SAPIENZA
UNIVERSITÀ DI ROMA

***Sapienza* University of Rome**

Department of Basic and Applied Sciences for Engineering

Doctoral Thesis
XXXIII

Differential Entropy Based Methods for Thresholding in Wavelet Bases and Other Applications

Author: ***Lorenzo Della Cioppa***

Supervisor: ***Vittoria Bruni***
Coordinator: ***Fabio Camilli***

"In mathematics, there are only tricks:
if the trick works twice is a technique,
if it works three times it is a method."

Juris Hartmanis

To my Grandparents

Contents

List of Figures

List of Tables

Acknowledgements

First, I'd like to thank my advisor Prof. Vittoria Bruni and Prof. Domenico Vitulano for having accepted me as their PhD student in the last three years. The lessons I have been able to learn from them are invaluable.

I believe that PhD is a proving experience that can not be reduced to another bachelor or master degree. It is a situation where one constantly lives hand in hand with both uncertainty and wonder. None of this would have been possible without the people whose names are listed here. A special thank goes to my family: my father, Michele, my mother, Delia, and my sister, Anita. They have always been by my side in these years. I also wish to thank my grandparents, Elena and Franco, and my aunt, Arianna, for their thoughts and support. In particular, I wish to thank my aunt Arianna for her precious heater, which spared me several obnoxious frostbites.

In addition to my family, there are a few people I would like to mention, in no particular order.

A special thank goes to Martina Magliocca, PhD. She helped me a lot in the typesetting of this thesis (without her the pages you are reading now would look rather lame) and she supported and encouraged me a lot in these last difficult months.

I really wish to thank Diana (for her support, strong encouragement and gleeful company) and my other dear Lindy Hop friends: Veronica, Elisabetta, Francesco, Caterina, Clara and so on. More generally, I wish to thank the entire Rome Lindy Hop scene and Lindy Hop dancers, as it has been a wonderful new passion I discovered in my PhD years that really helped me.

I wish to thank Michela, which has been more than a colleague, and all the other peoples in SBAI: Caterina, Laura, Alessandro, Elisa, Davide, Davide, Simone, Riccardo and so on. (ci sono due davidi distinti?)

A big thank goes to Alessandro (Crimi) who, without asking any question, has been very supportive in a difficult moment for me.

Another big thank goes to Martina and her wonderful husband, Riccardo. Both have been invaluable to me, and I really hope they realize it (moreover, thank you Riccardo for the car, she's still alive and kicking!).

Along the way, a PhD experience allows meeting several new interesting persons, and some of them remain in your life.

It is the case of my ICVSS friends: Cristiano, Cristina and Marco. I thank they all for their contributions in my life, but I have to specially thank Cristiano for his direct support and his encouragement in the last months.

I have to mention my friends Simone and Anna Chiara who, in light of their experience, never stopped supporting me and giving me precious advices, and for that there are not words to aptly convey my thanks.

I thank my long-time friend Maurizio, for having introduced me to the practice of Uechiryu Karate, which improved my mental and body health in the last months.

I thank Daniele, high school friend and fellow PhD student for his friendship.

Even if our friendship too often works like an accordion, I could never value enough my dear friend Marina. She has always been a source of fun and insightful conversations.

Nothing of this would have been possible without my now-long-time-friends (which my mother refer to as the "neighbourhood friends", even if they do not live close): Ruggero, Giuliano, Luca, Federico, Federico, Nicolò, Jacopo, Alessandro and Paolo. At different times and in different ways, they have all been pivotal for me and for my life.

In the end, I wish to thank Viola, Max, Giorgia and all the people at "Todomodo" library, without that place, this thesis might have been come out worse. The special atmosphere of that place helped me a lot in writing, giving me a glimpse of normality in a spectral summer.

Introduction

Nowadays, lossless and lossy compression are themes of central importance: efficient data transmission and storage is a fundamental problem even for modern computers and broadband connections. Lossless compression has been extensively studied since mid 20-th century [1], and many methods exist, such as Huffman coding (used in jpeg encoding) [2], arithmetic coding [3] or Lempel-Ziv algorithms (used in the zip compressed file format) [4].

Regarding lossy compression, several approaches exist. Most of the times a suitable representation for the uncompressed data is chosen and then reduced using some heuristic, usually related to the specific task. For example, in the jpeg format a real cosine transform is applied to the data and then its coefficients are quantized and discarded according to considerations based on the human perception system [5]. Similarly, in mp3 audio format a modified cosine transform is implemented and the quantization process is constructed around the psychoacoustic effect of frequency masking [6].

However, in a more general context, such particular information about the data might not be available, and hence it might not be possible to formulate any particular ad-hoc strategy.

For this reason, since the 1980's, methods to compactly approximate data have been investigated. Most of the methods belong to the field of function approximation.

Wavelets and wavelet transform have been one of the most important mathematical tools developed to this aim in the last decades [7]. Although defined as a continuous transform, a proper sampling of the wavelet transform provides a representation which is complete and stable in $L^2(\mathbb{R})$; if the sampling is performed at dyadic scales the representation is an orthonormal basis for the Hilbert space $L^2(\mathbb{R})$ endowed with the usual scalar product [8]. Different sampling schemes produce frames instead, which provide stable and invertible, yet redundant, representations. As a consequence, discrete wavelet transform (DWT) is derived, defining an orthogonal, complete, stable and invertible representation in $\mathbb{R}^n$. If the wavelet is chosen accurately, DWT is capable of achieving a good sparsity for a wide class of functions, in the sense that most of the coefficients of the transform are almost zero. Donoho and Johnstone [9, 10] proposed to threshold wavelet coefficients of a function $f(t)$ corrupted by additive gaussian noise in order to recover the original function. If the noise has variance σ^2, they proved that thresholding the wavelet coefficients at $\sigma\sqrt{2\log(n)}$, where n is the length of the discrete signal, is asymptotically optimal – this threshold is commonly known as Donoho universal threshold. Even if no noise is present, thresholding the wavelet coefficients is a viable solution to approximate $f(t)$. This approach has been generalized and formalized by DeVore and is known as

nonlinear approximation [11, 12, 8].

Instead of choosing an orthonormal basis a priori, basis selection aims at constructing the best representation of a function $f(t)$ by selecting elements from a redundant dictionary according to some cost function [13, 8]. Basis pursuit formulates bases selection as a linear programming problem, for which several efficient algorithms exist [8, 14, 15, 16, 17, 18]. Mallat and Zhang proposed a faster, greedy approach to basis pursuit called matching pursuit [19, 20] for which several variations and improvements have been proposed [21, 22, 23, 24, 25, 26, 27, 28].

Another approach to achieve sparsity is through the definition of a suitable convex cost-fidelity functional where, given a fixed representation of a discrete function f (i.e. a basis or a frame), a vector of coefficients x is found such that it balances fidelity to f with some penalization term, which is usually taken as a combination of ℓ^p norms of x, $p \geq 1$, weighted by a regularization parameter λ. Methods like least squares regularization (or Tikonov regularization)[29, 30], LASSO [31, 32] or elastic net regularization [33] belong to this approach. Non-convex functionals, which further promote sparsity, have been proposed as well [34, 35, 36, 37, 38, 39, 40, 41, 42, 24]. Since non-convex functionals are usually non-differentiable as well and usual optimization techniques are not feasible, several dedicated numerical techniques have been proposed [43, 44, 45, 46, 47, 48]. It is worth noting that, if the chosen representation is an orthonormal basis, approximations obtained through thresholding realize the minimum of specific cost-fidelity functionals for some λ, which is related to the threshold value.

A more recent approach to sparse representation is given by compressed sensing, introduced by Donoho, Candés and Tao [49, 50], which aims at finding the sparsest representation of a given discrete signal. Compressed sensing revived interest in non-convex optimization and sparsity in the last years; several theoretical results have been established on the topic and several numerical algorithms have been proposed [51, 52, 53, 54, 55, 56].

The main issue with all the aforementioned methods and approaches is that none of them is truly completely automatic. In thresholding methods the threshold has to be set from the user, basis selection and matching pursuit require a maximum error tolerance or a limit on the number of selected vector; basis pursuit also require the constraints in the penalization term to be set by the user.

The issue persists in the case of regularization functionals, where the choice of the regularization parameter λ is crucial.

Several approaches have been proposed for the estimation of parameters but, apart from ad-hoc techniques, few general methods exist.

One of the most successful is the L-curve method, where the residual approximation error is considered as a function of the regularization parameter λ [57]. The shape of the curve strongly resembles that of the letter "L", i.e. it is composed of a first part that decays rapidly, and then of a second part that decays more shallowly. The optimal point, separating these two regions, is then located as the point maximizing the curvature of the L-curve. In practice, computation of the second derivative is impossible due to numerical error or noise, and methods to approximate the optimal point have been proposed [58].

Another well established method is generalized cross validation (GCV), where the explicit dependency of the approximation on the regularization parameter λ is expressed, and a particular function depending on it is minimized.

Unfortunately, it requires either strong theoretic results about the relation between x and λ, or numerical computation of x.

The contribution of this thesis

In this thesis, novel information-based methods for the automatic selection of expansion coefficients in Hilbert space are proposed. Considering the curve of the monotonically decreasing rearranged absolute values of the expansion coefficients, the proposed methods aim at finding the optimal point that separates important coefficients from superfluous ones. Indeed, following the L-curve approach [57, 58], it is possible to separate the rearranged coefficients curve into two portions: the first one, rapidly decaying, is composed of the "more representative" coefficients, the second one, which decays slowly, is composed of the "less representative" coefficients. The proposed approach is to assume each of those sets of coefficients emitted from a different stochastic source, and a separation point is searched by looking at their difference in terms of information contribution. In order to do so, three different novel information measures have been proposed. The first one is inspired by Vitanyi's normalized compression distance (NCD). While in NCD a compressor C is used to measure the distance between two strings a and b – by comparing the size of their compressed concatenation $C(ab)$ against $C(a)$ and $C(b)$ – in the proposed measure the compressor is replaced by properly weighted differential entropy. For this reason the measure has been named ENID (entropic NID). For each pair of rearranged coefficients sets, ENID is computed and the separation point between sources is individuated as the one realizing the global minimum of ENID.

In order to formalize ENID, differential entropy of a function f has been defined by defining a correspondence between functions and random variables, following the approach of Kolmogorov [59]. More recently, a similar approach has been used in [60]. As a result, differential entropy of a function $f(t)$ can be expressed in time-domain in terms of its monotonically decreasing rearrangement $f^*(t)$: if $f(t) : \Omega \to \mathbb{R}$ is a measurable limited function and E_f is its differential entropy, then

$$E_f = \frac{1}{|\Omega|} \int_0^{|\Omega|} \ln(|\Omega||f^{*\prime}(t)|)dt. \tag{I.1}$$

The above representation formula also allows to design a consistent numerical scheme to approximate E_f in time-domain. Moreover, it holds that differential entropy of a properly normalized function $f(t)$ is always negative and is equal to 0 if and only if $f(t) = t$. As a result, differential entropy is a complexity measure that takes into account the difference of $f(t)$ from the identity.

Pushing the investigation one step forward in the same direction, Kullback-Leibler divergence $D_{KL}(f||g)$ has been considered as direct asymmetric complexity measure between two functions $f(t)$ and $g(t)$.

Similarly to eq. (I.1), it is possible to write Kullback-Leibler divergence between f and g in terms of their monotonically decreasing rearrangements. Moreover, a stronger relation is proved in this thesis: given two monotone functions $f, g : [0, 1] \to [0, 1]$ then the Kullback-Leibler divergence of f from g is equal to the negative of the differential entropy of the composition of g^{-1} with f:

$$D_{KL}(f||g) = -E_{g^{-1}\circ f}. \tag{I.2}$$

As a result, two novel information measures are defined as different symmetrizations of $D_{KL}(f||g)$: KLID (Kullback-Leibler information distance) is defined as the maximum between $D_{KL}(f||g)$ and $D_{KL}(g||f)$; JID (Jeffrey information distance) is defined as the sum of $D_{KL}(f||g)$ and $D_{KL}(g||f)$, which is also known as Jeffrey divergence.

The relation expressed in eq. (I.2) allows one to give approximations of KLID and JID in terms of differential entropies and in terms of averages of f and g, along with conditions on f and g such that the approximations are valid. Hence it is possible to compute KLID and JID without explicitly computing inverse functions.

As in the case of ENID, coefficients selection for both KLID and JID is performed computing the chosen information measure for each pair of rearranged coefficients sets; the separation point between sources is individuated by its global minimum.

The performances of the proposed methods have been investigated through several numerical experiments. Different wavelet bases and frames have been considered, and comparisons with other methods and other a priori selection criteria have been presented.

In particular, is has been noted that the behaviours of both ENID, KLID and JID qualitatively match approximation error measured through NCD: critical points of the proposed measures mark change of trend in NCD approximation error.

Moreover, KLID and JID are robust to noise, and their performances under noisy conditions are in line with state of the art methods and close to the theoretical optimum.

Apart from wavelet basis, an application to shape descriptors in Fourier basis [61] is presented.

As a last contribution, the problem of locating modes interferences in multicomponent chirp-like signals is addressed [62, 63, 64]. A novel time-domain method is proposed, employing KLID and JID to automatically select the optimal smoothing kernel for an energy-based signal. Interferences are then located as extrema of this smoothed energy signal. The proposed interference locating method proved capable of automatically locating interference regions in both synthetic data and simulated micro-doppler data [65, 66].

Plan of the thesis

The thesis is structured as follows.

Chapter 1 provides the reader the necessary mathematical tools and with an overview of the state of the art methods in approximation theory and regularization.

In Chapter 2, differential entropy and Kullback-Leibler divergence of deterministic functions are formally defined, and eqs. (I.1) and (I.2) are proved, along with some other technical results. The numerical scheme to compute differential entropy is defined and its consistency is proved. To further validate the advantages of the proposed approach, a new proof of Pinkser inequality with a tighter constant is given in the case of two random variables attaining the same range of values.

Chapter 3 is the core of this thesis, as it is devoted to ENID, KLID and JID. In the first part, ENID is introduced and its connection with NCD is illustrated. The meaning of differential entropy as a complexity measure is explained and the existence of a global minimum is proved. Numerical experiments in several wavelet bases and frames are presented. KLID and JID are presented in the second part and the rationale behind their definition is explained, detailing a connection with Kolmogorov complexity. The proposed approximations are derived and validity conditions are established. Numerical experiments are presented, in both noiseless and noisy conditions. In particular, in noisy conditions the results are compared with state of the art denoising methods and with the optimum. To conclude the chapter, an application of the proposed measures to shape descriptors is proposed.

In Chapter 4, the problem of locating chirp interferences is presented, detailing the proposed interference locating method and the role that KLID and JID play in it. Numerical evidence is provided, proving that it performs well with both synthetic data and simulated micro-doppler data.

Resumé of Activities

The main research work I've been conducted during the three years of PhD program concerned the study of novel information measures for the optimal expansion coefficients set problem. In this period, I worked with and have been supervised by Prof. Vittoria Bruni (SBAI) and Prof. Domenico Vitulano (SBAI).

The first year of PhD program has been devoted to the study of the field and to the investigation of the relation between information quantities (such as differential entropy) and other properties of functions, such as regularity. During this period an initial versions of the work proposed in this thesis has been presented at EUSIPCO 2018, applied to the problem of Fourier shape descriptors, and at MASCOT 2018.

In my second year, the general case of wavelet basis has been addressed. ENID has been defined and the results have been published. In parallel, I started to work on the problem of modes interference in multi component signals. The attendance to the conference ISPA 2019 greatly improved my knowledge about the topic.

Moreover, during the first two years I attended several courses offered by department SBAI, expanding my mathematical preparation to more modern topics.

Last year has been devoted to the definition of KLID and JID, and to the related theoretical and numerical studies. The research about modes interference continued and resulted in the definition of a time-domain method for interference detection. This result has been presented at ICIAR 2020.

In addition, from the second year of PhD program, along with fellow PhD student Michela Tartaglione, I started working with Prof. Francesca Pitolli (SBAI) and Annalisa Pascarella, PhD (IAC - CNR) on the extension of RAMSET random sampling inversion method for MEG to the EEG case. The results of the work should have been presented at SIMAI 2020, which has been postponed to 2021 due to the COVID-19 pandemic.

Courses

- **Nonlinear diffusion in inhomogenous environments**

  - Prof. Anatoli Tedeev (Institute of Appl. Mathematics and Mechanics, Donetsk, Ukraine)

- **An introduction to the functions of bounded variation and existence results for elliptic equations with p-laplacian principal part (p greater or equal to 1) and singular lower order terms**

  - Prof. Virginia De Cicco and Prof. Daniela Giachetti (SBAI - Sapienza)

- **Master M4 Metodi Numerici per l'Ingegneria Biomedica** (Corso di Laurea in Ingegneria Biomedica)

  - Prof. Francesca Pitolli (SBAI - Sapienza)

- **Lectures on Mean Field Games**

  - Prof. Fabio Camilli (SBAI - Sapienza)

Schools

- **Summer school "Mathematics in Imaging Science"** [35 hrs]
  University of Bologna, 28/05/2018 - 01/06/2018
- **Summer school "ICVSS – International Computer Vision Summer School"** [30 hrs]
  University of Catania, 08-14/07/2018

Conferences

- **Eusipco 2018, Rome, 3-7/09/2018** – Speaker (V. Bruni, L. Della Cioppa, D. Vitulano, "An entropy-based approach for shape description")
- **MASCOT 2018, Rome, 2-5/10/2018** – Speaker (V. Bruni, L. Della Cioppa, D. Vitulano, "A normalized information based method for efficient signal representation")
- **ISPA 2019, Dubrovnik (Croatia), 23-25/09/2019**
- **ICIAR 2020, online** – Speaker (V. Bruni, L. Della Cioppa, D. Vitulano, "A Multiscale Energy-Based Time-Domain Approach for Interference Detection in Non-stationary Signals")

Publications

- V. Bruni, L. Della Cioppa, D. Vitulano, "An entropy-based approach for shape description", *Proceedings of Eusipco 2018*, 2018
- V. Bruni, L. Della Cioppa, D. Vitulano, "A normalized information based method for efficient signal representation", *Proceedings of MASCOT 2018*, 2018
- V. Bruni, L. Della Cioppa, D. Vitulano, "An automatic and parameter-free information-based method for sparse representation in wavelet basis", *Mathematics and Computers in Simulation* , vol. 176 2020
- V. Bruni, L. Della Cioppa, D. Vitulano, "A Multiscale Energy-Based Time-Domain Approach for Interference Detection in Non-stationary Signals", In: *Campilho A., Karray F., Wang Z. (eds) Image Analysis and Recognition. ICIAR 2020. Lecture Notes in Computer Science*, vol 12132, 2020
- V. Bruni, L. Della Cioppa, D. Vitulano, "An information-based method for shape representation and classification", in preparation

Chapter 1

Approximation and sparsity: preliminaries and mathematical tools

Contents

Let $\mathbb{H}$ be a real Hilbert space endowed with inner product $\langle \cdot, \cdot \rangle$ and let $\mathcal{B} = \{\psi_n\}_{n \in \mathbb{N}}$ be a basis of $\mathbb{H}$. Every function $f \in \mathbb{H}$ can be written (up to density) as a linear combination of elements of the basis, i.e. there exists a sequence $f = \{f_n\}_{n \in \mathbb{N}} \in \ell^2$ such that

$$f = \sum_{n \in \mathbb{N}} f_n \psi_n.$$

If the basis vector are pairwise orthogonal (i.e. $\langle \psi_i, \psi_j \rangle \neq 0$ if and only if $i = j$), then the generic coefficient f_n is the projection of f on the n-th element of the basis via the inner product of $\mathbb{H}$:

$$f_n = \langle f, \psi_n \rangle .$$

We say that $\tilde{f}_I$ is an **approximation** of f if it is the linear combination of elements of the basis $\mathcal{B}$ indexed by $I \subset \mathbb{N}$:

$$\tilde{f}_I = \sum_{n \in I} f_n \psi_n,$$

It is trivial that the approximation $\tilde{f}_I$ of f gets better the larger the set I is, but in several applications compactness of representation is a desirable property. Hence it is of great interest the problem of finding the best approximation $\tilde{f}_I$ of f using only M elements of the basis $\mathcal{B}$. This problem is called **sparsity problem**.

More formally, let $\mathbb{H}$ be finite dimensional, let $\mathcal{B} = \{\psi_j\}_{j\in\mathbb{N}}$ be a basis of $\mathbb{H}$ and let $\Psi \in \mathbb{R}^{N\times N}$ be the matrix whose columns are the elements of $\mathcal{B}$. Define the 0-norm of a vector $v \in \mathbb{R}^N$ as the number of its non zero entries:

$$\|v\|_0 = |\{i \ : \ v_i \neq 0\}|,$$

then

Problem 1.0.1 (Sparsity problem)**.** *Let $\mathbb{H}$ of dimension N. Given a function $f \in \mathbb{H}$, a frame $\mathcal{B}$ of $\mathbb{H}$ and its matrix $\Psi \in \mathbb{R}^{N\times N}$ and an integer M, find a vector $x^* \in \mathbb{R}^N$ such that*

$$x^* = \arg\min_{x\in\mathbb{R}^N} \|f - \Psi x\| \qquad \|x\|_0 \leq M.$$

A similar statement holds true if $\mathbb{H}$ is infinite dimensional, with Ψ a suitable synthesis operator and $x \in \ell^2$.

Let us remark that the constraint $\|x\|_0 \leq M$ is equivalent to $\|x\|_0 = M$, since once we know how to solve one problem, then we are able to solve the other one in a finite number of steps.

The sparsity problem belongs to the class of NP-hard optimization problems [67, 68]. This implies that is not known whether a polynomial time algorithm exists or the problem is impossible to be solved in polynomial time. In this chapter, several approaches to provide an approximate solution to sparsity problems are reviewed.

1.1 Bases and frames

A brief introduction to the mathematical tools that will be used in this thesis will now be given. For a more complete discussion, the reader may refer to [8].

In a generic Hilbert space $\mathbb{H}$, it is possible to define an analogous of the basis of a finite dimensional space.

Definition 1.1.1 (Hilbert basis)**.** *Let $\mathbb{H}$ be a Hilbert space. A set $\mathcal{B} = \{\psi_i\}_{i\in I} \subset \mathbb{H}$, $I \subseteq \mathbb{N}$, is said to be an **orthonormal basis** for the Hilbert space if*

1. $\langle \psi_i, \psi_j \rangle = 0$ *if and only if* $i = j$;
2. $\|\psi_i\| = 1$, $\forall i \in I$;
3. $\mathrm{Span}(\mathcal{B})$ *is dense in* $\mathbb{H}$.

If the space $\mathbb{H}$ is separable, then it admits a countable orthonormal basis and it is possible to uniquely write each $f \in \mathbb{H}$ as

$$f = \sum_{i \in \mathbb{N}} \langle f, \psi_i \rangle \, \psi_i.$$

Furthermore, the Plancherel identity holds:

$$\|f\|^2 = \sum_{i \in \mathbb{N}} |\, \langle f, \psi_i \rangle \,|^2.$$

1.1.0.1 Riesz bases

In many applications the requirements for an orthogonal basis are too strong. If the elements of $\mathcal{B}$ are not orthogonal, but they are linearly independent instead, a Riesz basis is obtained.

