelets on compact manifolds. Mathematische Zeitschrift, 262(4), 895.

Girolami, M., & He, C. (2003). Probability density estimation from optimally condensed data samples. IEEE Transactions on pattern analysis and machine intelligence, 25(10), 1253-1264.

Good, I. J., & Gaskins, R. A. (1971). Nonparametric roughness penalties for probability densities. Biometrika, 58(2), 255-277.

Grohs, P., & Wallner, J. (2009). Interpolatory wavelets for manifold-valued data. Applied and Computational Harmonic Analysis, 27(3), 325-333.

Grohs, P., Keiper, S., Kutyniok, G., & Schaefer, M. (2013). Alpha molecules: curvelets, shearlets, ridgelets, and beyond. In Wavelets and Sparsity XV (Vol. 8858, p. 885804). International Society for Optics and Photonics.

Grossmann, A., & Morlet, J. (1984). Decomposition of Hardy functions into square integrable wavelets of constant shape. SIAM journal on mathematical analysis, 15(4), 723-736.

Grove, K. (1976). Center of mass and G-local triviality of G-bundles. Proceedings of the American Mathematical Society, 54(1), 352-354.

Gu, C., & Qiu, C. (1993). Smoothing spline density estimation: Theory. The Annals of Statistics, 217-234.

Haar, A. (1910). Zur theorie der orthogonalen funktionensysteme. Mathematische Annalen, 69(3), 331-371. "On the Theory of Orthogonal Function Systems" translated by Georg Zimmermann.

Hald, A. (1999). On the history of maximum likelihood in relation to inverse probability and least squares. Statistical Science, 14(2), 214-222.

Hall, P. (1982). Limit theorems for stochastic measures of the accuracy of density estimators. Stochastic Processes and their Applications, 13(1), 11-25.

Hall, P. (1983). Large sample optimality of least squares cross-validation in density estimation. The Annals of Statistics, 11(4), 1156-1174.

Hall, P. (1983), On near neighbour estimates of a multivariate density. Journal of Multivariate Analysis, 13(1), 24-39.

Hall, P., & Marron, J. S. (1987). Extent to which least-squares cross-validation minimises integrated square error in nonparametric density estimation. Probability Theory and Related Fields, 74(4), 567-581.

Hall, P., Penev, S., Kerkyacharian, G., & Picard, D. (1997). Numerical performance of block thresholded wavelet estimators. Statistics and Computing, 7(2), 115-124.

Hall, P., & Penev, S. (2001). Cross-validation for choosing resolution level for nonlinear wavelet curve estimators. Bernoulli, 7(2), 317-341.

Hall, P., Penev, S., & Tran, J. (2018). Wavelet methods for erratic regression means in the presence of measurement error. Statistica Sinica, 28(4), 2289-2307.

Härdle, W., Kerkyacharian, G., Picard, D., & Tsybakov, A. (1998). Wavelets, Approximation and Statistical Applications. Lecture Notes in Statistics, Springer, New York.

Hardy, M. (2006). Combinatorics of Partial Derivatives. the electronic journal of combinatorics, 13(R1), 1.

Hazelton, M. L., & Cox, M. P. (2016). Bandwidth selection for kernel log-density estimation. Computational Statistics & Data Analysis, 103, 56-67.

Henze, N. (1988). A multivariate two-sample test based on the number of nearest neighbor type coincidences. The Annals of Statistics, 772-783.

Hoeffding, W. (1963). Probability Inequalities for Sums of Bounded Random Variables. Journal of the American Statistical Association, 58(301), 13–30.

Hong, X., Chen, S., & Harris, C. J. (2008). A forward-constrained regression algorithm for sparse kernel density estimation. IEEE Transactions on Neural Networks, 19(1), 193-198.

Izenman, A. J. (1991). Review papers: Recent developments in nonparametric density estimation. Journal of the American Statistical Association, 86(413), 205-224.