Definition 1.1.2 (Riesz basis)**.** *Let $\mathbb{H}$ be a Hilbert space. A set $\mathcal{B} = \{\psi_i\}_{i \in I} \subset \mathbb{H}$, $I \subseteq \mathbb{N}$, is said to be a **Riesz basis** for the Hilbert space if*

1. *$\mathcal{B}$ is linear independent;*
2. *$\|\psi_i\| = 1$, $\forall i \in I$;*
3. *for every $f \in \mathbb{H}$ there exist two costants $A > 0$ and $B > 0$ and a sequence $\lambda = \{\lambda_i\}_{i \in \mathbb{N}}$ such that*

$$f = \sum_{i \in I} \lambda_i \psi_i$$

and

$$\frac{1}{B} \|f\|^2 \leq \sum_{i \in I} |\lambda_i|^2 \leq \frac{1}{A} \|f\|^2 .$$

The absence of the orthogonality condition on $\mathcal{B}$ implies that, in general, $\lambda_i \neq \langle f, \psi_i \rangle$. However, if $\mathbb{H} = L^2(\mathbb{R})$, the Riesz-Fréchet representation theorem shows that, for each element λ_i of the sequence λ, there exists a vector $\tilde{\psi}_i$ such that $\lambda_i = \langle f, \tilde{\psi}_i \rangle$ and, as a consequence,

$$f = \sum_{i \in I} \langle f, \tilde{\psi}_i \rangle \psi_i.$$

The set $\tilde{\mathcal{B}} = \{\tilde{\psi}_i\}_{i \in \mathbb{N}}$ is also a Riesz basis of $\mathbb{H}$ and it is possible to prove that

$$f = \sum_{i \in I} \langle f, \psi_i \rangle \tilde{\psi}_i.$$

Since

$$\langle \psi_i, \tilde{\psi}_j \rangle = \delta_{ij}, \qquad \forall i, j \in \mathbb{N}$$

the bases $\mathcal{B}$ and $\tilde{\mathcal{B}}$ are said to be **biorthogonal bases**.

1.1.0.2 Frames

It is possible to further weaken the requirements on the set $\mathcal{B}$, by dropping the linear independence. In this case the set $\mathcal{B}$ is called **frame**.

Definition 1.1.3 (Frame)**.** *Let $\mathbb{H}$ be a Hilbert space. A set $\mathcal{B} = \{\psi_i\}_{i\in I} \subset \mathbb{H}$, $I \subseteq \mathbb{N}$, is said to be a **frame** of the Hilbert space if there exist two constants $A > 0$ and $B > 0$ such that for any $f \in \mathbb{H}$*

$$A \|f\|^2 \leq \sum_{i\in I} |\langle f, \psi_i \rangle|^2 \leq B \|f\|^2 .$$

If $A = B$ the frame is said to be tight. A frame is an orthonormal basis if and only if $A = B = 1$.

Similarly to a Riesz basis, a frame allows for a function $f \in \mathbb{H}$ to be reconstructed from its frame coefficients $\langle f, \psi_i \rangle$, but not necessarily on the same set of vectors. Let us define the **frame operator** U as the operator that assigns f to its coefficients:

$$(Uf)_i = \langle f, \psi_i \rangle .$$

It is possible to prove that, if $\mathcal{B}$ is linearly dependent, U is invertible on its image ImU and it admits an infinite number of left inverses. In this case, the pseudo inverse $\tilde{U}^{-1}$ is defined as the left inverse that is zero on Im$U^{\perp}$.

The pseudo inverse can be written as follows,

$$\tilde{U}^{-1} = (U^*U)^{-1}U^*,$$

where U^* is the adjoint operator of U.

The following theorem proves that it is possible to reconstruct $f \in \mathbb{H}$ as a linear combination of a dual frame.

Theorem 1.1.4. *Let $\mathcal{B} = \{\psi_i\}_{i\in I}$ be a frame with bounds A and B. The dual frame is defined by*

$$\tilde{\psi}_i = (U^*U)^{-1}\psi_i$$

and, for any $f \in \mathbb{H}$, it satisfies

$$\frac{1}{B} \|f\|^2 \leq \sum_{i\in I} |\langle f, \tilde{\psi}_i \rangle|^2 \leq \frac{1}{A} \|f\|^2$$

and

$$\begin{aligned} f &= \tilde{U}^{-1}Uf \\ &= \sum_{i\in I} \langle f, \psi_i \rangle \tilde{\psi}_i \\ &= \sum_{i\in I} \langle f, \tilde{\psi}_i \rangle \psi_i. \end{aligned}$$

If the frame $\mathcal{B}$ is tight, then

$$\tilde{\psi}_i = \frac{1}{A}\psi_i$$

and

$$f = \frac{1}{A}\sum_{i\in I}\langle f, \psi_i\rangle\psi_i.$$

The frame operator is also called **Analysis Operator** and the pseudo inverse operator is also called **Synthesis Operator**.

Efficient and general algorithms exist to invert the frame operator but, in most cases, ad-hoc methods are used, such as for Gabor frames or wavelet frames.

Several different Hilbert basis have been proposed, depending on the context. A brief summary will be given about Fourier basis, wavelet bases, Gabor bases and Riesz bases, that have been implemented in several real world applications, such as voice communication [6, 19], lossy image compression [69, 70, 71], audio processing [72], signal and image denoising [73, 74, 75, 9] and pattern recognition [76]. A more deeper discussion may be found in [8] and in [77].

In the rest of this thesis the space of real square Lebesgue-integrable functions $L^2(\mathbb{R})$ or the euclidian space of dimension N $\mathbb{R}^N$ will be mostly considered.

1.2 Transforms

Integral transforms are a fundamental tool in mathematics. In signal processing, integral transforms, and their discrete counterparts, are commonly used for a wide range of applications, from analysis to compression to approximation and extrapolation [8]. Some of the most commonly used integral transforms are closely related to bases and frames.

1.2.1 Fourier transform

Fourier series were originally introduced by Joseph Fourier as a tool to study heat diffusion problems on bounded domains. In one dimension, a function is decomposed on an orthogonal basis of sine and cosine functions, whose frequencies are integer multiples of the dimension of the domain interval.

In particular, let $f(t) \in L^2([0, L])$, and let $a = \{a_n\}_{n\in\mathbb{N}}$ and $b = \{b_n\}_{n\in\mathbb{N}}$ two sequences such that

$$a_n = \frac{2}{L}\int_0^L f(t)\cos(2\pi nt/L)dt$$

and

$$b_n = \frac{2}{L}\int_0^L f(t)\sin(2\pi nt/L)dt.$$

It is easy to prove that $\mathcal{B} = \{\cos(2\pi nt/L), \sin(2\pi nt/L)\}_{n\in\mathbb{N}}$ is an orthogonal system, meaning that its elements are pairwise orthonormal with respect to the inner product of $L^2([0, L])$. It is less trivial but not too difficult to prove that, for every $f \in L^2([0, L])$, the series

$$\sum_{n\in\mathbb{N}} a_n \cos(2\pi nt/L) + b_n \sin(2\pi nt/L)$$

is convergent. This proves that $\mathcal{B}$ is an orthogonal basis for $L^2([0, L])$.

Fourier transform generalizes the approach of Fourier series for functions defined on the entire real line.

Definition 1.2.1 (Fourier transform)**.** *Let $f \in L^1(\mathbb{R})$. The Fourier transform $\hat{f}$ of f is a function $\hat{f}(\xi) : \mathbb{R} \to \mathbb{C}$ defined as*

$$\hat{f}(\xi) = \int_{\mathbb{R}} f(t)e^{i\xi t}dt.$$

It is easy to prove that Fourier transform is well defined for $L^1(\mathbb{R})$ functions. Indeed, for all $\xi \in \mathbb{R}$,

$$\begin{aligned} |\hat{f}(\xi)| &\leq \left| \int_{\mathbb{R}} f(t)e^{i\xi t}dt \right| \\ &\leq \int_{\mathbb{R}} |f(t)|dt \\ &< +\infty. \end{aligned}$$

In the case of Fourier series, f is decomposed on an orthogonal basis, hence the decomposition is invertible. Since Fourier transform does not project on an orthogonal basis, the previous results do not apply here. Nevertheless, the following theorem proves that, provided it is in $L^1(\mathbb{R})$, Fourier transform is invertible.

Theorem 1.2.2 (Inverse Fourier transform [8])**.** *Let $f \in L^1(\mathbb{R})$ and $\hat{f} \in L^1(\mathbb{R})$. Then*

$$f(t) = \frac{1}{2\pi} \int_{\mathbb{R}} \hat{f}(\xi)e^{it\xi}d\xi.$$

1.2.1.1 Linear time-invariant filtering

From a physical point of view, Fourier series decompose a function on the natural modes of its domain. In the one dimensional case this is particularly evident as the basis is composed of trigonometric functions. Similarly, $\hat{f}(\xi)$ may be thought as the projection of f on the complex sine wave of frequency ξ. This intuition is self-evident if acoustic signals are taken into account. Manipulation of a signal affecting its frequency content is commonly referred to as filtering. In particular, if the filtering is obtained as a convolution product of f with a kernel h it is called linear time-invariant filtering. Let $f * h$ denote the convolution product between two functions f and h and let us recall the convolution theorem.

Theorem 1.2.3 (Convolution theorem [8]). *Let $f \in L^1(\mathbb{R})$ and $h \in L^1(\mathbb{R})$. Then $g = h * f \in L^1(\mathbb{R})$ and*

$$\hat{g}(\xi) = \hat{h}(\xi)\hat{f}(\xi).$$

Convolution theorem says that the convolution product of a signal f with a signal h results in a frequency dependent weighting of $\hat{f}$. The signal h is called the **impulse response** of the filter and its transform $\hat{h}$ is called the **frequency response** of the filter.

1.2.1.2 Extension to $L^2(\mathbb{R})$

Recall that $L^2(\mathbb{R})$ is an Hilbert space, endowed with the scalar product

$$\langle f, g \rangle = \int_{\mathbb{R}} f(t)g^*(t)dt,$$

where $g^*(t)$ is the complex conjugate of $g(t)$, and with the resulting norm

$$\begin{aligned} \|f\| &= \langle f, f \rangle^{\frac{1}{2}} \\ &= \left(\int_{\mathbb{R}} |f(t)|^2 dt \right)^{\frac{1}{2}}. \end{aligned}$$

Since the inner product will be used for the Fourier transform as well, it is defined as an hermitian product if a complex function is considered.

If the Fourier transform is restricted to the intersection of $L^2(\mathbb{R})$ and $L^1(\mathbb{R})$, then it preserves inner product and norm, up to a proportionality constant.

Theorem 1.2.4 (Parseval identity [8]). *Let f and h in $L^1(\mathbb{R}) \cap L^2(\mathbb{R})$. Then*

$$\int_{\mathbb{R}} f(t)h^*(t)dt = \frac{1}{2\pi} \int_{\mathbb{R}} \hat{f}(\xi)\hat{h}^*(\xi)d\xi.$$

Corollary 1.2.5 (Plancharel identity [8]). *Let f and h in $L^1(\mathbb{R}) \cap L^2(\mathbb{R})$. Then*

$$\int_{\mathbb{R}} |f(t)|^2 dt = \frac{1}{2\pi} \int_{\mathbb{R}} |\hat{f}(\xi)|^2 d\xi.$$

Since $L^1(\mathbb{R}) \cap L^2(\mathbb{R})$ is dense in $L^2(\mathbb{R})$, using theorems 1.2.4 and 1.2.5 it is possible to extend the definition of Fourier transform to the whole $L^2(\mathbb{R})$ space.

1.2.1.3 Regularity

A remarkable property of Fourier transform is that the global regularity of a function f depends on the decay of its Fourier transform $\hat{f}$, as stated in the following proposition.

Proposition 1.2.6. *Let $f \in L^2(\mathbb{R})$. If*

$$\int_{\mathbb{R}} |\hat{f}(\xi)|(1 + |\xi|^p)d\xi < +\infty$$

then f is bounded and p times continuously differentiable with bounded derivatives.

Using the frequency interpretation, the less high frequencies a function has, the more it is regular. If $\hat{f}$ is compactly supported, then $f \in C^\infty(\mathbb{R})$ and it is called a **bandlimited function**.

This last proposition allows to deduce global regularity of f from its Fourier transform $\hat{f}$, but nothing can be said about its local regularity. Global regularity means worst regularity, hence if a function is discontinuous in one point but infinitely differentiable everywhere else, its Fourier transform still decays like ξ^{-1}, giving no clue that most of the function is regular. As it will be shown, wavelet transform will overcome this issue.

1.2.1.4 DTFT, DFT and FFT

Fourier transform is well defined for continuous functions, but in many applications there is the need for a discrete equivalent, which is provided by the **discrete time Fourier transform (DTFT)**.

Definition 1.2.7 (DTFT)**.** *Let $f = \{f_j\}_{j\in\mathbb{Z}} \subset \mathbb{R}$. The DTFT of f is defined as*

$$\mathrm{DTFT}_f(\xi) = \sum_{j\in\mathbb{Z}} f_j e^{-i\xi j}.$$

Discrete time Fourier transform will be simply denoted as $\hat{f}(\xi)$, if no risk of confusing it with regular Fourier transform is present.

Inversion of the DTFT is possible by the following reconstruction formula:

$$f_j = \frac{1}{2\pi}\int_0^{2\pi} \mathrm{DTFT}_f(\xi) e^{i\xi j} d\xi.$$

To numerically compute the DTFT a sampling of the frequencies is required. The most common choice is to sample ξ uniformly in the interval $[0, 2\pi]$. In this case it is called the **discrete Fourier transform (DFT)**.

Definition 1.2.8 (DFT)**.** *Let $f \in \mathbb{R}^N$. The DFT of f is defined as*

$$\mathrm{DFT}_f(k) = \sum_{j=0}^{N-1} f_j e^{-i2\pi\frac{k}{N}j}.$$

Similarly to the DTFT, the DFT is invertible as well. The inverse is provided by the following formula:

$$f_j = \frac{1}{N}\sum_{k=0}^{N-1} \mathrm{DFT}_f(k) e^{i2\pi\frac{k}{N}j}.$$

By precalculating the coefficients $e^{i2\pi\frac{k}{N}j}$, DFT can be computed in $\Theta(N^2)$ operations.

In practice, a complexity of $\Theta(N^2)$ can easily be too large, but there exists a category of algorithms that compute DFT in $O(N\log(N))$. These algorithms are collectively known as **fast Fourier transform (FFT)** algorithms. Albeit the first known application of a FFT algorithm is due to C. F. Gauss, the first contemporary FFT algorithm has been published in 1965 and it is known as Cooley-Tukey algorithm [78]. For further details, the reader may refer to [79].

1.2.1.5 STFT

Despite its many good properties, Fourier transform (and hence its discrete counterpart DTFT), being a global operator does not yield any local information. To overcome the lack of time localization, in 1946 Gabor introduced a windowed version of Fourier transform, known as **short time Fourier transform (STFT)**.

Definition 1.2.9 (STFT)**.** *Let $f \in L^2(\mathbb{R})$ and let $g \in L^2(\mathbb{R})$ such that $\|g\|_2 = 1$. The short time Fourier transform is defined as*

$$Sf(u,\xi) = \int_{\mathbb{R}} f(t)g(t-u)e^{-i\xi t}dt.$$

If g is chosen properly (usually with compact support or energy concentrated around $t = 0$), the STFT performs the Fourier transform of the signal f only in the neighbourhood of u, thus providing frequency information for that portion of signal only. Time and frequency resolutions of the STFT depend on the window function g: time resolution depends on the spread of g, frequency resolution depends on the spread of $\hat{g}$.

Unfortunately, it is not possible to achieve arbitrary resolution in both time and frequency. If the window function g is narrow, resolution in space is high but resolution in frequency is low; conversely, if g is wide, resolution in frequency is high while resolution in time is low.

The following theorem proves that STFT is invertible and, as for Fourier transform, the energy is preserved. A proof can be found in [8].

Theorem 1.2.10. *Let $f \in L^2(\mathbb{R})$. Then*

$$f(t) = \frac{1}{2\pi}\int_{\mathbb{R}}\int_{\mathbb{R}} Sf(u,\xi)g(t-u)e^{i\xi t}d\xi dt$$

and

$$\int_{\mathbb{R}} |f(t)|^2 dt = \frac{1}{2\pi}\frac{1}{2\pi}\int_{\mathbb{R}}\int_{\mathbb{R}} |Sf(u,\xi)|^2 d\xi dt.$$

If $g(t)$ is a normalized gaussian function, then the collection $g_{u,\xi}(t) = g(t-u)e^{-i\xi t}$, properly sampled, is a frame for $L^2(\mathbb{R})$ called the Gabor frame.

1.2.2 Wavelet transform

Unlike Fourier transform, which analyses a function f in the frequency domain, wavelet transform analyses a function in scale domain. Fourier transform of a function f at frequency ξ may be thought as the projection of f on the complex sinusoid $e^{-i\xi t}$ using the scalar product of $L^2(\mathbb{R})$ — since $e^{-i\xi t} \notin L^2(\mathbb{R})$, the previous result holds in the sense of distributions. Wavelet transform, on the contrary, does not project on a sinusoid, but on dilated and translated copies of a function whose energy is concentrated around $t = 0$ called wavelet.

Definition 1.2.11 (Wavelet)**.** *A function $\psi \in L^2(\mathbb{R})$ is said to be a* ***wavelet*** *if*

$$\int_{\mathbb{R}} \psi(t)dt = 0,$$

$$\begin{aligned}\|\psi\|_2^2 &= \int_{\mathbb{R}} |\psi(t)|^2 dt \\ &= 1,\end{aligned}$$

and its energy is mostly concentrated around $t = 0$.

Definition 1.2.12 (Wavelet transform)**.** *Given a function $f \in L^2(\mathbb{R})$ and a complex wavelet ψ, the* ***continuous wavelet transform*** *at time u and scale s is defined as*

$$Wf(u,s) = \frac{1}{\sqrt{s}} \int_{\mathbb{R}} f(t)\psi^* \left(\frac{t-u}{s}\right) dt,$$

where ψ^* is the complex conjugate of ψ.

The wavelet transform can be rewritten as a convolution:

$$Wf(u,s) = f * \bar{\psi}_s(u),$$

where

$$\bar{\psi}_s(t) = \frac{1}{\sqrt{s}} \psi^* \left(\frac{-t}{s}\right).$$

Hence the wavelet transform can be understood as a linear filtering of f using a filter with frequency response $\widehat{\bar{\psi}_s}(\xi) = \sqrt{s}\widehat{\psi}^*(\xi)$. Since $\int_{\mathbb{R}} \psi(t)dt = 0$, the filter is a band-pass filter and the wavelet transform is the result of the filtering of f using dilated band-pass filters.

The wavelet ψ can be a real function as well as a complex one. In this thesis, we focus on real wavelet transform. More details about wavelet transform using complex wavelet can be found in [8].

Similarly to STFT, wavelet transform is invertible and preserves the energy of f.

Theorem 1.2.13 (Calderón, Grossmann, Morlet)**.** *Let $\psi \in L^2(\mathbb{R})$ be a real function such that*

$$C_\psi = \int_0^{+\infty} \frac{|\hat{\psi}(\xi)|^2}{\xi} d\xi < +\infty.$$

Then, for any $f \in L^2(\mathbb{R})$,

$$f(t) = \frac{1}{C_\psi} \int_0^{+\infty} \int_{\mathbb{R}} Wf(u,s) \frac{1}{\sqrt{s}} \psi \left(\frac{t-u}{s}\right) du \frac{ds}{s^2},$$

and

$$\int_{\mathbb{R}} |f(t)|^2 dt = \frac{1}{C_\psi} \int_0^{+\infty} \int_{\mathbb{R}} |Wf(u,s)|^2 du \frac{ds}{s^2}$$

A proof can be found in [8].

1.2.3 Regularity measurements

As seen in Subsection 1.2.1.3, Fourier transform of f is related to its regularity, but only in a global sense. On the contrary, wavelet transform measures local regularity.

Definition 1.2.14 (Lipschitz regularity)**.** *Let $f \in L^2(\mathbb{R})$. Then*

1. *f is pointwise Lipschitz γ at v if there exist a constant $\alpha > 0$ and a polynomial $p_v(t)$ of degree $m = \lfloor \gamma \rfloor$ such that, $\forall t \in \mathbb{R}$*

$$|f(t) - p_v(t)| \leq \alpha |t - v|^\gamma. \tag{1.1}$$

2. *f is uniformly Lipschitz γ over $[a,b]$ if it satisfies eq. (1.1) for all $v \in [a,b]$, with a constant α that is independent of v.*

3. *The Lipschitz regularity of f at v over $[a,b]$ is the supremum of the γ such that f is Lipschitz γ.*

Lipschitz regularity extends the concept of regularity understood as the number of times f is differentiable at a point, meaning that if f is n times differentiable in v, then f is at least n-Lipschitz regular in v. If $0 < \gamma < 1$, then the degree of $p_v(t)$ is zero, hence it is a constant. In this case Lispchitz regularity is more commonly known as Hölder regularity.

Wavelet transform is related to Lipschitz regularity, under a condition on the number of **vanishing moments** of the wavelet ψ.

Definition 1.2.15. *A wavelet ψ is said to have n vanishing moments if, for every k such that $0 \leq k < n$,*

$$\int_{\mathbb{R}} t^k \psi(t) dt = 0.$$

The following theorem has been proved by Jaffard [80] and it relates pointwise Lipschitz regularity of a function with its wavelet transform.

Theorem 1.2.16 (Jaffard)**.** *Let ψ be a wavelet with n vanishing moments and let $f \in L^2(\mathbb{R})$, f Lipschitz $\gamma \leq n$ at v. Then, there exists a constant A such that, for every $u \in \mathbb{R}$ and every $s \in \mathbb{R}^+$,*

$$|Wf(u,s)| \leq A s^{\gamma+1/2} \left(1 + \left|\frac{u-v}{s}\right|^\gamma\right).$$

Conversely, if $\gamma < n$ is not an integer and there exist A and $\gamma' < \gamma$ such that, for every $u \in \mathbb{R}$ and every $s \in \mathbb{R}^+$,

$$|Wf(u,s)| \leq As^{\gamma+1/2}\left(1 + \left|\frac{u-v}{s}\right|^{\gamma'}\right)$$

then f is Lipschitz γ in v.

In [81], Mallat and Hwang used this relation to estimate the local regularity of a function through its wavelet transform.

1.2.3.1 Multiresolution approximations

Similarly to DFT, it is possible to define a discrete equivalent of the wavelet transform.

Definition 1.2.17. *Given a parameter a and a wavelet ψ, the **discrete wavelet transform** (DWT) of $f \in L^2(\mathbb{R})$ is a sequence $\{d_{j,n}\}_{(j,n)\in\mathbb{Z}^2}$ defined as*

$$\{d_{j,n}\}_{(j,n)\in\mathbb{Z}^2} = \frac{1}{\sqrt{a^j}}\int_{\mathbb{R}} f(t)\psi\left(\frac{t-a^j n}{a^j}\right)dt.$$

If $a = 2$ then the DWT is said to be dyadic.

Definition 1.2.17 defines DWT as the sampling across times and across scales of the continuous time wavelet transform, but a better understanding of the DWT, and a numerically efficient and stable method to compute it, demands the introduction of **multiresolution approximations**.

Loosely speaking, multiresolution approximations are the formalization of the intuitive fact that, since wavelet transform analyses f at different scales, those scales are not completely apart one another. Indeed, a relation exists between adjacent sampled scales. This implications allow one to define an efficient procedure to compute one scale from another.

Definition 1.2.18 (Multiresolutions)**.** *A sequence $\{V_j\}_{jin\mathbb{Z}}$ of closed subspaces of $L^2(\mathbb{R})$ is a **multiresolution analysis** if the following six properties are satisfied:*

1. $$f(t) \in V_j \iff f(t-2^j k) \in V_j, \qquad \forall (j,k) \in \mathbb{Z}^2;$$

2. $$V_{j+1} \subset V_j, \qquad \forall j \in \mathbb{Z};$$

3. $$f(t) \in V_j \iff f(t/2) \in V_{j+1}, \qquad \forall (j,k) \in \mathbb{Z}^2;$$

4.

$$\lim_{j\to+\infty} V_j = \bigcap_{j=-\infty}^{+\infty} V_j = \{0\};$$

5.

$$\lim_{j\to-\infty} V_j = \text{Closure}\left(\bigcup_{j=-\infty}^{+\infty} V_j\right) = L^2(\mathbb{R});$$

6. There exists $\theta \in L^2(\mathbb{R})$ such that $\{\theta(t-n)\}_{n\in\mathbb{Z}}$ is a Riesz basis of V_0.

More details concerning multiresolution approximations can be found in [8]; here we will just give some basic notions that are needed to construct the discrete wavelet transform.

By using the existence of a Riesz basis in the space V_0 and combining properties (1) and (3), it is possible to verify that $\{2^{-j/2}\theta(2^{-j}t-n)\}_{j\in\mathbb{Z}}$ is indeed a Riesz basis of V_j, whose constants A_j and B_j are bounded uniformly with respect to the scales: there exist $A > 0$ and $B > 0$ such that $B_j > B$ and $A_j < A$, for all $j \in \mathbb{Z}$.

We omit the details, but it is possible to construct, for each $j \in \mathbb{Z}$, an orthonormal basis $\{\phi_{j,n}\}_{n\in\mathbb{Z}}$ for the space V_j by dilating and rescaling a single function ϕ called scaling function. The elements of the basis of V_j are defined as

$$\phi_{j,n}(t) = \frac{1}{\sqrt{2^j}}\phi\left(\frac{t-n}{2^j}\right) \qquad n \in \mathbb{Z}.$$

Since $\{\phi_{j,n}\}_{n\in\mathbb{Z}}$ is an orthonormal basis, the projection $P_{V_j}f$ of f onto the space V_j is written as

$$P_{V_j}f = \sum_{n\in\mathbb{Z}} \langle f, \phi_{j,n}\rangle\, \phi_{j,n}.$$

As a result, the coefficients $a_{j,n} = \langle f, \phi_{j,n}\rangle$ can be rewritten as a convolution product:

$$\begin{aligned} a_{j,n} &= \langle f, \phi_{j,n}\rangle \\ &= \int_{\mathbb{R}} \frac{1}{\sqrt{2^j}}\phi\left(\frac{t-2^j n}{2^j}\right) dt \\ &= f * \bar{\phi}_j(2^j n), \end{aligned}$$

with $\bar{\phi}_j(t) = \sqrt{2^{-j}}\phi(2^{-j}t)$. It is straightforward to see that, for each j, $\bar{\phi}_j$ is the impulse response of a low pass filter.

Since $V_1 \subset V_0$, using property (3) it is possible to write

$$\frac{1}{\sqrt{2}}\phi\left(\frac{t}{2}\right) = \sum_{n\in\mathbb{Z}} g_n \phi(t-n), \tag{1.2}$$

with

$$g_n = \left\langle \frac{1}{\sqrt{2}}\phi\left(\frac{t}{2}\right), \phi(t-n) \right\rangle.$$

Hence $g = \{g_n\}_{n\in\mathbb{Z}}$ is the impulse response of a discrete-time filter that relates V_j to V_{j+1}.

Taking the DTFT of both sides of eq. (1.2), we get

$$\hat{\phi}(2\xi) = \hat{g}(\xi)\hat{\phi}(\xi).$$

Using this form, it is clear that it is not said whether such an g_n does exist.

It is indeed true that g_n exists for any scaling function, and also it satisfies a property known as **conjugate mirror filter** property

Definition 1.2.19. *A discrete time filter with impulse response $g = \{g_n\}_{n\in\mathbb{Z}}$ and frequency response $\hat{g}(\xi) = \mathrm{DTFT}_g(\xi)$ is a conjugate mirror filter if*

$$|\hat{g}(\xi)|^2 + |\hat{g}(\xi+\pi)|^2 = 2.$$

The converse is true if the conjugate mirror filter property holds for g_n and if $\inf_{\xi\in[-\pi/2,\pi/2]} |\hat{g}(\xi)| > 0$. Further details and a proof can be found in [8].

To see how multiresolution approximations relate with wavelet transform, consider the orthogonal complement W_j of V_j in V_{j-1}:

$$V_{j-1} = V_j \oplus W_j.$$

The projection of f onto V_{j-1} can be written as the sum of the projections onto V_j and onto W_j,

$$P_{V_{j-1}} f = P_{V_j} f + P_{W_j} f.$$

The following theorem and the subsequent lemma prove that a basis of W_j can be constructed by dilating and translating a proper wavelet function.

Theorem 1.2.20 (Mallat, Meyer)**.** *Let ϕ be a scaling function and g the corresponding conjugate mirror filter. Let ψ be the function whose Fourier transform si*

$$\hat{\psi}(\xi) = \frac{1}{\sqrt{2}}\hat{h}\left(\frac{\xi}{2}\right)\hat{\phi}\left(\frac{\xi}{2}\right),$$

with

$$\hat{h}(\xi) = e^{-i\xi}\hat{g}^*(\xi+\pi).$$

Let us denote

$$\psi_{j,n}(t) = \frac{1}{\sqrt{2^j}} \psi\left(\frac{t - 2^j n}{2^j}\right).$$

For any scale 2^j, $\{\psi_{j,n}\}_{n\in\mathbb{Z}}$ is an orthonormal basis of W_j. For all scales, $\{\psi_{j,n}\}_{(j,n)\in\mathbb{Z}^2}$ is an orthonormal basis of $L^2(\mathbb{R})$.

Lemma 1.2.21. *The family $\{\psi_{j,n}\}_{n\in\mathbb{Z}}$ is an orthonormal basis of W_j if and only if*

$$|\hat{g}(\xi)|^2 + |\hat{g}(\xi + \pi)|^2 = 2$$

and

$$\hat{g}^*(\xi)\hat{h}(\xi) + \hat{g}^*(\xi + \pi)\hat{h}(\xi + \pi) = 0.$$

Theorem 1.2.20 says that, if the scales are sampled dyadically, consecutive scales are obtained by linear filtering. It also implies that the DWT is invertible, since any $f \in L^2(\mathbb{R})$ can be written as

$$f(t) = \sum_{(n,j)\in\mathbb{Z}^2} \langle f, \psi_{j,n}\rangle \, \psi_{j,n}(t).$$

1.2.4 Fast DWT

In order to describe an efficient computation scheme for the discrete wavelet transform, let us define, respectively, the detail coefficients and the approximation coefficients as

$$d_{j,n} = \langle f, \psi_{j,n}\rangle$$

and

$$a_{j,n} = \langle f, \phi_{j,n}\rangle .$$

It is possible to compute the approximation $\{a_{j,n}\}_{n\in\mathbb{Z}}$ and the detail coefficients $\{d_{j,n}\}_{n\in\mathbb{Z}}$ at dyadic scale j from the approximation coefficients at the preceding dyadic scale $j-1$ using the discrete filters $\{g_n\}_{n\in\mathbb{Z}}$ and $\{h_n\}_{n\in\mathbb{Z}}$:

$$a_{j,n} = (a_{j-1} * g)_n$$

$$d_{j,n} = (a_{j-1} * h)_n,$$

where a_j denotes the sequence $\{a_{j,n}\}_{n\in\mathbb{Z}}$. Using a cascade algorithm it is possible to compute the entire DWT using only discrete convolution operations.

It is possible to achieve something more. From the theory of critically sampled filterbanks, if the filter impulse responses h and g are compactly supported, it is possible to reduce the number of coefficients by downsampling by a factor of 2 after each convolution, still having perfect reconstruction.

Downsampling is denoted by the symbol $\downarrow$ followed by a number $k \in \mathbb{N}$ indicating the downsampling factor. The opposite of downsampling is upsampling, and it is denoted by $\uparrow$ followed by a number indicating the upsampling factor. Let f_n be a sequence, then

$$(f \downarrow k)_n = f_{kn}.$$

and

$$(f \uparrow k)_n = \begin{cases} f_{n/k} & \text{if } n \equiv 0 \bmod k \\ 0 & \text{otherwise} \end{cases}$$

In order to compute the frequency response of the downsampling operator, define the auxiliary sequence

$$\Delta_n^{(k)} = \sum_{r\in\mathbb{Z}} \delta_{n,rk},$$

where $\delta_{n,m}$ is the Kronecker delta.

The sequence $\Delta_n^{(k)}$ is 1 if n is an integer multiple of k and 0 else. In addition, it is easy to verify that

$$\begin{aligned} \Delta_n^{(k)} &= \sum_{r\in\mathbb{Z}} \delta_{n,rk} \\ &= \frac{1}{k} \sum_{r=1}^{k} e^{i2\pi rn/k}, \end{aligned}$$

Hence,

$$\begin{aligned}
\widehat{(f \downarrow k)}(\xi) &= \sum_{n\in\mathbb{Z}} (f \downarrow k)_n e^{-i\xi n} \\
&= \sum_{n\in\mathbb{Z}} f_{kn} e^{-i\xi n} \\
&= \sum_{n\in k\mathbb{Z}} f_n e^{-i\xi n/k} \\
&= \sum_{n\in\mathbb{Z}} \Delta_n^{(k)} f_n e^{-i\xi n/k} \\
&= \sum_{n\in\mathbb{Z}} \frac{1}{k} \sum_{r=1}^{k} e^{i2\pi rn/k} f_n e^{-i\xi n/k} \\
&= \frac{1}{k} \sum_{r=1}^{k} \sum_{n\in\mathbb{Z}} f_n e^{-i(\xi/k - 2\pi r/k)n} \\
&= \frac{1}{k} \sum_{r=1}^{k} \hat{f}(\xi/k - 2\pi r/k)
\end{aligned}$$

.

With regard to the frequency response of the upsampling operator,

$$\begin{aligned}
\widehat{(f \uparrow k)}(\xi) &= \sum_{n\in\mathbb{Z}} (f \uparrow k)_n e^{-i\xi n} \\
&= \sum_{n\in k\mathbb{Z}} f_{n/k} e^{-i\xi n} \\
&= \sum_{n\in\mathbb{Z}} f_n e^{-i\xi kn} \\
&= \hat{f}(k\xi)
\end{aligned}$$

As a result, the frequency response of the downsampling operator is the arithmetic mean of k shifted and rescaled copies of the DTFT $\hat{f}$ of the sequence f_n.

In the case $k = 2$, the downsampling operation is also called decimation and the frequency response reduces to

$$\widehat{(f \downarrow 2)}(\xi) = \frac{1}{2}\left[\hat{f}(\xi/2) + \hat{f}(\xi/2 + \pi)\right].$$

Define the decimated approximation coefficients and the decimated detail coefficients as

$$\tilde{a}_{j,n} = ((a_{j-1} * g) \downarrow 2)_n$$

$$\tilde{d}_{j,n} = ((a_{j-1} * h) \downarrow 2)_n.$$

As the next proposition proves, it is possible to perfectly reconstruct a_{j-1} from $\tilde{a}_{j,n}$ and $\tilde{d}_{j,n}$.

Proposition 1.2.22. *For every $j \in \mathbb{Z}$, it holds that*

$$a_{j-1,n} = ((\tilde{a}_j \uparrow 2) * g')_n + ((\tilde{d}_j \uparrow 2) * h')_n,$$

where h'_n and g'_n are the reconstruction filters defined by

$$\widehat{h'}(\xi) = -e^{-iw}\hat{g}(\xi + \pi)$$

and

$$\widehat{g'}(\xi) = e^{-iw}\hat{h}(\xi + \pi)$$

A proof of Proposition 1.2.22 can be found in [8].

The decimated cascade filterbank algorithm, also known as **fast discrete orthogonal wavelet transform**, is very efficient as it takes only $O(N)$ number of operations, where N is the length of the input signal and produces $M < \log_2(N)$ detail scales $\{\tilde{d}_{j,n}\}_{n\in\mathbb{Z}}$, $1 \leq j \leq M$ and an approximation scale, each consisting of $N/2^j$ samples. If N is not a power of 2, the signal can be zero-padded. A similar algorithm for bi-orthogonal wavelet families can be found in [8].

1.2.4.1 SWT

A limitation of DWT is that, since the downsampling/upsampling operations are not time-invariant, the transform itself is not, meaning that the DWT of a translated copy of a signal f is not the translation of the DWT of f. Addressing this problem, in [82] the **stationary wavelet transform (SWT)** is defined. SWT is approximately shift-invariant and can be computed efficiently by the so called algorithm-a-trous [8]. The only difference with ordinary DWT is that the signal decimation step is omitted and the filters are upsampled each step instead. The time-shift invariance property is a consequence of the fact that SWT is the averaging of time-shifted DWT [83].

1.3 Linear & non-linear approximations

Let us assume that $\mathbb{H} = L^2(\mathbb{R})$, and rewrite the sparsity problem accordingly.

Problem 1.3.1 (Sparsity problem)**.** *Given a function $f \in L^2(\mathbb{R})$, a basis $\mathcal{B} = \{\psi_m\}_{m\in\mathbb{N}}$ of $L^2(\mathbb{R})$ and an integer M, find a sequence $x^* \in \ell^2$ such that*

$$x^* = \arg\min_{x\in\ell^2} \|f - f_M\|_2 \qquad \|x\|_0 \leq M,$$

where

$$f_M = \sum_{m\in\mathbb{N}} x_m \psi_m.$$

Let $\mathcal{B} = \{\psi_m\}_{m\in\mathbb{N}}$ be an orthonormal basis of the Hilbert space $L^2(\mathbb{R})$. Any $f \in L^2(\mathbb{R})$ can be decomposed as

$$f = \sum_{m\in\mathbb{N}} \langle f, \psi_m \rangle \psi_m.$$

If only the first M inner products $\langle f, \psi_m \rangle$ are considered, the resulting function is a **linear approximation** of f:

$$f_M = \sum_{m=1}^{M} \langle f, \psi_m \rangle \psi_m.$$

Since f_M is the orthogonal projection of f on the subspace V_M spanned by the first M elements of $\mathcal{B}$, the resulting error is

$$\begin{aligned} \epsilon_M^l &= \|f - f_M\| \\ &= \sum_{m=M+1}^{+\infty} |\langle f, \psi_m \rangle|^2. \end{aligned}$$

The error ϵ_M^l goes to zero as M grows toward infinity, but the decay rate of ϵ_M^l is related to the regularity of f, as the following theorem states.

Theorem 1.3.2. *For any $s > 1/2$, there exist two constants $A > 0$ and $B > 0$ such that if $\sum_{m\in\mathbb{N}} |m^{2s}| |\langle f, \psi_m \rangle|^2 < +\infty$ then*

$$A \sum_{m\in\mathbb{N}} |m^{2s}| |\langle f, \psi_m \rangle|^2 \leq \sum_{M\in\mathbb{N}} M^{2s-1} \epsilon_m^l \leq B \sum_{m\in\mathbb{N}} |m^{2s}| |\langle f, \psi_m \rangle|^2$$

and $\epsilon_m^l = o(M^{-2s})$.

If $\mathcal{B}$ is a Fourier basis or a wavelet basis, then the space

$$W_{\mathcal{B},s}(\mathbb{R}) = \left\{ f \in L^2(\mathbb{R}) \ : \ \sum_{m\in\mathbb{N}} |m^{2s}| |\langle f, \psi_m \rangle|^2 < +\infty \right\}$$

is a Sobolev space.

Linear approximation projects f on the first M elements of $\mathcal{B}$, but it is possible to adaptively select the M vectors over which f is projected. This is called **non linear approximation** [11, 8]. If I_M is the set of the indices of the chosen elements of the basis then

$$f_M = \sum_{m\in I_M} \langle f, \psi_m \rangle \psi_m$$

and the approximation error is

$$\epsilon_M = \sum_{m \notin I_M} |\langle f, \psi_m \rangle|^2.$$

To minimize the error, the indices in I_M should refer to the M highest magnitude coefficients. Let $f_m = \langle f, \psi_m \rangle$ and let $\{f_k^*\}_{k \in \mathbb{N}}$ the decreasing rearrangement of the coefficients $\{f_m\}_{m \in \mathbb{N}}$, that is a sequence $f_k^* = f_{m_k}$ such that

$$|f_k^*| \geq |f_{k+1}^*|.$$

Hence the best approximation is

$$f_M = \sum_{k=1}^{M} f_k^* \psi_{m_k}$$

and the relative minimum error is

$$\epsilon_M = \sum_{k=M+1}^{+\infty} |f_k^*|^2.$$

Similarly to the linear case, the decay rate of the error is related to the decay of the rearranged coefficients f_k^*.

Theorem 1.3.3. *Let $s > 1/2$. If there exists a constant $C > 0$ such that $|f_k^*| \leq Ck^{-s}$ then*

$$\epsilon_M \leq \frac{C^2}{2s-1} M^{1-2s}. \tag{1.3}$$

Conversely, if eq. (1.3) holds, then

$$|f_k^*| \leq \left(1 - \frac{1}{2s}\right)^{-s} Ck^{-s}.$$

A proof can be found in [8].

The following theorem relates the decay of ϵ_M to the ℓ^p norm of the coefficients of f:

$$\|f\|_{\mathcal{B},p} = \left(\sum_{m \in \mathbb{N}} |f_m|^p \right)^{1/p}.$$

Theorem 1.3.4. *Let $p < 2$. If $\|f\|_{\mathcal{B},p} < +\infty$ then*

$$|f_k^*| \leq \|f\|_{\mathcal{B},p} \, k^{-1/p}$$

and $\epsilon_M = o(M^{1-2/p})$.

Theorem 1.3.5. *Let $f \in L^2([0,1])$ and suppose f has a finite number of discontinuities and is uniformly Lipschitz γ between these discontinuities. Then*

$$\epsilon_M = O(M^{-2\gamma}).$$

The space

$$B_{\mathcal{B},p} = \left\{ f \in L^2(\mathbb{R}) \ : \ \|f\|_{\mathcal{B},p} < +\infty \right\}$$

is a particular case of Besov space. More in general, Besov spaces are defined as follows

Definition 1.3.6 (Besov spaces)**.** *Let $\mathcal{M} = \{V_J, W_J, W_{J-1}, W_{J-2}, \ldots\}$ be a multiresolution approximation of $L^2([0,1])$ truncated at level J and let $\{\phi_{J,n}\}_{n\in\mathbb{Z}}$ be a basis of V_J and $\{\psi_{j,n}\}_{n\in\mathbb{Z}}$, $j = 0, \ldots, J$ bases of $W_0, \ldots, W_J$ and denote $\psi_{J+1,n} = \phi_{J,n}$.*

*The **Besov space** $B^s_{\beta,\alpha}[0,1]$ is the set of the functions $f \in L^2([0,1])$ such that*

$$\|f\|_{s,\beta,\alpha} = \left(\sum_{j=-\infty}^{J+1} \left[2^{-j(s+1/2-1/\beta)} \left(\sum_{n=0}^{2^{-j}-1} |\langle f, \psi_{j,n} \rangle|^\beta \right)^{1/\beta} \right]^\alpha \right)^{1/\alpha} < +\infty.$$

It has been shown [84, 85] that Besov spaces do not depend on the choice of the particular wavelet basis, as long as $\psi \in C^q(\mathbb{R})$ has $q > s$ vanishing moments. Other characterizations of Besov spaces exist in literature, equivalent to the one given here [12, 86]. For a more in-depth discussion of Besov spaces in the context of function approximation, the reader may refer to [8].

1.4 Basis selection

Approximation error depends on the selected basis. As a result, it would be desirable to select the best basis to represent f among a collection of bases, called **dictionary**.

Definition 1.4.1 (Dictionary)**.** *A dictionary $\mathcal{D}$ is a union of orthonormal bases in finite dimensional space of dimension N:*

$$\mathcal{D} = \bigcup_{\lambda\in\Lambda} \mathcal{B}^\lambda,$$

where $\mathcal{B}^\lambda = \{\psi^\lambda_m\}_{1\le m\le N}$ are orthonormal basis.

A first example of dictionary is the wavelet packet dictionary, introduced in [13]. To build a wavelet packet, consider a DWT where, at each scale, both approximation and detail coefficients are filtered for both approximations and details at the next scale. In this way the wavelet packet tree is generated, each node corresponding to either an approximation or a detail at a certain scale.

Another example is the Gabor dictionary, introduced in [87] and [88], consisting in gaussian functions translated in both time and frequency.

Before searching for the best basis to represent f, it is needed to define *what* a best basis is. Let I_M^λ the set that indexes the M vectors that maximize $|\langle f, \psi_m^\lambda \rangle|$. The approximation error is computed as

$$\begin{aligned}\epsilon^\lambda[M] &= \sum_{m \notin I_m^\lambda} |\langle f, \psi_m^\lambda \rangle|^2 \\ &= \|f\|^2 - \sum_{m \in I_m^\lambda} |\langle f, \psi_m^\lambda \rangle|^2\end{aligned}$$

Definition 1.4.2 (Bases comparison)**.** *Given two bases $\mathcal{B}^{\lambda_1} = \{\psi_m^{\lambda_1}\}_{1 \le m \le N}$ and $\mathcal{B}^{\lambda_2} = \{\psi_m^{\lambda_2}\}_{1 \le m \le N}$, we say that $\mathcal{B}^{\lambda_1}$ is a better basis than $\mathcal{B}^{\lambda_2}$ for approximating f if for all $M \ge 1$*

$$\epsilon_M^{\lambda_1} \le \epsilon_M^{\lambda_2}.$$

This bases comparison defines a partial ordering in $\mathcal{D}$. The following theorem gives a criterion based on **Schur concave cost functions**.

Theorem 1.4.3. *A basis $\mathcal{B}^{\lambda_1}$ is a better basis than $\mathcal{B}^{\lambda_2}$ to approximate f if and only if for all concave functions $\Phi(u)$*

$$\sum_{m=1}^{N} \Phi\left(\frac{|\langle f, \psi_m^{\lambda_1} \rangle|^2}{\|f\|^2}\right) \le \sum_{m=1}^{N} \Phi\left(\frac{|\langle f, \psi_m^{\lambda_2} \rangle|^2}{\|f\|^2}\right).$$

A proof can be found in [8].

Checking all the concave functions is impratical, that is why a single concave function is commonly used.

The cost of approximating f in a basis $\mathcal{B}^\lambda$ is defined by the **Schur concave sum**:

$$C(f, \mathcal{B}^\lambda) = \sum_{m=1}^{N} \Phi\left(\frac{|\langle f, \psi_m^\lambda \rangle|^2}{\|f\|^2}\right).$$

The condition

$$C(f, \mathcal{B}^{\lambda_1}) \le C(f, \mathcal{B}^{\lambda_2})$$

is a necessary but not sufficient for $\mathcal{B}^{\lambda_1}$ to be a better basis than $\mathcal{B}^{\lambda_2}$. Coifman and Wickerhauser [13] find a best basis by minimizing the cost of f among all possible bases in $\mathcal{D}$:

$$C(f, \mathcal{B}^\lambda) = \min_{\lambda' \in \Lambda} C(f, \mathcal{B}^{\lambda'}).$$

The basis is optimal in the sense that no other basis exists better than $\mathcal{B}^\lambda$ but, since the ordering over $\mathcal{D}$ is partial, there may exist bases that are not comparable with the chosen one.

Several other cost functions have been proposed. Among the others, we recall the Shannon entropy function ($\Phi(u) = -u\log(u)$) and the ℓ^p cost function ($\Phi(u) = |u|^{p/2}$, $p < 2$).