Jansen, M. (2003). Wavelet thresholding on non-equispaced data. In Nonlinear Estimation and Classification (pp. 261-271). Springer, New York, NY.

Jiang, M., & Provost, S. B. (2011). Improved orthogonal polynomial density estimates. Journal of Statistical Computation and Simulation, 81(11), 1495-1516.

Johnson, N.L., Kotz, S., & Balakrishnan, N. (1994). Continuous Univariate Distributions (Volume 1). John Wiley and Sons, New York.

Jorgensen, P. E. (2006). Analysis and probability: wavelets, signals, fractals (Vol. 234). Springer Science & Business Media.

Karcher, H. (1977). Riemannian center of mass and mollifier smoothing. Communications on pure and applied mathematics, 30(5), 509-541.

Karcher, H. (2014). Riemannian center of mass and so called Karcher mean. arXiv preprint arXiv:1407.2087.

Kaushik, A., Parthasarathy, H., & Sengar, P. S. (2014). A novel technique for PDF estimation using DSP methods. In 2014 International Conference on Signal Processing and Integrated Networks (SPIN) (pp. 46-49). IEEE.

Kerkyacharian, G., & Picard, D. (1993). Density estimation by kernel and wavelet methods: optimality of Besov spaces. Statistics & Probability Letters, 18(4), 327-336.

Kittipoom, P., Kutyniok, G., & Lim, W. Q. (2011). Irregular shearlet frames: Geometry and approximation properties. Journal of Fourier Analysis and Applications, 17(4), 604-639.

Koo, J. Y., & Kim, W. C. (1996). Wavelet density estimation by approximation of log-densities. Statistics & probability letters, 26(3), 271-278.

Kooperberg, C., & Stone, C. J. (1991). A study of logspline density estimation. Computational Statistics & Data Analysis, 12(3), 327-347.

Kovacs, J. A., Helmick, C., & Wriggers, W. (2017). A balanced approach to adaptive probability density estimation. Frontiers in molecular biosciences, 4, 25.

Kozachenko, L. F., & Leonenko, N. N. (1987). Sample estimate of the entropy of a random vector. Problemy Peredachi Informatsii, 23(2), 9-16.

Kristan, M., Leonardis, A., & Skočaj, D. (2011). Multivariate online kernel density estimation with Gaussian kernels. Pattern Recognition, 44(10-11), 2630-2642.

Kuljus, K., & Ranneby, B. (2015). Generalized maximum spacing estimation for multivariate observations. Scandinavian Journal of Statistics, 42(4), 1092-1108.

Labate, D., Lim, W. Q., Kutyniok, G., & Weiss, G. (2005). Sparse multidimensional representation using shearlets. In Wavelets XI (Vol. 5914, p. 59140U). International Society for Optics and Photonics.

Labate, D., Weiss, G., & Wilson, E. (2013). Wavelets. Notices of the AMS, 60(1).

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278–2324.

LeCun, Y., Cortes, C., & Burges, C. J. (1998). The MNIST database of handwritten digits, 1998. URL http://yann.lecun.com/exdb/mnist, 10, 34.

Lee, G. R., Gommers, R., Waselewski, F., Wohlfahrt, K., & O'Leary, A. (2019). PyWavelets: A Python package for wavelet analysis. Journal of Open Source Software, 4(36), 1237.

Lee, J. M. (2013). Smooth manifolds. In Introduction to Smooth Manifolds (pp. 1-31). Springer, New York.

Leonard, T. (1973). A Bayesian Method for Histograms. Biometrika, 60(2), 297-308.

Lin, J. (2019, January). The neural hype and comparisons against weak baselines. In ACM SIGIR Forum (Vol. 52, No. 2, pp. 40-51). New York, NY, USA: ACM.

Lin, Y., & Jeon, Y. (2006). Random forests and adaptive nearest neighbors. Journal of the American Statistical Association, 101(474), 578-590.