Given a cost function, brute force comparison for searching the best basis is often prohibitive due to the large size of dictionaries. If the dictionary has a tree structure, such as wavelet packets, the dynamic programming approach proposed in [13] finds the best basis in $O(N\log_2(N))$ operations by taking advantage of this structure.

1.4.1 Basis pursuit

Let $\mathcal{D} = \{\psi_p\}_{1\le p\le P}$ be a dictionary of $P > N$ redundant vectors, with at least N linearly independent vectors. An approximation of f using M vectors is a linear combination of elements of $\mathcal{D}$:

$$f_M = \sum_{m=1}^{M} \alpha_{p_m}\psi_{p_m}. \tag{1.4}$$

Computing the approximation f_M that minimizes $\|f - f_M\|$ would solve the sparsity Problem 1.0.1.

Basis pursuit searches for a basis that minimizes the following ℓ^1 cost function:

$$C(f,\mathcal{B}) = \frac{1}{\|f\|}\sum_{m=1}^{N}|\alpha_{p_m}|.$$

On the one hand, the ℓ^1 cost function limits the spread of the energy among many vectors, promoting sparsity. On the other hand, it allows to write the problem as a linear programming problem.

A standard linear programming problem can be stated as follows. Given $A \in \mathbb{R}^{L\times N}$, $b \in \mathbb{R}^N$ and a linear cost function $C(x) = c^t x = \sum_{i=1}^{L} c_i x_i$, find $x^* \in \mathbb{R}^L$ such that

$$x^* = \arg\min_{x\in\mathbb{R}^L} C(x) \qquad \text{subject to} \quad Ax = b \quad \text{and} \quad x_i \ge 0, \quad \forall i = 1,\dots,L.$$

To this aim, define the matrix $G \in \mathbb{R}^{P\times N}$ as $G_{pn} = \psi_{p,n}$, $1 \le n \le N$, $1 \le p \le P$ and let $f \in \mathbb{R}^N$ the discrete function to approximate. The first constraint provides

$$G\alpha = f,$$

where $\alpha \in \mathbb{R}^P$ is the vector of all the possible unknown coefficients in eq. (1.4).

Since the linear programming computes a positive solution, let us introduce the so called *slack* variables $u, v \in \mathbb{R}^P$, $u_p \geq 0$, $v_p \geq 0$, $p = 1, \ldots, P$, such that

$$\alpha_p = u_p - v_p.$$

As a result

$$G\alpha = Gu - Gv = f$$

and

$$\sum_{p=1}^{P} |\alpha_p| = \sum_{p=1}^{P} u_p + \sum_{p=1}^{P} v_p.$$

The resulting linear programming problem is of size $L = 2P$ with

$$A = (G, -G), \quad x = \begin{pmatrix} u \\ v \end{pmatrix}, \quad b = f, \quad c = 1.$$

Standard results of linear programming prove that if the matrix A has rank N then x has at most N non-zero coefficients. Since the dictionary $\mathcal{D}$ has N linearly independent vectors, then A has rank N. Several algorithms exist for linear programming problems, such as interior point algorithm [14], Log-barrier method [16, 15], simplex algorithm [17] and ellipsoid algorithm [18].

1.4.2 Matching pursuit

Despite efficient algorithms exist for approximating the solution of linear programming problems, basis pursuit is computationally expensive. Matching pursuit, introduced by Mallat and Zhang [19], employs a greedy local approach to reduce computational cost. Let $\mathcal{D} = \{\psi_p\}_{1 \leq p \leq P}$ be a dictionary of $P > N$ vectors, with $\|\psi_p\| = 1$, and including at least a basis of $\mathbb{R}^N$. Matching pursuit iteratively searches for elements of the dictionary that minimizes the approximation error. Let $R^m f$ the approximation error at the $m-1$-th iteration. The next vector ψ_{p_m} is chosen such that

$$|\langle R^m f, \psi_{p_m} \rangle| \geq A \sup_{1 \leq p \leq P} |\langle R^m f, \psi_p \rangle|,$$

where $A \in (0, 1]$ is a sub-optimalty factor introduced for computational efficiency.

When the iteration stops, the approximation is given by

$$f_M = \sum_{m=1}^{M} \langle R^m f, \psi_{p_m} \rangle \psi_{p_m}$$

The following theorem [8] proves that matching pursuit converges and that the remainder $R^m f$ converges exponentially to zero.

Theorem 1.4.4. *There exists $\lambda > 0$ such that for all $m \geq 0$*

$$\|R^m f\| \leq 2^{-\lambda m} \|f\|.$$

As a consequence,

$$f = \sum_{m \in \mathbb{N}} \langle R^m f, \psi_{p_m} \rangle \psi_{p_m}$$

and

$$\|f\|^2 = \sum_{m \in \mathbb{N}} |\langle R^m f, \psi_{p_m} \rangle|^2$$

1.4.2.1 Orthogonal Matching Pursuit

When matching pursuit searches for the element of the dictionary that maximizes the inner product $\langle R^m f, \psi_{p_m} \rangle$ it might reintroduces directions that have already been included in the approximation of f due to the dictionary redundancy. A Graham-Schmidt orthogonalization procedure has been proposed in [21] and [22] to address this problem. Once ψ_{p_m} has been selected, it is orthogonalized with respect to the previously selected vectors:

$$u_m = \psi_{p_m} - \sum_{p=1}^{m} \frac{\langle \psi_{p_m}, u_p \rangle}{\|u_p\|^2} u_p.$$

Hence the remainder $R^m f$ can be decomposed as

$$R^m f = \frac{\langle R^m f, u_m \rangle}{\|u_m\|^2} u_m + R^{m+1} f.$$

As a result,

$$\begin{aligned} f &= \sum_{m=1}^{k} \frac{\langle R^m f, u_m \rangle}{\|u_m\|^2} u_m + R^k f \\ &= P_{V_k} f + R^k f, \end{aligned}$$

where V_k is the space spanned by $\{u_m\}_{1 \leq m \leq k}$ and $R^k f \perp V_k$.

Orthogonal matching pursuit converges in a finite number of iterations [8], but the Graham-Schmidt orthogonalization must be implemented carefully to avoid numerical instabilities. Orthogonalizing M vectors requires $O(NM^2)$ operations. In case of wavelet packets and Gabor dictionaries M orthogonal matching pursuit operations require $O(MN \log_2(N))$ operations. If M is large,it is often more efficient to implement non-orthogonal matching pursuit, due to the computational overhead of orthogonalization.

Improvements and modifications of matching pursuit have been proposed recently, such as Block Orthogonal Matching Pursuit (BOMP) [23, 24], Compressive Sampling Matching Pursuit (CoSaMP) [25], Generalized Orthogonal Matching Pursuit (gOMP) [26], Nonnegative Orthogonal Matching Pursuit (NOMP) [27] and Constrained Matching Pursuit (CMP) [28].

1.5 Thresholding

Linear approximation considers the first M coefficients while non-linear approximation considers the first M coefficients of greatest magnitude. In the latter case, it is equivalent to set to zero all the coefficients whose magnitude is less than $|f^*_{M+1}|$. This operation is called *hard thresholding*. There are several type of thresholding, in this section a brief review of the main ones will be given.

1.5.1 Wavelet shrinkage and universal threshold

Wavelet shrinkage has been introduced by Donoho and Johnstone in their seminal paper [9]. In order to recover a function corrupted by noise, they propose a technique, dubbed wavelet shrinkage, consisting of a thresholding of coefficients of the expansion of the same function in a wavelet basis.

Donoho and Johnstone propose two shrinkage procedures, called **RiskShrink** and **VisuShrink**, corresponding to hard and soft thresholding respectively. Let $w = (w_1, \dots, w_n)$ the vector of the coefficients of the wavelet transform. Given a threshold λ, the two thresholding operators on the coefficients w of a wavelet transform are

$$\begin{aligned} \eta_H(w_i, \lambda) &= w_i \mathbf{1}_{\{|w_i|>\lambda\}} && \text{(hard thresholding)} \\ \eta_S(w_i, \lambda) &= \operatorname{sign}(w_i)(|w_i| - \lambda)_+ && \text{(soft thresholding).} \end{aligned} \tag{1.5}$$

As it will be shown in Section 1.6, RiskShrink and VisuShrink minimize a proper functional related to a sparsity problem.

The choice of the threshold is not trivial. A minmax procedure has been proposed to compute the optimal thresholds in both the cases of hard and soft threshold, but it has been proved that, in both cases the optimal threshold is $\lambda^*_n = \sigma\sqrt{2\log(n)}$, whenever $n \to +\infty$ and gaussian noise of amplitude σ is considered. λ^*_n is also known as **Donoho universal threshold**.

Considering an oracle providing perfect information about noise strength, it comes out that the best choice is to retain the wavelet coefficients whose amplitude is larger than the noise level, while discarding the others. The expected error, in ℓ^2 norm, between real (w_i) and estimated $(\hat{w}_i)$ wavelet coefficients is

$$\mathbb{E}\left[\|w_i - \hat{w}_i\|^2\right] = \sum_{i=1}^{n} \min\{w_i^2, \sigma^2\}.$$

In addition, for soft thresholding it holds

$$\mathbb{E}\left[\|w_i - \hat{w}_i\|^2\right] \leq (1 + 2\log(n)) \left(\sigma^2 + \sum_{i=1}^{n} \min\{w_i^2, \sigma^2\}\right).$$

Hence, the performances of soft thresholding are within those of an oracle-guided reconstruction by a factor of $1 + 2\log(n)$.

As a matter of fact, a more precise error estimate can be given. If λ^*_n is such that

$$\lambda_n^* = \arg\max \inf_{\lambda} \sup_{\mu} \frac{\rho(\lambda, \mu)}{n^{-1} + \min(\mu^2, 1)},$$

where $\rho = \mathbb{E}\left[\eta_S(Y, \lambda) - \mu\right]^2$, with $Y \sim N(\mu, 1)$ a gaussian random variable, then

$$\mathbb{E}\left[\|w_i - \hat{w}_i\|^2\right] \leq \Lambda_n^* \left(\sigma^2 + \sum_{i=1}^{n} \min\{w_i^2, \sigma^2\}\right),$$

where Λ_n^* only depends on n and $\Lambda_n^* \leq 1 + 2\ln(n)$. In addition,

$$\Lambda_n^* \sim 2\log(n) \quad \text{as} \quad n \to +\infty,$$

and then

$$\lambda_n^* \sim \sqrt{2\log(n)}.$$

The same result holds for hard thresholding with a constant L_n that can not be exactly asymptotically estimated, but $L_n \sim 2\log(n)$ for large n.

1.5.2 SUREshrink

In [89], Donoho and Johnson introduced an improvement to the previous RiskShrink and VisuShrink. Since in this new method the threshold is chosen using SURE (Stein's Unbiased Risk Estimate) estimator [90], the method has been dubbed **SUREshrink**. SURE estimator gives an unbiased estimate of a multivariate normal random variable. Applying Stein's results to the ideal soft thresholding operator, an unbiased estimate of the error is given by

$$\text{SURE}(\lambda, w) = n - 2|\{i : |w_i| \leq \lambda\}| + \sum_{i=1}^{n} \min(w_i^2, \lambda^2). \tag{1.6}$$

Eq. (1.6) can be used to select a threshold λ_n^* as

$$\lambda_n^* = \arg \min_{0 \leq \lambda \leq \sqrt{2\log(n)}} \text{SURE}(\lambda, w).$$

Since threshold selection via SURE does not yield better results if the wavelet representation is very sparse, the authors proposed an hybrid scheme, combining universal threshold and SURE estimation.

1.6 Regularization

The sparsity Problem 1.0.1 is an NP-hard problem, but approximate solutions can be found by *relaxing* the problem, for example using ℓ^1 or ℓ^2 norms.

In case of ℓ^2 norm, the problem is known as the **Ridge Regression problem**.

Problem 1.6.1 (Ridge regression problem)**.** *Let $\mathbb{H}$ of dimension N. Given a vector $f \in \mathbb{H}$, a frame $\mathcal{B}$ of $\mathbb{H}$, its matrix $\Psi \in \mathbb{R}^{N\times M}$ and $n \in \mathbb{N}$, find a vector $x^* \in \mathbb{R}^M$ such that*

$$x^* = \arg\min_{x\in\mathbb{R}^M} \frac{1}{2}\|f - \Psi x\|_2^2 \qquad \|x\|_2^2 \leq n.$$

Due to the explicit constraint on the norm of x, this problem is still difficult to solve, hence it is relaxed again from the *constrained form* to the *penalized form*. The relaxed penalized form, or regularized form, of ridge regression is best known in mathematics as **least square problem** and it can be stated as follows.

Problem 1.6.2 (Least squares problem)**.** *Let $\mathbb{H}$ of dimension N. Given a vector $f \in \mathbb{H}$, a frame $\mathcal{B}$ of $\mathbb{H}$, its matrix $\Psi \in \mathbb{R}^{N\times M}$ and a regularization parameter λ, find a vector $x^* \in \mathbb{R}^M$ such that*

$$x^* = \arg\min_{x\in\mathbb{R}^M} \frac{1}{2}\|f - \Psi x\|_2^2 + \lambda \|x\|_2^2 .$$

Least squares problem is related to ridge regression problem in the sense that it does exist a parameter λ such that Problem 1.6.1 and Problem 1.6.2 have the same solution.

Despite the least squares problem has an explicit solution in terms of Moore-Penrose pesudo inverse and it has been extensively studied [29, 30], the ℓ^2 norm penalization term does not promote sparsity enough.

A more strict relaxation can be found considering ℓ^1 norm instead of ℓ^2: the resulting problem is called **LASSO problem**. It has been introduced by Tibshirani in [31], inspired by Breiman's Nonnegative Garrote [91] and it is widely used in statistics.

Problem 1.6.3 (LASSO)**.** *Let $\mathbb{H}$ of dimension N. Given a vector $f \in \mathbb{H}$, a frame $\mathcal{B}$ of $\mathbb{H}$, its matrix $\Psi \in \mathbb{R}^{N\times M}$ and a regularization parameter λ, find a vector $x \in \mathbb{R}^M$ such that*

$$x^* = \arg\min_{x\in\mathbb{R}^M} \frac{1}{2}\|f - \Psi x\|_2^2 + \lambda \|x\|_1 .$$

The solution of the LASSO problem is unique under the assumption that the columns of Ψ are in general position [32]. Similarly to the least squares problem, it is possible to write LASSO problem in penalized form and prove that, for some λ, it has the same solution of Problem 1.6.3. A good review of LASSO and its generalizations can be found in [92].

The functional associated with LASSO is convex but non-differentiable, hence standard minimization algorithms such as gradient descent can't be used. Alternate direction multiplier method (ADMM), used in conjunction with an augmented Lagrangian framework, is guaranteed to converge. In addition if the matrix Ψ is orthogonal, as it will be shown in next subsection, the LASSO problem admits an explicit solution in terms of the soft thresholding operator.

A generalization of LASSO, that combines LASSO with least squares problem, is the **Elastic Net problem** [33]

Problem 1.6.4 (Elastic net)**.** *Let $\mathbb{H}$ of dimension N. Given a vector $f \in \mathbb{H}$, a frame $\mathcal{B}$ of $\mathbb{H}$, its matrix $\Psi \in \mathbb{R}^{N\times M}$ and regularization parameters λ_1, λ_2, find a vector $x \in \mathbb{R}^M$ such that*

$$x^* = \arg\min_{x\in\mathbb{R}^M} \frac{1}{2}\|f - \Psi x\|_2^2 + \lambda_1 \|x\|_1 + \lambda_2 \|x\|_2^2 .$$

The functional associated with elastic net problem is differentiable and combines least squares approach with the sparsity of the LASSO.

We note that it is possible to write a penalized form relaxation of the sparsity Problem 1.0.1 and that a solution can be written in terms of the hard thresholding operator, but in this case it not true that it always exists a parameter λ such that the two problems have the same solution.

Problem 1.6.5 (Relaxed sparsity problem)**.** *Let $\mathbb{H}$ of dimension N. Given a vector $f \in \mathbb{H}$, a frame $\mathcal{B}$ of $\mathbb{H}$, its matrix $\Psi \in \mathbb{R}^{N\times M}$ and a regularization parameter λ, find a vector $x \in \mathbb{R}^M$ such that*

$$x^* = \arg\min_{x\in\mathbb{R}^M} \frac{1}{2}\|f - \Psi x\|_2^2 + \lambda \|x\|_0 .$$

1.6.1 Non-convex optimization

Another possible approach is to relax the ℓ^0 norm to a ℓ^p semi-norm, $0 < p < 1$ [34, 35]. Similarly to the case of LASSO, the problem is written in relaxed penalised form and the solution is found as the minimum of a non-convex functional.

Problem 1.6.6 (Non-convex relaxation)**.** *Let $\mathbb{H}$ of dimension N. Given a vector $f \in \mathbb{H}$, a frame $\mathcal{B}$ of $\mathbb{H}$, its matrix $\Psi \in \mathbb{R}^{N\times M}$ and a regularization parameter λ, find a vector $x^* \in \mathbb{R}^M$ such that*

$$x^* = \arg\min_{x\in\mathbb{R}^M} \frac{1}{2}\|f - \Psi x\|_2^2 + \lambda\|x\|_p^p. \tag{1.7}$$

The use of ℓ^p semi-norm, $0 < p < 1$ promotes sparsity more than ℓ^1 or ℓ^2 norm, but it is much more hard to deal with,

Except for some particular cases, the minimum (1.7) is computed numerically. Several approaches exist, such as gradient projection algorithms [43], Forward-Backward splitting methods [44, 45], ADMM iterations [46, 93] or ad-hoc iterations [47, 48].

To formalize some elementary results for convex and non convex optimization, it is necessary to introduce the theory of **proximal mapping** [94].

Definition 1.6.7 (Proximal operator)**.** *The proximal mapping operator for a function $g(x) : \mathbb{R}^n \to \mathbb{R}$ is defined as*

$$\mathrm{Prox}_g^\lambda(v) = \arg\min_{x\in\mathbb{R}^n} \left(g(x) + \frac{1}{2\lambda}\|x - v\|_2^2 \right)$$

Even if the tools of proximal calculus allow to simplify calculations, explicit proximal operators exists in very few cases, some of them will be listed in the sequel.

The proximal operator for ℓ^1 norm is the soft thresholding operator, performed component-wise,

$$\text{Prox}^{\lambda}_{\|\cdot\|_1}(v)_i = \text{sign}(v_i)(|v_i| - \lambda)_+.$$

In the case of ℓ^0 norm, the proximal operator does exist, but it is not a function:

$$\text{Prox}^{\lambda}_{\|\cdot\|_0}(v)_i = \begin{cases} v_i & \text{if } |v_i| > \sqrt{2\lambda} \\ \{0, v_i\} & \text{if } |v_i| = \sqrt{2\lambda} \\ 0 & \text{if } |v_i| < \sqrt{2\lambda} \end{cases}.$$

By redefining the value of $\text{Prox}^{\lambda}_{\|\cdot\|_0}(v)$ in $\sqrt{2\lambda}$, the hard thresholding operator is obtained:

$$\text{Prox}^{\lambda}_{\|\cdot\|_0}(v)_i = v_i \mathbf{1}_{\{|v_i| > \sqrt{2\lambda}\}}.$$

In general, proximal operators are not explicitly computable, but numerical methods have been proposed for efficient estimation [95, 96].

In the last years interest in non-convex functionals has been growing, especially for applications in machine learning, and many different penalty terms have been proposed other than ℓ^p norms [36, 37, 38, 39, 40, 41, 42].

Among the many recent proposals, it is worth noting the approach of Huang [24]: generalized Shannon entropy function and generalized Rényi entropy function are proposed as penalty terms and an efficient algorithm for the computation of the optimal solution is given. In addition he shows that sparsity is promoted with the proposed penalty terms in the sense that the solutions of the minimization problem are on the boundaries of the orthants of the euclidean space $\mathbb{R}^N$.

1.6.1.1 Explicit solutions with proximal operators

From the theory of proximal mapping, it is clear that explicit solutions to problems 1.6.3 and 1.6.5 can be stated using soft and hard thresholding operators in case Ψ is orthogonal, i.e. $\mathcal{B}$ is an orthonormal basis. Let us consider the LASSO problem. If Ψ is orthogonal,

$$\begin{aligned} \arg\min_{x\in\mathbb{R}^M} \frac{1}{2}\|f - \Psi x\|_2^2 + \lambda \|x\|_1 &= \arg\min_{x\in\mathbb{R}^M} \frac{1}{2}\|\Psi(\Psi^t f - x)\|_2^2 + \lambda \|x\|_1 \\ &= \arg\min_{x\in\mathbb{R}^M} \frac{1}{2}\|\Psi^t f - x\|_2^2 + \lambda \|x\|_1 \\ &= \text{Prox}^{\lambda}_{\|\cdot\|_1}(\Psi^t f), \end{aligned}$$

and the same result holds for the relaxed constrained sparsity problem.

Also, Donoho's VisuShrink (eq. (1.5)) and RiskShrink (eq. (1.5)) operators solve the following problems

$$x^* = \arg\min_{x\in\mathbb{R}^M} \|f - \Psi x\|_2^2 + 2\lambda\|x\|_1$$

and

$$x^* = \arg\min_{x\in\mathbb{R}^M} \|f - \Psi x\|_2^2 + 2\lambda\|x\|_0$$

where Ψ is the matrix of the discrete wavelet transform. In order to see it, observe that Ψ is orthogonal and then the coefficients of the transform are given by $\Psi^{-1}f$. It turns out that the proof is the same as for the LASSO.

1.6.2 Parameter choice

Solutions of relaxed problems are solution to the actual problem only for some parameter λ. Hence, it is of great importance to correctly estimate the parameter. In classical convex minimization problems many methods have been proposed, such as discrepancy principle [30], but any method is guaranteed to correctly estimate the right parameter. Recently, Mousavi et al. [97] studied the solutions of LASSO as functions of λ, producing an algorithm based on approximate message passing (AMP) that converges to a solution with the same mean square error as LASSO solution with the optimal parameter. A naive approach to the issue is to numerically compute numerical solutions for several choices of the parameters, and then pick the one realizing the minimum error, but such approach is numerically intensive and is not guaranteed to produce the optimum solution.

In many real world applications it is often possible to use ad-hoc choices, such as in video, image and audio coding [5, 6], where thresholds are chosen according to consideration about human perception system. On the other hand, to the best of the author knowledge, only two general methods for optimum selection of parameters exist.

1.6.2.1 L-curve method

In general, one of the proposed method is the L-curve method. Introduced by O'leary and Hansen in [57], the L-curve for a regularization problem is the curve of the size of the regularized solution versus the size of the corresponding residual as function of λ. The actual way to measure the sizes depends on the considered problem: for example in the case of LASSO the size of the regularized solution is the term $\|x\|_1$, while the size of the residual is $2^{-1}\|f - \Psi x\|_2^2$. The L-curve strongly resembles rate/distortion curves used in compression and information theory [98]. Through the study of several examples, the authors observe that the L-curve is divided in two regions: a first region in which the curve is strongly decreasing and almost vertical and a second region, where the curve has a more shallow behaviour. The first region is where the error is dominant, the second region is where the regularization is dominant. The optimal parameter choice is the point separating these two regions. In [57], it is proposed to identify this point as the maximum curvature point but, in real cases, numerical errors do not allow for a correct estimation. Addressing this issue, in [99] Natarajan proposes to search for the optimal point as the intersection of two linear functions, one estimated at the beginning and the other at the end of the L-curve.

1.6.2.2 Generalized Cross Validation

Another method for λ estimation is the generalized cross validation method (GDV), introduced in [100] and in [101]. To illustrate it, let us consider, for example, least squares problem and let $x(\lambda)$ the solution as a function of λ and $\Psi^I(\lambda)$ a matrix such that $x(\lambda) = \Psi^I(\lambda)f$. GCV finds the best parameter as the one minimizing the function

$$G(\lambda) = \frac{\|\Psi x(\lambda) - y\|_2^2}{(\text{trace}(I - \Psi\Psi^I(\lambda)))^2}.$$

Although GCV is widely used in several fields, it requires either the numerical computation of the solution $x(\lambda)$ or theoretical results about the form of the solution with respect to λ.

1.7 Compressed sensing

Introduced simultaneously by Donoho [49] and by Candès and Tao [50] in 2006, compressed sensing aim is to find a sparse representation of a signal x from few observations y, provided that it is sparse in some basis. In the last years the advent of compressed sensing revived interest in sparsity problems and in regularization problems. Many techniques in convex and non-convex optimization have been borrowed by compressed sensing.

The **compressed sensing problem** can be stated as follows.

Problem 1.7.1 (Compressed sensing)**.** *Given a vector of observations $y \in R^M$ and a measurement matrix $\Phi \in \mathbb{R}^{M\times N}$, $M \ll N$, find x^* such that*

$$x^* = \arg\min_{x\in\mathbb{R}^N} \|x\|_0 \qquad \text{subject to} \qquad \Phi x = y. \tag{1.8}$$

The formulation of compressed sensing is inherently discrete, but in several contexts the columns of Φ are elements of a basis or elements of a frame, often coming from a discrete transform (e.g. a stationary wavelet transform or a discrete Fourier transform), meaning that the components of the observation vector y are indeed coefficients of said transform. To better understand it, let the columns of Φ be the elements of the Fourier basis and let $y = (y_1, \dots, y_M)$ the sampling of a real function $f(t)$ at equispaced time steps t_i, i.e. $y_i = f(t_i)$. Solving the compressed sensing problem for this setting means to find the trigonometric polynomial of minimum degree interpolating y.

More generaly, let us consider a discrete transform represented by a matrix $\Psi \in \mathbb{R}^{N\times N}$. Suppose the goal is to achieve sparsity in the transform, i.e. to have few non zero entries in the vector $x = \Psi y$ of the coefficients, for a given vector y. Assuming the transform is invertible, the problem is to find x^* such that

$$x^* = \arg\min_{x\in\mathbb{R}^N} \|x\|_0 \qquad \text{subject to} \qquad \Psi^{-1}x = y.$$

The sparsity problem is, in some sense, dual to the compressed sensing problem, but if the observation matrix Φ is orthogonal (i.e. its columns form an orthonormal basis of $\mathbb{R}^n$), the two problems are equivalent.

It is easy to check that, for an arbitrary matrix Ψ, a solution to the compressed sensing Problem 1.7.1 is a solution for Problem 1.0.1 with a proper choice of parameters, but the converse is not always true.

Interest in compressed sensing has been growing since its introduction [102], and a complete review is over the scopes of this thesis. A brief overview of the basic concepts along with the main results will be given.

1.7.1 Existence results

Existence results for the compressed sensing problem can be stated in term of the so called restricted isometric property.

Definition 1.7.2 (s-sparse vector)**.** *Let $x \in \mathbb{R}^N$. The vector x is said to be $s-$sparse if*

$$\|x\|_0 \leq s.$$

Definition 1.7.3 (Restricted isometry property)**.** *Let $\Phi \in \mathbb{R}^{M\times N}$. For each s, Φ is said to possess the **restricted isometric property** if*

$$(1-\delta_s)\|x\|_2^2 \leq \|\Phi x\|_2^2 \leq (1+\delta_s)\|x\|_2^2 \tag{1.9}$$

for each s-sparse vector x.

The smallest δ_s satisfying eq. (1.9) is called the **restricted isoperimetric constant**.

The existence of solutions for the compressed sensing problem is stated in terms of the restricted isoperimetric constant in the following theorem.

Theorem 1.7.4 (Candès [51])**.** *The compressed sensing problem (1.8) admits a solution provided that $\delta_{2s} < 1$.*

The compressed sensing problem is not easy to solve, in fact, similarly to the sparsity problem, it is an NP-hard combinatorial problem. In order to compute an approximate solution, the problem is relaxed, using less restrictive conditions on the sparsity of x.

The first relaxation of Problem 1.7.1 is the so called **basis pursuit problem**, in which ℓ^1 norm subsitutes ℓ^0 norm:

Problem 1.7.5 (Basis pursuit)**.** *Given a vector of observations $y \in R^M$ and a measurement matrix $\Phi \in \mathbb{R}^{M\times N}$, $M \ll N$, find x^* such that*

$$x^* = \arg\min_{x\in\mathbb{R}^N} \|x\|_1 \qquad \text{subject to} \qquad \Phi x = y. \tag{1.10}$$

A sufficient condition for the solution of the basis pursuit problem to be a solution of the compressed sensing problem is provided by the following theorem.

Theorem 1.7.6 (Candès [52])**.** *Assume that $\delta_{2s} < \sqrt{2}-1$ and let x_s be the s-sparse vector of the s largest entries of x. The solution x^* to the problem (1.10) obeys*

$$\|x^* - x\|_1 \leq C_0\|x - x_s\|_1 \tag{1.11}$$

and

$$\|x^* - x\|_2 \leq C_0 s^{-1/2}\|x - x_s\|_1,$$

for some constant C_0. In particular, if x is s-sparse, the recovery is exact.

In order to clarify, let $\tilde{x}$ be a solution of the compressed sensing problem (which exists since $\sqrt{2}-1<1$). The observation vector y is the same in both problems; hence, in the basis pursuit problem it is possible to consider $y=\Phi\tilde{x}$. Then, since $\tilde{x}$ is $s-sparse$, the estimate (1.11) becomes

$$\begin{aligned}\|x^*-\tilde{x}\|_1 &\leq C_0\|x-\tilde{x}_s\|_1\\ &=0.\end{aligned}$$

proving that the solutions coincide.

Several improvements to the bound on δ_{2s} have been proved; the latest stated by the following theorem.

Theorem 1.7.7 (Mo & Li [53])**.** *The compressed sensing problem* (1.8) *and the basis pursuit problem* (1.10) *uniquely obtain the same s-sparse solution if* $\delta_{2s}\leq 0.4931$.

The basis pursuit is not the only possible relaxation to the compressed sensing problem. Another common approach, useful in the case of noisy measurements, is the **quadratically constrained basis pursuit problem**:

Problem 1.7.8 (Quadratically constrained basis pursuit problem)**.** *Given a vector of possibly noisy observations* $y\in R^M$*, a measurement matrix* $\Phi\in\mathbb{R}^{M\times N}$*,* $M\ll N$*, and a noise amplitude* ε*, find* x^* *such that*

$$x^*=\arg\min_{x\in\mathbb{R}^N}\|x\|_1 \qquad \text{subject to} \quad \|\Phi x-y\|_2\leq\varepsilon. \tag{1.12}$$

Definition 1.7.9 (s-compressible vector)**.** *Let* $x\in\mathbb{R}^N$*,* $p>0$ *and let.*

$$\sigma_s(x)_p=\min_{z\in\mathbb{R}^N}\|x-z\|_p \qquad \text{subject to} \qquad \|z\|_0\leq s.$$

The vector x *is said to be* $s-compressible$ *in norm* ℓ^p *if* $\sigma_s(x)_p$ *is small for some* p*.*

Theorem 1.7.10 (Candès [52], Foucart & Rauhut [103])**.** *Given* $x\in\mathbb{R}^N$ *and a matrix* $\Phi\in\mathbb{R}^{M\times N}$ *such that* $\delta_{2s}\leq 4/\sqrt{41}\approx 0.6246$*, then every solution of* (1.12) *satisfies*

$$\|x^*-x\|_2\leq Cs^{-1/2}\sigma_s(x)_1+D\varepsilon,$$

with $C>0$ *and* $D>0$*.*

In practice, Least Absolute Shrinkage and Selection Operator (LASSO), introduced by Tibshirani in [31] and widely used in statistics, is more efficient for computations [54]. Similarly to the LASSO problem in Section 1.6, the **LASSO problem** for compressed sensing is stated as follows.

Problem 1.7.11 (LASSO)**.** *Given a vector of observations* $y\in R^M$ *and a measurement matrix* $\Phi\in\mathbb{R}^{M\times N}$*,* $M\ll N$*, find* x^* *such that*

$$x^*=\arg\min_{x\in\mathbb{R}^N}\frac{1}{2}\|y-\Phi x\|_2^2+\lambda\|x\|_1.$$

Using a sequence of LASSO estimation, an efficient algorithm for the basis pursuit problem has been proposed [55]. The following theorem proves that LASSO estimator is capable of recovering sparsity patterns.

Theorem 1.7.12 (Wainwright [56]). *Let $y = \Phi x + \varepsilon$, where x is supported on $S \subset \{1, \ldots, n\}$, with $|S| \leq s$ and ε is a zero-mean additive noise. Let Φ_S and Φ_{S^c} the restriction of Φ to, respectively, S and its complement S^c. Assume Φ satisfies*

$$\|\Phi_{S^c}^T \Phi_S (\Phi_S^T \Phi_S)^{-1}\|_\infty < 1,$$

$\lambda_{min}(A_S^T A_S) > M$ and $\max_{j \in S^c} \|\Phi_j\| \leq \sqrt{M}$.

Further, assume that $N = O(e^{M^\gamma})$, $s = O(M^\alpha)$, $\min_{i \in S} x_i > 1/M^{1-\beta} 2$ with $0 < \alpha + \gamma < \beta < 1$. For $\lambda = M^{\frac{1-\delta}{2}}$ such that $\delta \in (\gamma, \beta - \alpha)$, the LASSO recovers the sparsity pattern S with probability $1 - e^{-cM^\delta}$ for a constant $c > 0$.

Non-convex regularization techniques are also used for compressed sensing [102].

Chapter 2

Deterministic Information

Contents

In the following chapter some of the fundamental instruments that will be used in this thesis are introduced. Some information quantities will be correctly defined for functions, whilst they are typically defined on random variables.

In the following the concepts of entropy and Kullback-Leibler divergence of a function will be formally defined, and some classical information theory results will be derived in this new context. A strong connection between differential entropy and Kullback-Leibler divergence will be established in terms of composition of functions, which will allow to consider both these quantities as complexity measures. Moreover, the influence of noise will be investigated.

2.1 Differential entropy

Let us recall the definition of differential entropy for a random variable X [98].

Definition 2.1.1. *Let X be a random variable with associated probability density function (p.d.f.) $p_X(y)$ compactly supported in $S \subset \mathbb{R}$. The **differential entropy** of X is defined as*

$$E_X = -\int_S p_X(y) \ln(p_X(y)) dy. \tag{2.1}$$

Differential entropy differs from Shannon entropy in many ways. Shannon entropy is the average number of bits needed to encode a message sent from a stochastic source p; differential entropy lacks such a precise interpretation. Furthermore, differential entropy might be negative, where Shannon entropy is always positive.

Although the differential entropy defined in Definition 2.1.1 is well-defined for a stochastic source, in the present discussion we consider deterministic signals. The following definition concerns the differential entropy of a function.

Definition 2.1.2. *Let $f : \Omega \to \mathbb{R}$ be a measurable function.*

*The **differential entropy of a function** is defined as the differential entropy of f composed with a random variable U uniformly distributed in Ω:*

$$E_f := E_{f \circ U}.$$

Albeit definition 2.1.2 formally defines differential entropy for any function f, it gives no insight on what it is, or on how it can be computed. Indeed, the following definitions will lead to a convenient and useful representation.

Definition 2.1.3. *Let Ω be a limited subset of $\mathbb{R}$, μ be a positive measure defined on Ω and $f(t) : \Omega \to \mathbb{R}$ be a μ-measurable function. The **determinisitc distribution function** $M_f(y)$ of f is defined as follows:*

$$M_f(y) = \mu(\{t \in \Omega : f(t) > y\}). \tag{2.2}$$

This definition of distribution is akin to those used in measure theory, for example to state Markov inequality or to define the layer-cake representation.

We remark that, despite similar names that might cause confusion, deterministic distribution functions are different objects from cumulative probability distribution functions: the cumulative probability distribution function of a random variable attains values in the range $[0, 1]$ and is non-decreasing, while the distribution function of a function f attains values in $[0, \mu(\Omega)]$ and is non-increasing, making it more similar to a tail distribution.

We will refer to M_f just as distribution function when the context is clear and no confusion is possible.

Despite this differences, from a mathematical point of view these two objects are very similar. In Kolmogorov's interpretation of probability, a random variable can be represented as the composition of a function with a uniform random variable defined on an appropriate event space [104, 59].

Indeed, there is a deep connection between deterministic distribution functions and random variables, as it will be pointed out in the following discussion. A similar approach has been taken in [60].

Let us consider a measure μ, a measurable function $f : \Omega \to \mathbb{R}$ and a uniform random variable U that attains values in Ω and let us consider the probability space $(\Omega, \mathcal{F}, \mathbb{P}_\mu)$, where $\mathcal{F}$ is the Borel σ-algebra and $d\mathbb{P}_\mu = \mu(\Omega)^{-1} d\mu$. Assume p_F to be the probability density function of the random variable $F = f(U)$.

With the preceding assumptions,

$$\begin{aligned}
\frac{1}{\mu(\Omega)} M_f(y) &= \mu(\{t \in \Omega : f(t) > y\}) \\
&= \mathbb{P}_\mu(f(U) > y) \\
&= \mathbb{P}_\mu(F > y) \\
&= 1 - \mathbb{P}_\mu(F < y) \\
&= \int_y^{+\infty} p_F(z) d\mu(z)
\end{aligned}$$

Mathematically, the definition of deterministic distribution function is completely identical to that of cumulative distribution function (apart from a multiplicative constant and different sign).

In light of this discussion, the following definition derives naturally.

Definition 2.1.4. *Let Ω be a limited subset of $\mathbb{R}$, μ be a positive measure defined on Ω and $f(t) : \Omega \to \mathbb{R}$ be a μ-measurable function. The* ***deterministic density function*** *$m_f(y)$ of f is defined as the derivative of $M_f(y)$:*

$$m_f(y) = \frac{d}{dy} M_f(y) \tag{2.3}$$

From the distribution function M_f of f it is possible to define the decreasing rearrangement of f [105, 106, 107, 108], which will be central in our discussion.

Definition 2.1.5. *Let $f(t) : \Omega \to \mathbb{R}$ be a μ-measurable function. The* ***decreasing rearrangement*** *f^* of f is a monotonic decreasing function $f^*(t) : [0, \mu(\Omega)] \to f(\Omega)$ defined as the generalized left-continuous inverse of $M_f(y)$:*

$$\begin{aligned} f^*(t) &= (M_f)^{-1}(t) \\ &= \inf_y \{M_f(y) < t\}. \end{aligned}$$

Decreasing rearrangements are a well-known tool in mathematical analysis [108]. The definition we use has been previously used in [105], where several interesting properties of decreasing rearrangements have been pointed out. Among them, the following are obtained easily from the definitions, and we report them here for their fundamental importance:

$$M_f(y) = M_{f^*}(y)$$

and

$$\int_\Omega f(t) d\mu(t) = \int_0^{\mu(\Omega)} f^*(t) dt.$$

The decreasing rearrangement may be understood as the unique non-increasing function f^* with the same distribution function as f.

In the following we will consider μ as the usual Lebesgue measure on $\mathbb{R}$, and it will be simply denoted as $\mu(\Omega) = |\Omega|$.

The following proposition, as foretold, gives an interesting and useful representation for the differential entropy of a function.

Proposition 2.1.6. *Let $f(t) : \Omega \to \mathbb{R}$ be a measurable function that is non-constant almost everywhere. Then the following identity holds:*

$$E_f = \frac{1}{|\Omega|} \int_0^{|\Omega|} \ln(|\Omega||f^{*\prime}(t)|) dt,$$

where E_f is the differential entropy of f and f^* is the decreasing rearrangement of f.

Proof. From eq. (2.1) and eq. (2.3) and since $(M_f)^{-1} = f^*$, we have

$$\begin{aligned} E_f &= -\int_{f(\Omega)} p_f(y) \ln(p_f(y)) dy \\ &= -\int_{f(\Omega)} \frac{1}{|\Omega|} |m_f(y)| \ln\left(\frac{1}{|\Omega|}|m_f(y)|\right) dy \\ &= -\frac{1}{|\Omega|}\int_{f(\Omega)} \left|\frac{d}{dy}M_f(y)\right| \ln\left(\frac{1}{|\Omega|}\left|\frac{d}{dy}M_f(y)\right|\right) dy \\ &= -\frac{1}{|\Omega|}\int_{f(\Omega)} \left|\frac{d}{dy}(f^*)^{-1}(y)\right| \ln\left(\frac{1}{|\Omega|}\left|\frac{d}{dy}(f^*)^{-1}(y)\right|\right) dy. \end{aligned} \tag{2.4}$$

Since M_f and f^* are monotone functions, then they are differentiable almost everywhere and their Radon-Nikodym derivatives exist. Taking the change of variable $y = f^*(t)$ and observing that the equality

$$\frac{d}{dy}(f^*)^{-1}(y) = \frac{1}{f^{*\prime}((f^*)^{-1}(y))}$$

holds almost everywhere, then

$$\begin{aligned} E_f &= -\frac{1}{|\Omega|}\int_{f(\Omega)} \left|\frac{1}{f^{*\prime}((f^*)^{-1}(y))}\right| \ln\left(\frac{1}{|\Omega|}\left|\frac{1}{f^{*\prime}((f^*)^{-1}(y))}\right|\right) dy \\ &= \frac{1}{|\Omega|}\int_0^{|\Omega|} \left|\frac{1}{f^{*\prime}(t)}\right| \ln\left(\frac{1}{|\Omega|}\left|\frac{1}{f^{*\prime}(t)}\right|\right) f^{*\prime}(t) dt \\ &= \frac{1}{|\Omega|}\int_0^{|\Omega|} \ln\left(|\Omega|\left|f^{*\prime}(t)\right|\right) dt \end{aligned}$$

□

A first important property of differential entropy is that a rescaling of the domain of f does not change differential entropy, while rescaling the values of f results in an additive constant.

Lemma 2.1.7. *Let $f : \Omega = [a, b] \to \mathbb{R}$ a measurable function and let $\tilde{f}(t) = \beta f(\alpha t)$, $\alpha, \beta > 0$. Then*

$$E_{\tilde{f}} = E_f + \ln(\beta).$$

Proof. It suffices to observe that, by definition of rearrangement,

$$(\tilde{f})^*(t) = \beta f^*(\alpha t),$$

and, using the change of variable $\tau = \alpha t$,

$$\begin{aligned} E_{\tilde{f}} &= \frac{\alpha}{|\Omega|}\int_0^{|\Omega|/\alpha} \ln\left(\frac{|\Omega|}{\alpha}\beta\alpha f^{*\prime}(\alpha t)\right) dt \\ &= E_f + \ln(\beta). \end{aligned}$$

□

2.1.1 Some bounds

In this section some bounds for differential entropy will be developed. A straightforward trivial upper bound for differential entropy follows from Jensen's inequality:

$$\begin{aligned}\frac{1}{|\Omega|}\int_0^{|\Omega|} \ln\left(|\Omega|\left|f^{*\prime}(t)\right|\right) dt &\leq \ln\left(-\int_0^{|\Omega|} f^*(t)dt\right) \\ &= \ln(f^*(0) - f^*(|\Omega|)) \\ &= \ln(|f(\Omega)|).\end{aligned} \tag{2.5}$$

This result is in agreement with the classic results in information theory, where entropy is trivially bounded by the logarithm of the size of the alphabet for Shannon entropy or by the logarithm of the size of the support of p_f for differential entropy [98].

The following proposition provides a lower bound for differential entropy, guaranteeing that $E_f > -\infty$ whenever $m_f \in L^2$.

Proposition 2.1.8. *Let $f : \Omega \to \mathbb{R}$ be a measurable function. Then*

$$\begin{aligned}E_f &\geq -\ln\left(\frac{1}{|\Omega|^2}\int_0^{|\Omega|}\frac{1}{|f^{*\prime}(t)|}dt\right) \\ &= -\ln\left(\left\|\frac{1}{|\Omega|}m_f\right\|_2^2\right).\end{aligned}$$

Proof. Lemma 2.1.7 allows us to limit our proof to the case $|\Omega| = 1$. Using Jensen inequality,

$$\begin{aligned}E_f &= \int_0^1 \ln(|f^{*\prime}(t)|)dt \\ &= -\int_0^1 \ln\left(\frac{1}{|f^{*\prime}(t)}\right)dt \\ &\geq -\ln\left(\int_0^1 \frac{1}{|f^{*\prime}(t)|}dt\right).\end{aligned}$$

Using the change of variables $y = f^*(t)$ and observing that $m_f(y) = (f^{*-1})'(y)$ then

$$\begin{aligned}\int_0^1 \ln(|f^{*\prime}(t)|)dt &\geq -\ln\left(\int_0^1 \frac{1}{|f^{*\prime}(t)|}dt\right) \\ &= -\ln\left(\int_0^1 |(f^{*-1})'(y)|^2 dy\right) \\ &= -\ln\left(\int_0^1 |m_f(y)|^2 dy\right).\end{aligned}$$

Rescaling f completes the proof. □

Getting results about differential entropy in terms of the rearrangement is not straightforward. One of the difficulties is that the derivative of the rearrangement appears in the differential entropy, rather than the rearrangement itself.

It is easily proven from the definition of rearrangement that, if f is sufficiently regular, then

$$|f^{*\prime}(t)| = \left(\sum_{t_i \in f^{-1}(f^*(t))} \frac{1}{|f'(t_i)|} \right)^{-1}.$$

This pointwise relation between the derivative of f and the derivative of its rearrangement has been extendend to a global property by Duff in [106, 107]. We rewrite Duff's result in terms of entropy.

Proposition 2.1.9 (Duff)**.** *Let*

$$n_g(y) = \#|g^{-1}(y)|,$$

that is $n_g(y)$ is the number of (discrete) solutions of the equation $g(t) = y$. Let $f : \Omega \to \mathbb{R}$ be a measurable function. Then it holds that

$$\frac{1}{\Omega}\int_\Omega \ln(|f'(t)|)dt - \frac{1}{\Omega}\int_\Omega \ln\left(n_f(f(t))\right) dt \leq E_f - \ln(|\Omega|) \leq \frac{1}{\Omega}\int_\Omega \ln(|f'(t)|)dt.$$

Proposition 2.1.9 shows that the differential entropy of a function is different from the differential entropy of a random variable. Indeed, the following Proposition 2.1.10 can be seen as the discrete version of Proposition 2.1.9, as it relates the differential entropy of a function f with the differential entropy of the restrictions of f on a partition of Ω.

Proposition 2.1.10. *Let $\Omega \subset \mathbb{R}$ bounded and let $\Omega_1, \dots \Omega_n$ be a covering of Ω, i.e. such that $\Omega = \bigcup_{k=1}^n \Omega_k$, $\Omega_i \cap \Omega_j = \emptyset$ if $i \neq j$. Let $f : \Omega \to \mathbb{R}$ be a measurable function, let f_k be the restriction of f to Ω_k and define $\lambda_k = |\Omega_k|/|\Omega|$, $k = 1, \dots, n$. Then*

$$\sum_{k=1}^n \lambda_k E_{f_k} + E_\lambda - \ln(n) \leq E_f \leq \sum_{k=1}^n \lambda_k E_{f_k} + E_\lambda.$$

where

$$E_\lambda = -\sum_{k=1}^n \lambda_k \ln(\lambda_k)$$

is the discrete Shannon entropy of $\lambda = (\lambda_1, \dots \lambda_2)$.