Loftsgaarden, D. O., & Quesenberry, C. P. (1965). A nonparametric estimate of a multivariate density function. The Annals of Mathematical Statistics, 36(3), 1049-1051.

Mack, Y. P., & Rosenblatt, M. (1979). Multivariate k-nearest neighbor density estimates. Journal of Multivariate Analysis, 9(1), 1-15.

Mallat, S. G. (1989). Multiresolution approximations and wavelet orthonormal bases of $L^2(\mathbb{R})$. Transactions of the American mathematical society, 315(1), 69-87.

Mallat, S. (1999). A wavelet tour of signal processing. Elsevier.

Marron, J. S. (1987). A comparison of cross-validation techniques in density estimation. The Annals of Statistics, 15(1), 152-162.

Marron, J. S., & Härdle, W. (1986). Random approximations to some measures of accuracy in nonparametric curve estimation. Journal of Multivariate Analysis, 20(1), 91-113.

Marron, J. S., & Wand, M. P. (1992). Exact mean integrated squared error. The Annals of Statistics, 712-736.

Marron, J. S., & Alonso, A. M. (2014). Overview of object oriented data analysis. Biometrical Journal, 56(5), 732-753.

McFadden, D. (2003). Economic choices. In T. Persson (ed.), Nobel Lectures in Economic Sciences 1996-2000, pp. 330-365, World Scientific, Singapore.

Meyer, Y., (1992). Wavelets and Operators. Cambridge University Press.

Miller, R. G. (1974). The jackknife-a review. Biometrika, 61(1), 1-15.

Mohlenkamp, M. J., & Pereyra, M. C. (2008). Wavelets, their friends, and what they can do for you (Vol. 8). European Mathematical Society.

Mondal, P.K., Biswas, M., & Ghosh, A.K. (2015). On high dimensional two-sample tests based on nearest neighbors, Journal of Multivariate Analysis, 141, 168-178.

Morlet, J. (1976). Seismic tomorrow, interferometry and quantum mechanics. In Geophysics (Vol. 41, No. 2, pp. 366-366). 8801 S Yale st, Tulsa, OK 74137: Soc Exploration Geophysicist.

Nason, G. P., & Silverman, B. W. (1994). The discrete wavelet transform in S. Journal of Computational and Graphical Statistics, 3(2), 163-191.

Nason, G. P. (1996). Wavelet shrinkage using cross-validation. Journal of the Royal Statistical Society: Series B (Methodological), 58(2), 463-479.

Nason, G. (2010). Wavelet methods in statistics with R. Springer Science & Business Media.

Ng, A. Y., Jordan, M. I., & Weiss, Y. (2002). On spectral clustering: Analysis and an algorithm. In Advances in neural information processing systems (pp. 849-856).

Nixon, M., & Aguado, A. (2019). Feature extraction and image processing for computer vision. Academic Press.

O'Sullivan, F. (1988). Fast computation of fully automated log-density and log-hazard estimators. SIAM Journal on scientific and statistical computing, 9(2), 363-379.

Omohundro, S. M. (1989). Five balltree construction algorithms (pp. 1-22). Berkeley: International Computer Science Institute.

Park, B. U., & Marron, J. S. (1990). Comparison of data-driven bandwidth selectors. Journal of the American Statistical Association, 85(409), 66-72.

Parker, J. R. (2010). Algorithms for image processing and computer vision. John Wiley & Sons.

Parzen, E. (1962). On estimation of a probability density function and mode. The Annals of Mathematical Statistics, 33(3), 1065-1076.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. the Journal of machine Learning research, 12, 2825-2830.

Peherstorfer, B., Pflüge, D., & Bungartz, H. J. (2014). Density estimation with adaptive sparse grids for large data sets. In Proceedings of the 2014 SIAM international conference on data mining (pp. 443-451). Society for Industrial and Applied Mathematics.

Penev, S., & Dechevsky, L. (1997). On non-negative wavelet-based density estimators, Journal of Nonparametric Statistics, 7, 365-394.