Furthermore, both bounds are tights: the upper bound is attained if f is monotonic, the lower bound is attained if, for any $i \neq j$, $\Omega_i = \Omega_j$ and $f_i \equiv f_j$.

Proof.

From the definition of distribution function in eq. (2.3), it is trivial that

$$M_f(y) = \sum_{k=1}^n M_{f_k}(y)$$

and hence

$$m_f(y) = \sum_{k=1}^{n} m_{f_k}(y).$$

To ease the notation, we will denote

$$m_k(y) = m_{f_k}(y).$$

Upper bound. Since $-\ln(t)$ is a monotonically decreasing, then for any strictly positive finite sequence $\{a_k\}_{k=1,\dots,n}$ and any $i = 1, \dots, n$,

$$-\ln\left(\sum_{k=1}^{n} a_k\right) \leq -\ln(a_i).$$

Hence, from eq. (2.4),

$$\begin{aligned}
E_f &= -\frac{1}{|\Omega|}\int_{f(\Omega)} \sum_{k=1}^{n} |m_k(y)| \ln\left(\frac{1}{|\Omega|}\sum_{h=1}^{n}|m_h(y)|\right) dy \\
&= -\sum_{k=1}^{n}\frac{1}{|\Omega|}\int_{f_k(\Omega_k)} |m_k(y)| \ln\left(\frac{1}{|\Omega|}\sum_{h=1}^{n}|m_h(y)|\right) dy \\
&\leq -\sum_{k=1}^{n}\frac{1}{|\Omega|}\int_{f_k(\Omega_k)} |m_k(y)| \ln\left(\frac{1}{|\Omega|}|m_k(y)|\right) dy \\
&= -\sum_{k=1}^{n}\frac{|\Omega_k|}{|\Omega|}\left[\frac{1}{|\Omega_k|}\int_{f_k(\Omega_k)} |m_k(y)| \ln\left(\frac{1}{|\Omega_k|}|m_k(y)|\right) dy + \ln\left(\frac{|\Omega_k|}{|\Omega|}\right)\right] \\
&= \sum_{k=1}^{n}\lambda_k E_{f_k} - \sum_{k=1}^{n}\lambda_k \ln(\lambda_k).
\end{aligned}$$

In order to prove that the bound is tight, observe that, if f is monotonic, the M_{f_k}'s have disjoint supports and then, for any i,

$$\int_{f(\Omega_i)} \frac{1}{|\Omega|}|m_i(y)| \ln\left(\frac{1}{|\Omega|}\sum_{k=1}^{n}|m_k(y)|\right) dy = \int_{f(\Omega_i)} \frac{1}{|\Omega|}|m_i(y)| \ln\left(\frac{1}{|\Omega|}|m_i(y)|\right) dy.$$

Lower bound. Since the function $-t\ln(t)$ is concave, then

$$
\begin{aligned}
-\frac{1}{|\Omega|}\int_{f(\Omega)}\sum_{k=1}^{n}|m_k(y)|\ln\left(\frac{1}{|\Omega|}\sum_{h=1}^{n}|m_h(y)|\right)dy = \\
&-\frac{1}{|\Omega|}\int_{f(\Omega)} n\frac{1}{n}\sum_{k=1}^{n}|m_k(y)|\ln\left(n\frac{1}{n}\frac{1}{|\Omega|}\sum_{h=1}^{n}|m_h(y)|\right)dy \\
&=-\frac{n}{|\Omega|}\int_{f(\Omega)}\frac{1}{n}\sum_{k=1}^{n}|m_k(y)|\ln\left(\frac{1}{n}\frac{1}{|\Omega|}\sum_{h=1}^{n}|m_h(y)|\right)dy-\ln(n) \\
&\geq -n\frac{1}{n}\sum_{k=1}^{n}\frac{1}{|\Omega|}\int_{f_k(\Omega_k)}|m_k(y)|\ln\left(\frac{1}{|\Omega|}|m_k(y)|\right)dy-\ln(n) \\
&=\sum_{k=1}^{n}\lambda_k E_{f_k}-\sum_{k=1}^{n}\lambda_k\ln(\lambda_k)-\ln(n).
\end{aligned}
$$

To see that the bound is tight observe that, if $\Omega_i = \Omega_j$ and $f_i \equiv f_j$, then

$$
\begin{aligned}
\sum_{k=1}^{n}\frac{m_k(y)}{|\Omega|} &= n\frac{m_1(y)}{|\Omega|} \\
&= \frac{m_1(y)}{|\Omega_1|}.
\end{aligned}
$$

□

In the following corollary we deduce the differential entropy of a mixture of independent random variables.

Corollary 2.1.11. *Let $X_1,\ldots,X_n$ be random variables and $q=(q_1,\ldots,q_n)$ a discrete probability density. Let X be the mixture of the X_i's according to q.*

Then the entropy of X is

$$E_X = \sum_{k=1}^{n} q_i E_{X_i} + E_q,$$

where E_q is the Shannon entropy of q.

Proof. The mixture X is defined as

$$X = X_i \quad \text{with probability } q_i.$$

It is immediate to see that the probability density function of the mixture X is the convex combination of the probability density functions of the X_i's according to p:

$$p_X(y) = \sum_{k=1}^{n} q_k p_{X_k}(y). \tag{2.6}$$

Let $Q_k = \sum_{i=1}^{k} q_i$, let $f_k : [Q_{k-1}, Q_k] \to \mathbb{R}$ be increasing monotonic functions such that $f_k((Q_k - Q_{k-1})U_k) = X_k$, where the U_k's are independent random variables uniformly distributed on $[0,1]$ and let

$$f(t) = \sum_{k=1}^{n} [f_k(t) + a_k] \mathbf{1}_{[Q_{k-1}, Q_k)}(t),$$

where the a_k are chosen so that f is monotonically increasing. Then

$$m_f(y) = \sum_{k=1}^{n} m_{f_k}(y).$$

Since $p_{X_k}(y) = m_{f_k}(y)/q_k$, from eq. (2.6) $p_X(y) = m_f(y)$ and, applying Proposition 2.1.10 to f, we get

$$\begin{aligned} E_X &= E_f \\ &= \sum_{k=1}^{n} p_k E_{f_k} + E_p \\ &= \sum_{k=1}^{n} p_k E_{X_k} + E_p. \end{aligned}$$

□

2.1.2 Numerical scheme

One of the most important advantages of representing differential entropy of f in terms of the rearrangement f^* is that it is possible to design a simple and stable numerical scheme to correctly approximate it.

By Lemma 2.1.7 let us consider, without loss of generality, $f : [0, 1] \to \mathbb{R}$. Then, assuming constant discretization step $h = 1/N$ and $f_j^* = f^*(jh)$, the numerical approximation $E_f^{(N)}$ of E_f is defined as follows

$$\begin{aligned} E_f &\approx E_f^{(N)} \\ &= \frac{1}{N} \sum_{j=0}^{N-1} \ln(N|f_{j+1}^* - f_j^*|). \end{aligned} \tag{2.7}$$

The following proposition proves that the numerical scheme given in eq. (2.7) is convergent as $h \to 0$ and gives an upper bounds for the error.

Proposition 2.1.12. *Let $|f^{*(3)}(t)/f^{*\prime}(t)| \leq L_1$ and $|f^{*\prime\prime}(t)/f^{*\prime}(t)|^2 \leq L_2$. Then the numerical scheme (2.7) is consistent with order 2 as $h \to 0$ and*

$$\left| E_f - E_f^{(N)} \right| \leq \frac{h^2}{8} \left[L_1 + \frac{1}{3} L_2 \right]$$

Proof. In order to study the consistency of the scheme, let us introduce the points $t_j = h(j - 1/2)$ and the function $F(t) = \ln(|f^{*\prime}(t)|)$. Using composite rectangular integration, it follows that there exists $\xi \in [0, 1]$ such that

$$\begin{aligned} E_f &= \int_0^1 F(t)dt \\ &= h\sum_{j=0}^{N-1} F(t_j) + \frac{N}{24}h^3F''(\xi) \\ &= \frac{1}{N}\sum_{j=0}^{N-1} F(t_j) + \frac{1}{24}h^2F''(\xi) \end{aligned} \tag{2.8}$$

Then

$$\begin{aligned} F(t_j) &= \ln(|f^{*\prime}(t_j)|) \\ &= \ln\left(\left|\frac{f^*_{j+1} - f^*_j}{h}\right| + \frac{h^2}{12}f^{*(3)}(\zeta_j)\right) \\ &= \ln\left(\left|\frac{f^*_{j+1} - f^*_j}{h}\right|\right) + R_j(h), \end{aligned} \tag{2.9}$$

where $\zeta_j \in [h(j-1), hj]$ and

$$R_j(h) = \frac{h^2 f^{*(3)}(\zeta_j)/12}{|f^{*\prime}(t_j)| + \theta_j h^2 f^{*(3)}(\zeta_j)/12}, \tag{2.10}$$

with $\theta_j \in [0,1]$.

Combining eqs. (2.8), (2.9) and (2.10) and invoking the discrete mean value theorem, it follows that there exist $\tau, \xi, \zeta, \theta \in [0,1]$ such that the total error is

$$\begin{aligned} \epsilon(h) &= \left|E_f - E_f^{(N)}\right| \\ &= h^2\left[\frac{1}{12f^{*\prime}(\tau)/f^{*(3)}(\zeta) + h^2\theta} + \frac{1}{24}\left(\frac{f^{*(3)}(\xi)}{f^{*\prime}(\xi)} - \left(\frac{f^{*\prime\prime}(\xi)}{f^{*\prime}(\xi)}\right)^2\right)\right] \\ &\leq h^2\left[\frac{1}{12|f^{*\prime}(\tau)/f^{*(3)}(\zeta)|} + \frac{1}{24}\left|\frac{f^{*(3)}(\xi)}{f^{*\prime}(\xi)}\right| + \frac{1}{24}\left|\frac{f^{*\prime\prime}(\xi)}{f^{*\prime}(\xi)}\right|^2\right] \\ &\leq h^2\left[\frac{1}{8}\left|\frac{f^{*(3)}(\xi)}{f^{*\prime}(\xi)}\right| + \frac{1}{24}\left|\frac{f^{*\prime\prime}(\xi)}{f^{*\prime}(\xi)}\right|^2\right] \\ &\leq h^2\left[\frac{1}{8}L_1 + \frac{1}{24}L_2\right] \end{aligned}$$

□

2.2 Kullback-Leibler divergence

Another information quantity of fundamental importance is the Kullback-Leibler divergence. In classical discrete information theory, given two stochastic sources p and q, the Kullback-Leibler divergence of p from q is the excess quantity of bits needed to encode a message sent from p if it is mistakenly assumed distributed as q.

For two random variable X and Y, the Kullback-Leibler divergence is defined as follows [98].

Definition 2.2.1. *Let X be a random variable with associated probability density function $p_X(y)$ and let Y be a random variable with associated probability density function p_Y, both compactly supported in $S \subset \mathbb{R}$. Then the* ***Kullback-Leibler divergence*** *of X from Y is defined as*

$$D_{KL}(X||Y) = -\int_S p_X(y) \ln\left(\frac{p_X(y)}{p_Y(y)}\right) dy.$$

As we did for the entropy, we will define Kullback-Leibler divergence for functions.

Definition 2.2.2. *Let $f : \Omega_1 \to \mathbb{R}$, $g : \Omega_2 \to \mathbb{R}$ measurable functions and let U_1, U_2 be independent uniform random variables attaining values respectively on Ω_1 and Ω_2. The Kullback-Leibler divergence of f from g is defined as the Kullback-Leibler divergence of $f(U_1)$ from $g(U_2)$:*

$$D_{KL}(f||g) := D_{KL}(f(U_1)||g(U_2)).$$

Considering monotonic functions defined from the unit interval to the unit interval, the following lemma establishes a strong connection between Kullback-Leibler divergence and entropy: given such f and g, the Kullback-Leibler divergence of f from g is the entropy of the composition $g^{-1} \circ f$.

Lemma 2.2.3. *Let $f, g : [0,1] \to \mathbb{R}$ be strictly increasing monotone functions such that $f([0,1]) = g([0,1])$ and $g^{-1} \circ f$ is well defined. Then*

$$D_{KL}(f||g) = -E_{g^{-1}\circ f}$$

Proof. Let U_1 and U_2 two independent uniform random variables attaining values in $[0,1]$ and let $F = f(U_1)$ and $G = g(U_2)$.

Since $p_F(y) = m_f(y)$ and $p_G(y) = m_g(y)$, the Kullback-Leibler divergence of f from g is

$$\begin{aligned} D_{KL}(f||g) &= D_{KL}(f(U_1)||g(U_2)) \\ &= -\int_0^1 |m_f(y)| \ln\left(\frac{|m_g(y)|}{|m_f(y)|}\right) dy \\ &= -\int_0^1 \frac{1}{f'(f^{-1}(y))} \ln\left(\frac{f'(f^{-1}(y))}{g'(g^{-1}(y))}\right) dy. \end{aligned}$$

Operating the change of variable $y = f(t)$ then

$$D_{KL}(f||g) = -\int_0^1 \ln\left(\frac{f'(t)}{g'(g^{-1}(f(t)))}\right) dy.$$

On the other hand

$$\begin{aligned}(g^{-1}\circ f)'(t) &= (g^{-1})'(f(t))f'(t)\\ &= \frac{f'(t)}{g'(g^{-1}(f(t)))},\end{aligned}$$

and the thesis follows.

□

The next proposition extends Lemma 2.2.3 to functions defined on any bounded domain and not both monotonic.

Proposition 2.2.4. *Let $f:\Omega_1\to\mathbb{R}$, $g:\Omega_2\to\mathbb{R}$ two measurable functions, g strictly monotone and such that $f(\Omega_1)=g(\Omega_2)$ and $g^{-1}\circ f$ is well defined.*
Then the following equality holds

$$D_{KL}(f||g) = -E_{g^{-1}\circ f} + \ln(|\Omega_2|). \tag{2.11}$$

Proof. For $t\in[0,1]$, define $\tilde{f}(t)=f(|\Omega_1|t)$ and $\tilde{g}=g(|\Omega_2|t)$.
Since the difference between f and $\tilde{f}$ is a rescaling of the domain, then

$$m_{\tilde{f}}(y) = \frac{m_f(y)}{|\Omega_1|},$$

and similarly

$$m_{\tilde{g}}(y) = \frac{m_g(y)}{|\Omega_2|}.$$

Hence, proceeding as in the proof of Lemma 2.2.3,

$$\begin{aligned}D_{KL}(\tilde{f},\tilde{g}) &= D_{KL}(\tilde{f}(U_1),\tilde{g}(U_1))\\ &= -\int_S |m_{\tilde{f}}(y)|\ln\left(\frac{|m_{\tilde{f}}(y)|}{|m_{\tilde{g}}(y)|}\right)dy\\ &= -\int_S \frac{|m_f(y)|}{|\Omega_1|}\ln\left(\frac{|\Omega_2||m_f(y)|}{|\Omega_1||m_g(y)|}\right)dy\\ &= D_{KL}(f||g).\end{aligned}$$

On the other hand, observing that $\tilde{g}^{-1}(\tilde{f}(t)) = |\Omega_2|^{-1}g^{-1}(f(|\Omega_1|t))$ then Lemmas 2.2.3 and 2.1.7 prove that

$$\begin{aligned}D_{KL}(f||g) &= D_{KL}(\tilde{f}||\tilde{g})\\ &= -E_{\tilde{g}^{-1}\circ\tilde{f}}\\ &= -E_{g^{-1}\circ f} + \ln(|\Omega_2|)\end{aligned}$$

□

2.2.1 Pinsker inequality

The connection established in Proposition 2.2.4, although interesting, allows one to approach problems involving the Kullback-Leibler divergence from a new point of view. To better validate this claim, we will prove an improved bound for Pinsker inequality [109, 110, 111] for the particular case that the two random variables involved attain the same values. This new proof uses an argument similar to that of the famous short proof due to Pollard [112].

One of the classic results in information theory is Pinsker inequality [113, 109, 110], which sets a lower bound for Kullback-Leibler divergence in terms of L^1 norm between density functions: let X and Y random variables, with density functions p_X and p_Y, then

$$D_{KL}(X||Y) \geq \frac{1}{2}\|p_X - p_Y\|_1^2.$$

If X and Y attain the same values, it is immediate to rewrite this result in term of rearrangements. Let us consider two random variables X and Y, attaining values in some set $S \subseteq \mathbb{R}$ and with probability density functions p_x and p_y compactly supported in S. Then there exist two monotone functions $f, g : [0,1] \to \mathbb{R}$ and two independent uniform random variables U_1, U_2 such that $X = f(U_1)$, $Y = g(U_2)$ and $p_X(y) = m_f(y)$, $p_Y(y) = m_g(y)$. Then the following equality holds

$$\begin{aligned}
\|p_X - p_Y\|_1 &= \int_S |p_X(y) - q_Y(y)|dy \\
&= \int_S |m_f(y) - m_g(y)|dy \\
&= \int_S \left| \frac{1}{f'(f^{-1}(y))} - \frac{1}{g'(g^{-1}(y))} \right| dy \\
&= \int_0^1 \left| \frac{1}{f'(t)} - \frac{1}{g'(g^{-1}(f(t)))} \right| f'(t)dt \\
&= \int_0^1 \left| 1 - \frac{f'(t)}{g'(g^{-1}(f(t)))} \right| dt. \qquad (2.12)
\end{aligned}$$

Using the definitions given in this chapter, if p_X and p_Y have the same support, it is then possible to improve Pinsker inequality's constant from 1/2 to 1. This improvement is solely due to the fact that the functional approach developed in this thesis simplifies the problem, and allows one to have a deeper insight on the meaning of each passage of the proof. To the best of the authors' knowledge no such improvement exists in literature for the particular case considered here. The problem of the best constant for Pinsker inequality has been studied in [109], where a general method to generate polynomial lower bound of Kullback-Leibler divergence in terms of L^1 norm is given. Even so, the tighter bound given in [109] for the squared L^1 norm is 1/2.

Let us first prove the following simpler result.

Lemma 2.2.5. *Let $h(t) : [0,1] \to \mathbb{R}$ and $\int h(t)dt = 1$.*

Then

$$-\int_0^1 \ln(h(t))dt \geq \left(\int_0^1 |1-h(t)|dt\right)^2. \tag{2.13}$$

Proof.
The proof can be sketched as follows.

1. A lower bound for the logarithm function will be established depending on two constants a and b, detailing the set of viable values for a and b.
2. A first inequality depending on a and b is proved, using Cauchy-Schwartz inequality.
3. It is proved that, taking the supremum over a and b, eq. 2.13 holds.

Step 1 Generalizing the approach in [112], let us consider the following inequality, for some positive constants a and b:

$$\ln(t) > 1 - \frac{1}{t} + a\frac{(t-1)^2}{t(1+bt)}. \tag{2.14}$$

In order to prove that eq. 2.14 holds, define

$$g(t) = 1 - \frac{1}{t} + a\frac{(t-1)^2}{t(1+bt)} - \ln(t).$$

Then

$$g'(t) = -\frac{(t-1)[(bt+1)^2 - a((2b+1)t+1)]}{t^2(bt+1)^2}.$$

Note that $g'(1) = 0$ and $g(1) = 0$. If g attains a unique maximum in $t = 1$, then (2.14) holds. To this end, it suffices that $g'(t) > 0$ if $t < 1$ and $g'(t) < 0$ if $t > 1$. Hence, it suffices that

$$\begin{aligned} p(t) &= (bt+1)^2 - a((2b+1)t+1) \\ &= b^2t^2 + [(1-a)2b - a]t + 1 - a \\ &> 0. \end{aligned}$$

The roots t_1 and t_2 ($t_1 \leq t_2$) of the associated second order equation are

$$t_{1,2} = \frac{-((1-a)2b-a) \pm \sqrt{((1-a)2b-a)^2 - 4b^2(1-a)}}{2b^2}.$$

Since we are interested only in $t > 0$ two cases arise: $p(t) > 0$ if one of two cases happens: if $\Delta = ((1-a)2b-a)^2 - 4b^2(1-a) < 0$ or if $t_2 \leq 0$.
In the first case

$$\begin{aligned}
\Delta < 0 &\iff ((1-a)2b-a)^2 - 4b^2(1-a) < 0 \\
&\iff (1-a)^2 4b^2 + a^2 - 4a(1-a)b - 4b^2(1-a) < 0 \\
&\iff -a(1-a)4b^2 + a^2 - 4a(1-a)b < 0 \\
&\iff a - 4(1-a)[b^2+b] < 0 \\
&\iff b^2 + b - \frac{1}{4}\frac{1}{1-a} < 0 \\
&\iff b \in \left[-\frac{1}{2} - \frac{1}{2}\sqrt{1+\frac{1}{1-a}}, -\frac{1}{2} + \frac{1}{2}\sqrt{1+\frac{1}{1-a}}\right] \\
&\iff 0 < b < -\frac{1}{2} + \frac{1}{2}\sqrt{1+\frac{1}{1-a}}.
\end{aligned}$$

So, if

$$\begin{cases} 0 < a < 1 \\ 0 < b < -\frac{1}{2} + \frac{1}{2}\sqrt{1+\frac{1}{1-a}} \end{cases} \tag{2.15}$$

then inequality 2.14 holds.
In the second case a necessary condition is that

$$-((1-a)2b-a) \leq 0.$$

This is true if and only if

$$b \geq \frac{1}{2}\frac{a}{1-a}.$$

Hence

$$\begin{aligned}
t_2 < 0 &\iff ((1-a)2b-a) \geq \sqrt{((1-a)2b-a)^2 - 4b(1-a)} \\
&\iff ((1-a)2b-a)^2 \geq ((1-a)2b-a)^2 - 4b(1-a) \\
&\iff 4b(1-a) > 0,
\end{aligned}$$

which is always true if $a < 1$.
Since it is necessary that $\Delta \geq 0$, and

$$\Delta \geq 0 \iff b \geq \frac{-1+\sqrt{1+\frac{1}{1-a}}}{2},$$

it is necessary to verify which bound on b is tighter. The two curves $b_1(a) = a/(2(1-a))$ and $b_2(a) = -(1/2) + (1/2)\sqrt{1+1/(1-a)}$ intersect in the point $a = 3/2 - \sqrt{5}/2$ and

$$b_1(a) > b_2(a) \iff a > \frac{3-\sqrt{5}}{2}.$$

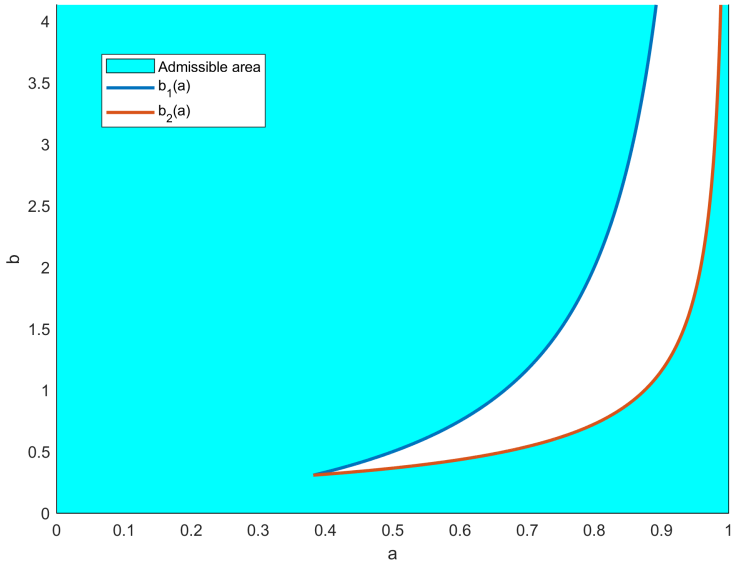


Figure 2.1: Graphical representation of conditions 2.17.