Pennec, X. (2006). Intrinsic statistics on Riemannian manifolds: Basic tools for geometric measurements. Journal of Mathematical Imaging and Vision, 25(1), 127.

Percus, A.G., & Martin, O.C. (1998). Scaling universalities of $k$th nearest neighbor distances on closed manifolds. Advances in Applied Mathematics, 21(3), 424-436.

Pesenson, I. (2015). Sampling, splines and frames on compact manifolds. GEM - International Journal on Geomathematics, 6(1), 43–81.

Peter, A.M., & Rangarajan, A. (2008). Maximum Likelihood Wavelet Density Estimation With Applications to Image and Shape Matching. IEEE Transactions on Image Processing, 17, 458-468.

Peter, A. M., Rangarajan, A., & Moyou, M. (2017). The geometry of orthogonal-series, square-root density estimators: Applications in computer vision and model selection. In Computational Information Geometry (pp. 175-215). Springer, Cham.

Pinheiro, A., & Vidakovic, B. (1997). Estimating the square root of a density via compactly supported wavelets. Computational Statistics & Data Analysis, 25(4), 399-415.

Pizer, S. M., & Marron, J. S. (2017). Object statistics on curved manifolds. In Statistical Shape and Deformation Analysis (pp. 137-164). Academic Press.

Ranneby, B., Jammalamadaka, S. R., & Teterukovskiy, A. (2005). The maximum spacing estimation for multivariate observations. Journal of statistical planning and inference, 129(1-2), 427-446.

Rissanen, J. (2000). MDL denoising. IEEE Transactions on Information Theory, 46(7), 2537–2543.

Roos, T., Myllymaki, P., & Rissanen, J. (2009). MDL Denoising Revisited. IEEE Transactions on Signal Processing, 57(9), 3347–3360.

Rosenblatt, M. (1956). Remarks on some nonparametric estimates of a density function. The Annals of Mathematical Statistics, 832-837.

Rudemo, M. (1982). Empirical choice of histograms and kernel density estimators. Scandinavian Journal of Statistics, 65-78.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. Nature, 323(6088), 533–536.

Rzeszotnik, Z. (2001). Calderón's condition and wavelets. Collectanea Mathematica, 52(2), 181-191.

Saeki, S. (1995). On the reproducing formula of Calderón. Journal of Fourier Analysis and Applications, 2(1), 15-28.

Sculley, D., Snoek, J., Wiltschko, A., & Rahimi, A. (2018). Winner's curse? On pace, progress, and empirical rigor. 6th International Conference on Learning Representations, Workshop track.

Seabold, S., & Perktold, J. (2010). Statsmodels: Econometric and statistical modeling with python. In Proceedings of the 9th Python in Science Conference (Vol. 57, p. 61).

Shirazi, E., & Chaubey, Y. P. (2019). Non-negative Density Estimation via Wavelet Block Thresholding for Biased Data. Journal of Statistical Theory and Practice, 13(1), 11. https://doi.org/10.1007/s42519-018-0019-2

Silverman, B. W. (1982). On the estimation of a probability density function by the maximum penalized likelihood method. The Annals of Statistics, 795-810.

Silverman, B. W. (2018). Density estimation for statistics and data analysis. London: Chapman and Hall.

Simonoff, J. S. (2012). Smoothing methods in statistics. Springer Science & Business Media.

Sprent, P., & Smeeton, N. C. (2016). Applied nonparametric statistical methods. CRC press.

Srivastava, A., Jermyn, I., & Joshi, S. (2007, June). Riemannian analysis of probability density functions with applications in vision. In 2007 IEEE Conference on Computer Vision and Pattern Recognition (pp. 1-8). IEEE.

Starck, J. L., Moudden, Y., Abrial, P., & Nguyen, M. (2006). Wavelets, ridgelets and curvelets on the sphere. Astronomy & Astrophysics, 446(3), 1191-1204.