Hence if

$$\begin{cases} b > -\frac{1}{2} + \frac{1}{2}\sqrt{1 + \frac{1}{1-a}} & \text{if } 0 < a < \frac{3-\sqrt{5}}{2} \\ b > \frac{1}{2}\frac{a}{1-a} & \text{if } \frac{3-\sqrt{5}}{2} < a < 1 \end{cases}. \tag{2.16}$$

then inequality 2.14 holds.

As a result, merging condition 2.15 and condition 2.16 together, inequality 2.14 holds if and only if

$$\begin{cases} b > 0 & \text{if } 0 < a < \frac{3-\sqrt{5}}{2} \\ b < -\frac{1}{2} + \frac{1}{2}\sqrt{1 + \frac{1}{1-a}} \text{ or } b > \frac{1}{2}\frac{a}{1-a} & \text{if } \frac{3-\sqrt{5}}{2} < a < 1 \end{cases}. \tag{2.17}$$

The set of admissible a's and b's is shown in figure 2.1. Let $\mathcal{D}$ denote the set of a and b that satisfy conditions 2.17 and denote by $\overline{\mathcal{D}}$ its topological closure.

Step 2 For any $(a, b) \in \mathcal{D}$,

$$\begin{aligned} -\int_0^1 \ln(h(t))dt &= \int_0^1 \ln\left(\frac{1}{h(t)}\right) \\ &\geq \int_0^1 \left[1 - h(t) + a\frac{\left(\frac{1}{h(t)} - 1\right)^2}{\frac{1}{h(t)}\left(1 + b\frac{1}{h(t)}\right)}\right] dt \\ &= \int_0^1 a\frac{(h(t)-1)^2}{h(t)+b}dt. \end{aligned}$$

Using Cauchy-Schwartz inequality,

$$\begin{aligned}
-\int_0^1 \ln(h(t)) &\geq \int_0^1 a\frac{(h(t)-1)^2}{h(t)+b}dt \\
&= \int_0^1 a\frac{(h(t)-1)^2}{h(t)+b}dt\frac{1}{1+b}\int_0^1 (b+h(t))dt \\
&= \int_0^1 a\frac{(h(t)-1)^2}{(\sqrt{h(t)+b})^2}dt\frac{1}{1+b}\int_0^1 (\sqrt{b+h(t)})^2dt \\
&\geq \frac{a}{1+b}\left(\int_0^1 |h(t)-1|\frac{h(t)+b}{h(t)+b}dt\right)^2 \\
&= \frac{a}{1+b}\left(\int_0^1 |h(t)-1|dt\right)^2.
\end{aligned}$$

As a result, for any $(a,b) \in \mathcal{D}$,

$$-\int_0^1 \ln(h(t)) \geq \frac{a}{1+b}\left(\int_0^1 |h(t)-1|dt\right)^2. \tag{2.18}$$

Step 3 Consider the function $U(a,b) : \overline{\mathcal{D}} \to \mathbb{R}$ defined as

$$U(a,b) = \frac{a}{1+b}.$$

Then, for each $(a,b) \in \mathcal{D}$,

$$\begin{aligned}
\nabla U(a,b) &= \left(\frac{1}{1+b}, -\frac{a}{(1+b)^2}\right) \\
&\neq 0.
\end{aligned}$$

As a result, since U is continuous in $\overline{\mathcal{D}}$, its maximum values are attained on its border $\partial\mathcal{D}$.

Let us first note that

$$U(1,0) = 1.$$

Then, it is easy to see that

$$\begin{aligned}
U(a, b_1(a)) &= \frac{a}{1+b_1(a)} \\
&= \frac{a}{1+\frac{1}{2}\frac{a}{1-a}} \\
&< 1
\end{aligned}$$

and

$$\begin{aligned} U(a, b_2(a)) &= \frac{a}{1 + b_2(a)} \\ &= \frac{a}{\frac{1}{2} + \frac{1}{2}\sqrt{1 + \frac{1}{1-a}}} \\ &< 1. \end{aligned}$$

As a result, for every $(a, b) \in \mathcal{D} \subset \overline{\mathcal{D}}$,

$$U(a, b) < 1.$$

Taking the supremum over a and b in inequality 2.18 concludes the proof. □

A direct consequence is the following Pinsker-like inequality for differential entropy.

Proposition 2.2.6 (Pinsker inequality for differential entropy)**.** *Let $f : [0, 1] \to [0, 1]$ be a measurable function. Then*

$$-E_f \geq \| |f^{*\prime}| - 1 \|_1^2. \tag{2.19}$$

Furthermore, let $f : \Omega \to \mathbb{R}$ be a measurable bounded function. Then

$$\begin{aligned} -E_f + \ln(|f(\Omega)|) &\geq \frac{|\Omega|^2}{|f(\Omega)|^2} \left(\frac{1}{|\Omega|} \int_0^{|\Omega|} \left| |f^{*\prime}| - \frac{|f(\Omega)|}{|\Omega|} \right| dt \right)^2 \\ &= \frac{1}{|f(\Omega)|^2} \left\| |f^{*\prime}| - \frac{|f(\Omega)|}{|\Omega|} \right\|_1^2 \end{aligned}$$

Proof. The first part trivially follows from Lemma 2.2.5 taking $h(t) = f^{*\prime}(t)$. For the second part it suffices to apply inequality (2.19) to $\tilde{f}(t) = f(|\Omega|t)/|f(\Omega)|$. □

Since the Kullback-Leibler divergence between two functions f and g is the entropy of the composition $g^{-1} \circ f$, Pinsker inequality follows immediately.

Proposition 2.2.7 (Pinsker inequality)**.** *Let $f, g : [0, 1] \to \mathbb{R}$ be measurable functions, g strictly monotonically increasing, such that $f([0, 1]) = g([0, 1])$ and the composition $g^{-1} \circ f$ is well defined. Then*

$$D_{KL}(f||g) \geq \left(\int_0^1 \left| \frac{|f^{*\prime}(t)|}{|g^{*\prime}(g^{*-1}(f^*(t)))|} - 1 \right| dt \right)^2.$$

Furthermore, let $f : \Omega_1 \to \mathbb{R}$, $g : \Omega_2 \to \mathbb{R}$, g monotonically increasing. Then

$$D_{KL}(f||g) + \ln(|\Omega_2|) \geq \frac{|\Omega_1|^2}{|\Omega_2|^2} \left(\frac{1}{|\Omega_1|} \int_0^{|\Omega_1|} \left| \frac{|f^{*\prime}(t)|}{|g^{*\prime}(g^{*-1}(f^*(t)))|} - \frac{|\Omega_2|}{|\Omega_1|} \right| dt \right)^2$$

Proof. Both the first and the second part follows from Proposition 2.2.6 observing that, if $f, g : [0,1] \to \mathbb{R}$, then $g^{-1} \circ f : [0,1] \to [0,1]$ and that

$$((g^*)^{-1}(f^*(t)))' = \frac{f^{*\prime}(t)}{g^{*\prime}(g^{*-1}(f^*(t)))}.$$

□

Remark 2.2.8. *From Proposition 2.2.7 it is possible to deduce the equivalent Pinsker inequality for random variables. Given two random variables X and Y attaining the same values, there exist two function $f, g : [0,1] \to \mathbb{R}$ such that $f([0,1]) = g([0,1])$ and two independent uniform random variables U_1 and U_2 such that $X = f(U_1)$ and $Y = g(U_2)$. Since $D_{KL}(X||Y) = D_{KL}(f||g)$ then, by eq. (2.12) and Proposition 2.2.7,*

$$D_{KL}(X||Y) \geq \|p_X - p_Y\|_1^2$$

Pinsker inequality confirms that differential entropy discriminates well a function f from the identity. Indeed, $D_{KL}(f, ||g) = 0$ if and only if $f = g$. In this sense, even if Kullback-Leibler is not a distance (it doesn't satisfy triangular inequality), it provides a metric between functions with respect to composition.

It is also possible to observe that in the bound given in Lemma 2.2.5, equality is actually reached by some function.

Proposition 2.2.9. *There exists at least one function $h(t) : [0,1] \to \mathbb{R}^+$, $\int h(t)dt = 1$ and $h(t) \not\equiv 1$ such that*

$$-\int_0^1 \ln(h(t))dt = \left(\int_0^1 |h(t) - 1|dt\right)^2. \tag{2.20}$$

Proof. In order to prove the proposition, we will explicitly construct a piece-wise constant function such that both sides of eq. 2.20 are equal to one.

Let us consider $\lambda \in (0,1)$, $a \in (0,1)$ and $b \in (1, +\infty)$ and define

$$h(t) = a\mathbf{1}_{[0,\lambda)}(t) + b\mathbf{1}_{\lambda,1)}(t).$$

The conditions on λ, a and b such that (2.20) is satisfied are

$$\begin{cases} -\lambda \ln(a) - (1-\lambda)\ln(b) = 1 \\ \lambda(1-a) + (1-\lambda)(b-1) = 1 \\ \lambda a + (1-\lambda)b = 1. \end{cases} \tag{2.21}$$

From the last equation we get $b = 1 + \lambda(1-a)/(1-\lambda)$, and note that, since $a < 1$, then $b > 1$. Substituting in the second equation, we get $a = 1 - 1/(2\lambda)$ and note that $a < 1$ if and only if $\lambda > 1/2$. Substituting both a and b in the first equation we get

$$-\lambda \ln\left(1 - \frac{1}{2}\frac{1}{\lambda}\right) - (1-\lambda)\ln\left(1 + \frac{1}{2}\frac{1}{1-\lambda}\right) = 1.$$

Now define

$$H(\lambda) = -\lambda \ln\left(1 - \frac{1}{2}\frac{1}{\lambda}\right) - (1-\lambda)\ln\left(1 + \frac{1}{2}\frac{1}{1-\lambda}\right) - 1$$

and observe that $H(\lambda)$ is continuous if $\lambda \in (1/2, 1)$ and that

$$\lim_{\lambda \to 1/2^+} H(\lambda) = +\infty > 0,$$
$$\lim_{\lambda \to 1^-} H(\lambda) = \ln(2) - 1 < 0.$$

As a result there exists $\tilde{\lambda} \in (1/2, 1)$ such that $H(\tilde{\lambda}) = 0$, a and b are well defined and all the equations in system 2.21 are satisfied.

Hence

$$\begin{aligned} -\int_0^1 \ln(h(t))dt = 1 & \\ &= \int_0^1 |h(t) - 1|dt \\ &= \left(\int_0^1 |h(t) - 1|dt\right)^2. \end{aligned}$$

□

2.3 Entropy and noise

An important issue that we will address in this section is how differential entropy is affected by additive noise. Several results exist about the rearrangement of a sum of functions [105, 108, 114], but since they are always given as bounds, they are of no use to estimate the derivative of the rearrangement of a sum of functions.

As a first step, let us establish how distribution functions behave with respect to noise.

Proposition 2.3.1. *Let $\Omega \subset \mathbb{R}$, μ be the Lebesgue measure and let $f(t), g(t) : \Omega \to \mathbb{R}$ be μ-measurable functions. Then*

$$M_{f+g}(y) = \frac{1}{\mu(\Omega)} \int_{\mathbb{R}} M_f(y - \eta)|m_g(\eta)|d\eta. \tag{2.22}$$

Proof. Let us consider two independent uniform random variables U_1, U_2 defined on Ω and denote by $F = f \circ U_1$ its composition with f and by $G = g \circ U_2$ its composition with g. As a result F and G are independent.

Observe that

$$p_F(y) = \frac{1}{\mu(\Omega)}|m_f(y)|$$

and the same holds for G and g.

Then, let us consider the probability space $\mathcal{P} = (\Omega, \mathcal{B}, \mathbb{P}_\mu)$, where $\mathbb{P}_\mu = \mu/\mu(\Omega)$ and $\mathcal{B}$ is the Borel σ-algebra. Using the law of total probabilities,

$$\begin{aligned}\frac{1}{\mu(\Omega)} M_{f+g}(y) &= \frac{1}{\mu(\Omega)} \mu(\{f(t) + g(t) > y\}) \\ &= \frac{1}{\mu(\Omega)} \mu(\{f(U_1) + g(U_2) > y\}) \\ &= \mathbb{P}(F + G > y) \\ &= \int_{\mathbb{R}} \mathbb{P}(G > y - \eta) p_G(\eta) d\eta \\ &= \frac{1}{|\mu(\Omega)|^2} \int_{\mathbb{R}} M_f(y - \eta) |m_g(\eta)| d\eta.\end{aligned}$$

□

Albeit Proposition 2.3.1 gives an explicit representation of the distribution function of the sum of two functions, the convolution in eq. (2.22) might not be easy or even possible to compute.

Nevertheless, it is possible to use this result to estimate the influence of noise on the rearrangement of f and on its derivative.

If the deterministic distribution function of the noise is known and eq. (2.22) can be explicitly computed and inverted, then the rearrangement of the sum can be written explicitly. If the distribution function of the noise is not known, we shall assume that the noise is the realization of some random variable with known probability density function. We shall also assume that the noise corrupting f has a deterministic distribution function that well approximates the behaviour of the underlining random variable.

Definition 2.3.2. *We say that a function $h : \Omega \to \mathbb{R}$ is ε-**representative** of a random variable H if there exists $\varepsilon > 0$ such that.*

$$\left| p_H(y) - \frac{1}{|\Omega|} |m_h(y)| \right| \leq \varepsilon.$$

The value ε is called the representativeness bound.

Proposition 2.3.3. *Let $f(t) : \Omega \to \mathbb{R}$ be a measurable function, let H be a random variable and let $h(t) : \Omega \to \mathbb{R}$ be ε-representative for H.*

Then

$$m_{f+h}(y) \approx \int_{\mathbb{R}} m_f(y - \eta) p_H(\eta) d\eta,$$

in the sense that

$$\left| m_{f+h}(y) - \int_{\mathbb{R}} m_f(y - \eta) p_H(\eta) d\eta \right| \leq \varepsilon |\Omega|.$$

Proof. Differentiating both sides of eq. (2.22) leads to

$$m_{f+h}(y) = \frac{1}{|\Omega|}\int_{\mathbb{R}} m_f(y-\eta)|m_h(\eta)|d\eta.$$

By hypothesis,

$$p_H(y) = \frac{1}{|\Omega|}|m_h(y)| + R_\varepsilon(y),$$

with $|R_\varepsilon(y)| \leq \varepsilon$. Hence

$$\begin{aligned}\left|m_{f+h}(y) - \int_{\mathbb{R}} m_f(y-\eta)p_H(\eta)d\eta\right| &= \left|m_{f+h}(y) - \int_{\mathbb{R}} m_f(y-\eta)\left(\frac{1}{|\Omega|}|m_h(\eta)| + R(\eta)\right)d\eta\right| \\ &\leq \int_{f(\Omega)\cap h(\Omega)} |m_f(y-\eta)||R_\varepsilon(\eta)|d\eta \\ &\leq \varepsilon\int_{\mathbb{R}} |m_f(y-\eta)|d\eta \\ &\leq \varepsilon|\Omega|.\end{aligned}$$

□

2.3.1 Uniform noise

Uniform noise is the simplest case. Let us suppose that a signal f is corrupted by uniform noise n distributed in $[-\delta, \delta]$.

We are going to provide an estimate of $|m_{f+n}((f+n)^*(t)) - m_f(f^*(t))|$ that depends on the noise width δ, the noise representativeness bound ε and the derivatives of f^* and of $(f+n)^*$. Even if the noise n is non differentiable, $(f+n)^*$ is indeed differentiable almost everywhere, since it is a monotonically decreasing function.

As a first step, adding and subtracting $m_f((f+n)^*(t))$, we get

$$\begin{aligned}|m_{f+n}((f+n)^*(t)) - m_f(f^*(t))| \leq\ & |m_{f+n}((f+n)^*(t)) - m_f((f+n)^*(t))| \\ & + |m_f((f+n)^*(t)) - m_f(f^*(t))|\,.\end{aligned} \tag{2.23}$$

Assuming that the noise realization ε-approximates the true distribution, i.e. $|m_n(y)/|\Omega| - p_n(y)| \leq \varepsilon|\Omega|$, in order to estimate the first term in (2.23) we set $y = (f+n)^*(t)$ and invoke Proposition 2.3.3 expanding $m_f(y-\eta)$ using a Taylor expansion around y with the remainder expressed in integral form:

$$\begin{aligned}m_{f+n}(y) &= \frac{1}{2\delta}\int_{-\delta}^{\delta} m_f(y-\eta)d\eta + R_\varepsilon(y) \\ &= m_f(y) + \frac{1}{2\delta}\int_{-\delta}^{\delta}\int_{y}^{y-\eta} m_f'(\xi)d\xi d\eta + R_\varepsilon(y).\end{aligned}$$

As a result, exploiting that $m_f(y) = f^{*\prime}(f^{*-1}(y))$,

$$
\begin{aligned}
|m_{f+n}(y) - m_f(y)| &\leq \varepsilon + \frac{1}{2\delta}\int_{-\delta}^{\delta}\int_{0}^{\eta} |m_f'(y+\xi)| d\xi d\eta \\
&\leq \varepsilon + \int_{-\delta}^{\delta} |m_f'(y+\xi)| \frac{1}{2\delta}\int_{-\delta}^{\delta} d\eta d\xi \\
&\leq \varepsilon + \int_{-\delta}^{\delta} |m_f'(y+\xi)| d\xi \\
&= \varepsilon + \int_{y-\delta}^{y+\delta} \left| \frac{f^{*\prime\prime}(f^{*-1}(\xi))}{(f^{*\prime}(f^{*-1}(\xi)))^3} \right| d\xi \\
&= \varepsilon + \int_{\Gamma_{y,\delta}} \frac{1}{|f^{*\prime}(\tau)|^2} d\tau, \qquad (2.24)
\end{aligned}
$$

where the change of variables $\tau = f^{*\prime}(f^{*-1}(y))$ has been used and $\Gamma_{y,\delta} = f^{*-1}([y-\delta, y+\delta])$.

The only obstacle left to give a useful estimate is that, since the rearrangement is a non-local operator, in general $(f+n)^*(t) \neq f^*(t)$.

It is easy to check that, if $|(f+n)(t) - f(t)| < 2\delta$ for any $t \in \Omega$, then $|(f+n)^*(t) - f^*(t)| < 2\delta$ for any $t \in [0, |\Omega|]$ (see also [105]).

With this simple observation it is possible to estimate $m_{f+n}(f(t))$ and bound the second term of (2.23).

Since

$$
m_f(f^*(t) + \Delta f^*) = m_f(f(t)) + \int_{f^*(t)}^{f^*(t)+\Delta f^*} m_f'(\tau) d\tau.
$$

then, setting $\Delta f^* = (f+n)^*(t) - f^*(t)$ and observing that $|\Delta f^*| \leq 2\delta$,

$$
\begin{aligned}
|m_f((f+n)^*(t)) - m_f(f^*(t))| &= |m_f(f^*(t) + \Delta f^*) - m_f(f^*(t))| \\
&\leq \int_{f^*(t)-|\Delta f^*|}^{f^*(t)+|\Delta f^*|} |m_f'(\xi)| d\xi \\
&\leq \int_{f^*(t)-\delta}^{f^*(t)+\delta} \left| \frac{f^{*\prime\prime}(f^{*-1}(\xi))}{(f^{*\prime}(f^{*-1}(\xi)))^3} \right| d\xi \\
&= \int_{\Gamma_{y,\delta}} \frac{1}{|f^{*\prime}(\tau)|^2} d\tau, \qquad (2.25)
\end{aligned}
$$

where $\Gamma_{y,\delta} = f^{*-1}([y-\delta, y+\delta])$.

Finally, putting together estimates (2.24) and (2.25), the following proposition is proved.

Proposition 2.3.4. *Let $f(t) : \Omega \to \mathbb{R}$ be a measurable function, let n be a noise uniformly distributed in $[-\delta, \delta]$, let ε be its representativeness bound and define*

$$
Q(t) = \int_{\Gamma_{((f+n)^*(t),\delta)}} \frac{1}{|f^{*\prime}(\tau)|^2} d\tau + \int_{\Gamma_{(f^*(t),\delta)}} \frac{1}{|f^{*\prime}(\tau)|^2} d\tau.
$$

Then, for any $t \in [0, |\Omega|]$,

$$
|m_{f+n}((f+n)^*(t)) - m_f(f^*(t))| \leq \varepsilon|\Omega| + Q_\delta(t).
$$

Proposition 2.3.4 guarantees that, if $\int |f'(t)|^{-2}dt$ is small, then the distribution functions of f and $f+n$ are close and, in particular, it is small when $f'(t) \gg 0$. As it was foreseeable, when f is "almost flat" (its derivative is close to zero), then the influence of noise is greater than where f is "steep" (its derivative is much greater than zero).

Proposition 2.3.5 (Entropy error under uniform noise)**.** *Let $f(t) : \Omega \to \mathbb{R}$ a measurable function and let n be a noise uniformly distributed in $[-\delta, \delta]$ and let ε be its representativeness bound. Define*

$$Q_\delta = \sup_{t\in[0,|\Omega|]} Q_\delta(t).$$

Then

$$|E_{f+n} - E_f| \leq |\varepsilon + Q_\delta|^2 \frac{1}{|\Omega|} \int_0^{|\Omega|} \frac{1}{|f^{*\prime}(t)|} dt$$

Proof. From Lemma 2.1.7 we can consider $\tilde{f} = f(|\Omega|t)$. In addition, for the sake of simplicity, we will consider $f : [0,1] \to \mathbb{R}$ and rescale the result at the end. By Proposition 2.3.4 it is possible to write

$$\begin{aligned} \ln(|m_{f+n}((f+n)^*(t))|) &= -\ln(-(f+n)^{*\prime}(t)) \\ &= -\ln\left(-f^{*\prime}(t) + R(t)\right), \end{aligned}$$

where

$$|R(t)| \leq \varepsilon + Q_\delta.$$

$$\begin{aligned} \int_0^1 \ln(|(f+n)^{*\prime}(t)|)dt &= \int_0^1 \ln(|f^{*\prime}(t)| + R(t))dt \\ &= \int_0^1 \left[\ln(|f^{*\prime}(t)|) + \int_{|f^{*\prime}(t)|}^{|f^{*\prime}(t)|+R(t)} \frac{1}{\xi}(|f^{*\prime}(t)| - \xi)d\xi\right] dt. \end{aligned}$$

Hence,

$$\begin{aligned} |E_{f+n} - E_f| &\leq \int_0^1 \int_{|f^{*\prime}(t)|}^{|f^{*\prime}(t)|+|R(t)|} \frac{1}{\xi}||f^{*\prime}(t)| - \xi| d\xi dt \\ &\leq \int_0^1 |R(t)| \int_{|f^{*\prime}(t)|}^{|f^{*\prime}(t)|+|R(t)|} \frac{1}{\xi} d\xi dt \\ &= \int_0^1 |R(t)| \ln\left(\frac{|f^{*\prime}(t)| + |R(t)|}{|f^{*\prime}(t)|}\right) dt \\ &\leq \sup_{t\in[0,|\Omega|]} |R(t)|^2 \int_0^1 \frac{1}{|f^{*\prime}(t)|} dt \\ &\leq |\varepsilon + Q_\delta|^2 \int_0^1 \frac{1}{|f^{*\prime}(t)|} dt. \end{aligned}$$

Rescaling the last term completes the proof. □

Remark 2.3.6. *Is it trivial that* $Q \leq \int_0^{|\Omega|} |f^{*\prime}(t)|^{-2} dt$*, but it is a less tight estimate. As noted before, even if the rearrangement is a global operator, its derivative is "local around the values", in the sense that, given* $t \in [0, |\Omega|]$*, it holds that*

$$|f^{*\prime}(t)| = \left(\sum_{t_i \in f^{*-1}(f(t))} \frac{1}{|f'(t)|} \right)^{-1},$$

meaning that $f^{*\prime}$ *in* t *depends on the values of* f' *in the set* $f^{*-1}(f(t))$*, i.e. the set of those* t_i *such that* $f^*(t_i) = f(t)$*.*

Remark 2.3.7. *Let us consider* $f : [0,1] \to \mathbb{R}$ *and a random variable* $F = f(U)$*, where* U *is a uniform random variable in* $[0,1]$*. Then, from definition 2.1.2 and from the definition of* p_F*,*

$$|E_{F+n} - E_F| \leq |\varepsilon + Q_\delta| \|p_F\|_2^2.$$

2.3.2 Bound on the entropy of the sum of two functions

Although Proposition 2.3.4 gives an explicit estimate on the error, the constants involved in the bound are very difficult to compute in practice.