Starck, J. L., Murtagh, F., & Fadili, J. M. (2010). Sparse image and signal processing: wavelets, curvelets, morphological diversity. Cambridge university press.

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. Journal of the Royal Statistical Society: Series B (Methodological), 36(2), 111-133.

Stone, C. J. (1980). Optimal rates of convergence for nonparametric estimators. The annals of Statistics, 1348-1360.

Stone, C. J. (1984). An asymptotically optimal window selection rule for kernel density estimates. The Annals of Statistics, 12(4), 1285-1297.

Strang, G. (1989). Wavelets and dilation equations: a brief introduction. SIAM Review, 31, 614-627.

Strang, G. (1993). Wavelet transforms versus Fourier transforms. Bulletin of the American Mathematical Society, 28(2), 288-305.

Sweldens, W. (1996). The lifting scheme: A custom-design construction of biorthogonal wavelets. Applied and computational harmonic analysis, 3(2), 186-200.

Taubman, D., & Marcellin, M. (2012). JPEG2000 image compression fundamentals, standards and practice: image compression fundamentals, standards and practice (Vol. 642). Springer Science & Business Media.

Tribouley, K. (1995). Practical estimation of multivariate densities using wavelet methods. Statistica Neerlandica, 49(1), 41-62.

Triebel, H. (1992). Theory of Function Spaces II, Birkhäuser Verlag, Basel.

Tukey, J. (1958). Bias and confidence in not quite large samples. Annals of Mathematical Statistics, 29, 614.

Vannucci, M. (1995). Nonparametric density estimation using wavelets. Institute of Statistics & Decision Sciences, Duke University.

Vannucci, M., & Vidakovic, B. (1997). Preventing the Dirac disaster: wavelet based density estimation. Journal of the Italian Statistical Society, 6(2), 145.

Valens, C. (1999). A really friendly guide to wavelets. ed. Clemens Valens.

Vanraes, E., Jansen, M., & Bultheel, A. (2001). Stabilized lifting steps in noise reduction for non-equispaced samples. In Wavelets: Applications in Signal and Image Processing IX (Vol. 4478, pp. 105-116). International Society for Optics and Photonics.

Vidakovic, B. (2009). Statistical modeling by wavelets (vol 503). John Wiley & Sons.

Walter, G. G. (1992). Approximation of the delta function by wavelets. Journal of Approximation Theory, 71(3), 329-343.

Wand, M.P., & Jones, M.C., (1995). Kernel smoothing. Chapman and Hall, London.

Wang, H., & Marron, J. S. (2007). Object oriented data analysis: Sets of trees. The Annals of Statistics, 35(5), 1849-1873.

Weiss, Y. (1999, September). Segmentation using eigenvectors: a unifying view. In Proceedings of the seventh IEEE international conference on computer vision (Vol. 2, pp. 975-982). IEEE.

Weiss, G., & Wilson, E. N. (2001). The mathematical theory of wavelets. In Twentieth century harmonic analysis—a celebration (pp. 329-366). Springer, Dordrecht.

Willett, R. M., & Nowak, R. D. (2007). Multiscale Poisson intensity and density estimation. IEEE Transactions on Information Theory, 53(9), 3171-3187.

Xiao, H., Rasul, K., & Vollgraf, R. (2017). Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747.

Yan, D., Huang, L., & Jordan, M. (2009). Fast approximate spectral clustering. Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 907–916.

Yang, W., Lu, K., Yang, P., & Lin, J. (2019). Critically Examining the "Neural Hype", Weak Baselines and the Additivity of Effectiveness Gains from Neural Ranking Models. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 1129-1132).

Yoon, B. J., & Vaidyanathan, P. P. (2004). A multirate DSP model for estimation of discrete probability density functions. IEEE transactions on signal processing, 53(1), 252-264.

Younes, L. (2010). Shapes and diffeomorphisms (Vol. 171). Berlin: Springer.