A more useful bound can be given by directly estimating the entropy of the sum of functions. In order to do this, the following lemmma is required.

Lemma 2.3.8. *Let* $z(t) = (z_1(t), \dots, z_n(t)) : \Omega \subset \mathbb{R} \to \mathbb{R}^n$*,* $n \in \mathbb{N}$ *be a curve, let* $\mu(\Omega) = 1$ *and let*

$$\varphi(x_1, \dots, x_n) = \ln\left(e^{x_1} + \dots e^{x_n}\right)$$

Then

$$0 \leq \varphi\left(\int_\Omega z(t)dt\right) - \int_\Omega \varphi(z(t))dt \leq \int_0^1 \left\| z(t) - \int_\Omega z(s)ds \right\|_2^2 dt.$$

Proof. Without loss of generality, consider $\Omega = [0,1]$.

Let $\tilde{z} = (\int_0^1 z_1(t)dt, \dots \int_0^1 z_n(t)dt)$ and consider the second order Taylor expansion of φ centered in $\tilde{z}$:

$$\varphi(z(t)) = \varphi(\tilde{z}) + \nabla\varphi(\tilde{z}) \cdot (z(t) - \tilde{z}) + \frac{1}{2} H_\xi(z(t) - \tilde{z}, z(t) - \tilde{z}),$$

where ξ is such that $\|\xi - \tilde{z}\| < \|z(t) - \tilde{z}\|$ and $H_\xi(\cdot,\cdot)$ is the bilinear form associated to the Hessian matrix of φ in ξ.

Hence

$$\begin{aligned}\int_0^1 \varphi(z(t))dt - \varphi(\tilde{z}) &= \int_0^1 \left[\nabla\varphi(\tilde{z})\cdot(z(t)-\tilde{z}) + \frac{1}{2}H_\xi(z(t)-\tilde{z}, z(t)-\tilde{z})\right] dt \\ &= \nabla\varphi(\tilde{z})\cdot\left(\int_0^1 [z(t)-\tilde{z}]\,dt\right) + \frac{1}{2}\int_0^1 H_\xi(z(t)-\tilde{z}, z(t)-\tilde{z})dt \\ &= \frac{1}{2}\int_0^1 H_\xi(z(t)-\tilde{z}, z(t)-\tilde{z})dt.\end{aligned}$$

Let $\lambda_{\max}$ and $\lambda_{\min}$ be, respectively, the maximum and minimum eigenvalue of the matrix $H_x i$ associated to the bilinear form $H_\xi(\cdot,\cdot)$. Then

$$\lambda_{\min}\|x\|_2^2 \le H_\xi(x,x) \le \lambda_{\max}\|x\|_2^2.$$

As a first step to estimate $\lambda_{\min}$ and $\lambda_{\max}$, let us write down explicitely the generic element $(H_\xi)_{ij}$ of H_ξ:

$$(H_\xi)_{ij} = \begin{cases} \dfrac{\partial^2}{\partial x_i^2}\varphi(\xi) = \dfrac{e^{\xi_i}\left(\sum_{k=1,k\neq i} e^{\xi_k}\right)}{\left(\sum_{k=1} e^{\xi_k}\right)^2} & i = j \\ \dfrac{\partial^2}{\partial x_i x_j}\varphi(\xi) = -\dfrac{e^{\xi_i}e^{\xi_j}}{\left(\sum_{k=1} e^{\xi_k}\right)^2} & i \neq j \end{cases}$$

It is easy to check that $\det(H_\xi) = 0$ and, since φ is convex, then $\lambda_{\min} = 0$.

The maximum eigenvalue can be bounded from above by the Frobenius norm of H_ξ:

$$\begin{aligned}\|H_\xi\|_F^2 &= \sum_{i=1}^n\sum_{j=1}^n (H_\xi)_{ij}^2 \\ &= \sum_{i=1}^n \left(\frac{\left(e^{\xi_i}\left(\sum_{k=1,k\neq i} e^{\xi_k}\right)\right)^2}{\left(\sum_{k=1} e^{\xi_k}\right)^4} + \sum_{j=1,j\neq i}^n \frac{\left(e^{\xi_i}e^{\xi_j}\right)^2}{\left(\sum_{k=1} e^{\xi_k}\right)^4} \right) \\ &= \frac{1}{\left(\sum_{k=1} e^{\xi_k}\right)^4}\sum_{i=1}^n (e^{\xi_i})^2 \left(\left(\sum_{k=1,k\neq i} e^{\xi_k}\right)^2 + \sum_{j=1,j\neq i}^n (e^{\xi_j})^2 \right).\end{aligned}$$

Let us consider polar coordinates, i.e. ρ and $\eta_1,\ldots,\eta_n$ such that $e_i^\xi = \rho\eta_i$ and $\sum_{i=1}^n \eta_i^2 = 1$. Then

$$\begin{aligned}\|H_\xi\|_F^2 &= \frac{1}{\rho^4\left(\sum_{k=1}\eta_k\right)^4}\sum_{i=1}^n \rho^2\eta_i^2 \left(\left(\sum_{k=1,k\neq i}\rho\eta_k\right)^2 + \sum_{j=1,j\neq i}^n \rho^2\eta_j^2 \right) \\ &= \frac{1}{\left(\sum_{k=1}\eta_k\right)^4}\sum_{i=1}^n \eta_i^2 \left(\left(\sum_{k=1,k\neq i}\eta_k\right)^2 + \sum_{j=1,j\neq i}^n \eta_j^2 \right) \\ &\le 2.\end{aligned}$$

As a result, $\lambda_{\max} \leq 2$ and

$$0 \leq \varphi\left(\int_0^1 z(t)dt\right) - \int_0^1 \varphi(z(t))dt \leq \int_0^1 \|z(t) - \tilde{z}\|_2^2 dt$$

□

The following results bounds the entropy of the sum of n functions.

Proposition 2.3.9. *Let $f_1, \ldots, f_n : \Omega \to \mathbb{R}$ be measurable strictly monotone functions, let $f = f_1 + \ldots + f_n$ and let $n_f(y)$ be the number of solutions of the equation $f(t) = y$. Then*

$$-l \leq E_{f_1+\ldots+f_n} - \ln\left(e^{E_{f_1}} + \ldots + e^{E_{f_n}}\right) \leq L,$$

where

$$l = \frac{1}{\Omega}\int_\Omega \ln(n_f(f(t)))dt$$
$$L = \frac{1}{|\Omega|}\int_\Omega \sqrt{\sum_{i=1}^n \left(\ln(|f_i^{*\prime}(t)|) - E_{f_i}\right)^2}dt.$$

Proof.
Without loss of generality, consider $\Omega = [0, 1]$. For the lower bound, from Proposition 2.1.9,

$$\begin{aligned}
E_{f_1+\ldots+f_n} &= \int_0^1 \ln\left((f_1 + \ldots + f_n)^{*\prime}(t)\right) dt \\
&\geq \int_0^1 \ln\left(|f_1'(t)| + \ldots + |f_n'(t)|\right) dt - l \\
&= \int_0^1 \ln\left(e^{\ln(|f_1'(t))|} + \ldots + e^{\ln(|f_n'(t)|)}\right) dt - l \\
&= \int_0^1 \ln\left(e^{\ln(|f_1^{*\prime}(t)|)} + \ldots + e^{\ln(|f_n^{*\prime}(t)|)}\right) dt - l.
\end{aligned}$$

Applying the leftmost inequality of Lemma 2.3.8 with $z_i(t) = \ln(f_i^{*\prime}(t))$, then

$$\begin{aligned}
E_{f_1+\ldots+f_n} &\geq \int_0^1 \ln\left(e^{\ln(|f_1^{*\prime}(t)|)} + \ldots + e^{\ln(|f_n^{*\prime}(t)|)}\right) dt - l \\
&\geq \ln\left(e^{\int_0^1 \ln(|f_1^{*\prime}(t)|)dt} + \ldots + e^{\int_0^1 \ln(|f_n^{*\prime}(t)|)dt}\right) - l \\
&= \ln\left(e^{E_1} + \ldots + e^{E_n}\right) - l
\end{aligned}$$

Regarding the upper bound, from Proposition 2.1.9,

$$\begin{aligned} E_{f_1+\ldots+f_n} &\leq \int_0^1 \ln(|f_1'(t)| + \ldots + |f_n'(t)|)dt \\ &= \int_0^1 \ln\left(e^{\ln(|f_1'(t))|} + \ldots + e^{\ln(|f_n'(t)|)}\right) dt \\ &= \int_0^1 \ln\left(e^{\ln(|f_1^{*\prime}(t)|)} + \ldots + e^{\ln(|f_n^{*\prime}(t)|)}\right) dt. \end{aligned}$$

Applying the rightmost inequality from Lemma 2.3.8 with $z_i(t) = \ln(f_i^{*\prime}(t))$, then

$$\begin{aligned} E_{f_1+\ldots+f_n} &\leq \int_0^1 \ln\left(e^{\ln(|f_1^{*\prime}(t)|)} + \ldots + e^{\ln(|f_n^{*\prime}(t)|)}\right) dt \\ &\leq \ln\left(e^{E_1} + \ldots + e^{E_n}\right) + L. \end{aligned}$$

□

Remark 2.3.10. *If all the f_i's are monotonically increasing, then so it is f, hence the lower bound is just*

$$\ln\left(e^{E_{f_1}} + \ldots + e^{E_{f_n}}\right) \leq E_{f_1+\ldots+f_n},$$

which is very similary to what is known as power entropy inequality in information theory [115].

It is easy to see that, if a signal f is corrupted by an excessive amount of noise, the entropy of f is non-influent in the entropy of the sum.

More formally, if $|g| \gg 0$, then the ratio between E_{f+g} and E_g is approximately one, meaning that, asymptotically, the "bigger" function dominates.

Proposition 2.3.11. *Let $f : [0,1] \to \mathbb{R}$ be a continuous function with entropy E_f, let $g : [0,1] \to \mathbb{R}$ be another function with entropy E_g corrupting f and consider $\sigma > 0$ and the function $g_\sigma = \sigma g$. Then*

$$\lim_{\sigma \to +\infty} \frac{E_{f+g_\sigma}}{E_{g_\sigma}} = 1.$$

Proof. In order to consider the limit $\sigma \to +\infty$ in Proposition 2.3.9, it is necessary to prove that l and L do not depend on σ.

Observe that $E_{g_\sigma} = E_g + \ln(\sigma)$ and $|g_\sigma^{*\prime}(t)| = \sigma|g^{*\prime}(t)|$, hence

$$\begin{aligned} L^2 &\leq \int_0^1 \left(\ln(|f^{*\prime}(t)|) - E_f\right)^2 dt + \int_0^1 \left(\ln(|{g_\sigma}^{*\prime}(t)|) - E_{g_\sigma}\right)^2 dt \\ &\leq \int_0^1 \left(\ln(|f^{*\prime}(t)|) - E_f\right)^2 dt + \int_0^1 \left(\ln(|g^{*\prime}(t)|) - E_g\right)^2 dt \\ &\leq C^2 \end{aligned}$$

where C does not depends on σ.

On the other hand, in order to bound l observe that, by definition,

$$\begin{aligned} n_{f+g_\sigma}(y) &= |\{t \in [0,1] | f(t) + \sigma g(t) = y\}| \\ &= |\{t \in [0,1] | f(t) = y - \sigma g(t)\}|. \end{aligned}$$

Assume by contradiction that, for any fixed y, it is possible to find a sequence $\{\sigma_k\}_{k\in\mathbb{N}}$ such that $n_{f+g_{\sigma_k}}(y) \to +\infty$ as $\sigma_k \to +\infty$.

Then, for any σ_k it is possible to find (at least) a point t_{σ_k} such that $f(t_{\sigma_k}) = y - \sigma g(t_{\sigma_k})$, i.e. $t_{\sigma_k} \in f^{-1}(y - \sigma g(t_{\sigma_k}))$. By construction t_{σ_k} is well defined and, since f is continuous in the closed interval $[0,1]$, it can assume a given value only a finite number of times, say M. Hence

$$\begin{aligned} n_{f+g_{\sigma_n}}(y) &= |f^{-1}(y - \sigma g(t_{\sigma_n}))| \\ &\leq M, \end{aligned}$$

contradicting the hypothesis that $n_{f+g_\sigma}(y) \to +\infty$ as $\sigma_n \to +\infty$. As a result, there exists a constant C', independent from σ, such that $-l > -C'$.

By Proposition 2.3.9,

$$\begin{aligned} \frac{E_{f+g_\sigma}}{E_{g_\sigma}} &\leq \frac{1}{E_{g_\sigma}} \ln\left(e^{E_f} + e^{E_{g_\sigma}}\right) + \frac{L}{E_{g_\sigma}} \\ &= \ln\left(1 + e^{E_f - E_{g_\sigma}}\right) + \frac{L}{E_{g_\sigma}}. \end{aligned}$$

Since $E_{g_\sigma} \to +\infty$ as $\sigma \to +\infty$ and L does not depend on σ, then

$$\lim_{\sigma \to +\infty} \frac{E_{f+g_\sigma}}{E_{g_\sigma}} \leq 1.$$

Since l also does not depend on σ, the same technique proves the opposite inequality.

□

Proposition 2.3.11 shows that, for a strong enough noise, the entropy of the noise is dominant and the constants L and l can be ignored.

Proposition 2.3.9 can be used to estimate the strength of the noise affecting a signal, provided the entropy of the signal is known.

Consider a function f and a noise n. Then, for Proposition 2.3.9,

$$E_{f+n} \approx \ln\left(e^{E_f} + e^{E_n}\right).$$

As a result, the entropy of the noise n can be estimated as

$$e^{E_n} \approx e^{E_{f+n}} - e^{E_f}. \tag{2.26}$$

Indeed, in figure 2.2 the result of a numerical experiment to validate propositions 2.3.9 and 2.3.11 is presented. The function Bumps, provided by the MatLab function **wnoise** has been normalized and then corrupted by gaussian noise of increasing variance σ and the exponential of the entropy of n has been estimated according to eq. (2.26).

If the entropy of the clean signal E_f is not known, early numerical evidence suggests that the entropy of f can be estimated from a smoothing of $f + n$, but no further studies have been conducted.

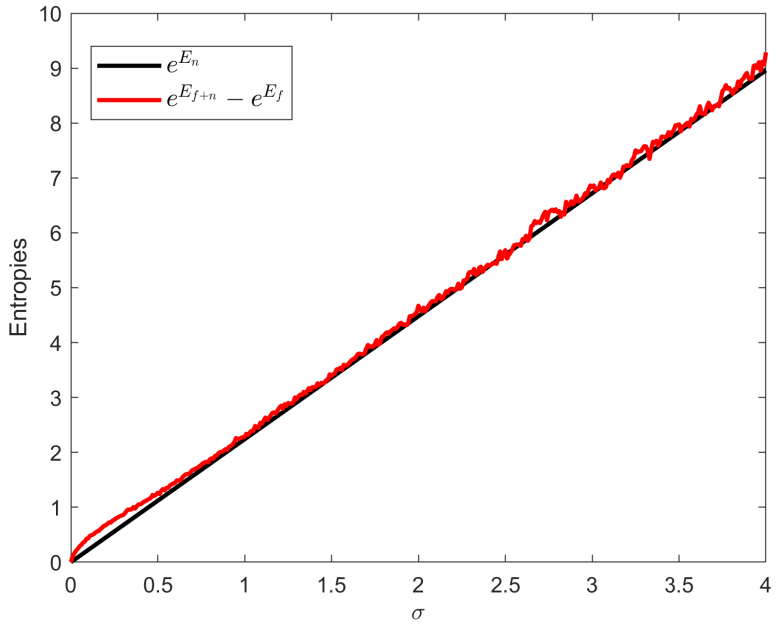


Figure 2.2: Estimation of the entropy of the noise n from the entropies of $f + n$ and f. Real and estimated entropy of noise n are displayed versus noise variance σ.

Chapter 3

Automatic coefficients selection in a wavelet basis

Contents

Function approximation is a mathematical tool that plays a key role in applied mathematics. In this thesis we are mainly concerned about the problem of function approximation in Hilbert spaces. Given a separable Hilbert space $\mathbb{H}$ and a function $f \in \mathbb{H}$, it is always possible (up to density) to write f as a linear combination of elements of an orthonomal basis of $\mathbb{H}$. In real applications, it is usually possible to choose a basis with good sparsity characteristics, depending on the problem. This has been done with great success in several real world applications, such as voice communications [19, 6], lossy image compression [69, 71, 70], audio processing [72], signal and image denoising [74, 75, 73, 116] and pattern recognition [76].

In Chapter 1 an extensive overview of existing methods for function approximation has been given and it should be noted that almost all the existing methods require some form of parameter tuning.

In the case of linear and non-linear approximation in a basis the number of coefficients must be set in advance while in the thresholding techniques, such as Donoho's universal thresholding or SURE thresholding, the knowledge of the noise's strength is required. Furthermore, any method requiring iterative procedures, such as matching pursuit or functional minimization (required in non-orthogonal LASSO or elastic net functional) requires a stopping criterion whose choice significantly affects the performances.

Threshold estimation may present difficulties, as well as the choice of the relaxation parameter in regularization problems. As pointed out in Chapter 1, existing methods such as L-curve and GCV present several technical problems that limit their application.

A different approach from the ones described in Chapter 1 is represented by the **Minimum Description Length** (MDL) principle, due to Rissanen [117, 118]. In the MDL framework, a collection $\mathcal{H}$ of suitable models is given a priori. Given an input datum x, for each $H \in \mathcal{H}$ define $L(H)$ as the length, in bit, needed to encode the model and $L(x|H)$ as the length, in bit, of the encoding of x using the model H. Then the model that minimizes $L(H) + L(x|H)$ is chosen. Since a greater length corresponds to a greater complexity, a trade off between the complexity of the model and the complexity of the description is reached in this way.

The MDL approach is notable for approaching the problem not from the point of view of *how good* the approximation is but from the point of view of *how complex* its description is. This different point of view is particularly advantageous in cases where overfitting may be a problem, such as in the case of polynomial fitting of noisy data without constraint on the degree of the polynomials. However, practical implementation of MDL may be difficult and, if it is not possible to write explicitly the minimum, numerical techniques must be employed.

In this chapter, a novel model for the automatic selection of expansion coefficients in an Hilbert basis is presented. Given a function $f \in \mathbb{H}$ and a basis $\mathcal{B} = \{\psi_j\}_{j\in\mathbb{N}} \subset \mathbb{H}$, the proposed model uses novel information measures to select the expansion coefficients that are the most significant, in the sense of information contribution. Three novel measures are presented in this chapter, relying on the differential entropy for continuous functions presented in Chapter 2: ENID, which has been inspired by Vitanyi's Normalized Information Distance (NID), KLID and JID, which have been derived from Kullback-Leibler divergence. It should be noted that the philosophy of the approach proposed in this thesis is the same as MDL approach, i.e. taking into account the complexity of the representation. A preliminary version of this model has been proposed in [119] and in [120]. These early results have been improved and expanded in this thesis.

3.1 Compression based information measure

Instead of relying on Shannon entropy, as many information theory techniques do, differential entropy is considered as the base for Entropic NID (ENID). Under suitable hypothesis, differential entropy may be seen as a measure of complexity of a continuous function and it is evaluated using the simple and straightforward numerical scheme proposed in Chapter 2.

3.1.1 The proposed model

Let us consider an Hilbert space $\mathbb{H}$ endowed with an orthonormal basis $\mathcal{B} = \{\psi_j\}_{j\in\mathbb{N}} \subset \mathbb{H}$ and a function $f \in \mathbb{H}$. Let $f_i = \langle f, \psi_i \rangle$ be the i-th expansion coefficient of f on the basis $\mathcal{B}$. As seen in the previous chapter, in order to compute an approximation of f, it is convenient to consider the sequence $\{f_i^*\}_{i\in\mathbb{N}}$ of decreasingly rearranged coefficients, i.e. $f_i^* = f_{k_i}$, where the indexes k_i's are chosen such that $|f_{i+1}^*| \leq |f_i^*|$. As seen in the previous chapter, a straightforward way to produce an approximation is to consider the first M rearranged coefficients and this non-linear approximation approach is equivalent to thresholding.

It is then possible to define a sequence of approximations

$$f_n = \sum_{i=1}^{n} f_i^* \psi_i, \qquad n \in \mathbb{N},$$

which corresponds to consider all the possible approximations for all the viable thresholding values. As pointed out in the previous chapter, it is trivial that the higher n, the lower the approximation error. Unfortunately, albeit methods have been proposed, finding n that realizes the best trade-off between number of coefficients and fidelity to f is still an open problem.

The proposed model looks at the problem as a source separation problem, assuming that the expansion coefficients f_i of f are emitted from two different sources: one source emits the "most representative" coefficients, while the other emits the "less representative" coefficients (see fig. 3.1). Then the problem of finding the optimal n is equivalent to the problem of separating these two sources. L-Curve method might be employed for this kind of problem but, as pointed out in the first chapter, usually the optimal point in an L-curve is identified with the point that maximizes its second derivative. Unfortunately, rearranged coefficients suffer from several small variations, making them non-smooth in practice: a direct computation of the second derivative is unreliable. Similarly, automatic fitting methods would over-regularize the coefficients curve. Another technique to find the optimal point in L-curves is the Occam filter technique [99]: two straight lines are estimated at both ends of the L-curve and the position of the optimal point is defined as the intersection point of said two lines. Although the Occam filter has been proven reliable, it strongly depends on how the two straight lines are estimated. Instead, the model proposed in this chapter relies on a novel information measure inspired by Vitanyi's Normalized Information Distance (NID).

3.1.1.1 Information measures

In the following a very brief introduction to information measures will be given.

Definition 3.1.1. *Let x be a string. Then its* ***Kolmogorov complexity*** $\mathbb{K}(x)$ *is defined as the length of the smallest program that outputs x on a Turing machine.*

Definition 3.1.2. *Let x and y two different strings. Then the* ***conditional Kolmogorov complexity*** $\mathbb{K}(x|y)$ *of x w.r.t. y is defined as the length of the smallest program on a Turing machine that outputs x, given y as an input.*

Kolmogorov complexity for a given string is universal in the sense that, up to a constant, it does not depend on the particular Turing machine considered. Kolmogorov complexity is a very large topic and a complete exposition is beyond the scope of this thesis. The interested reader may refer to [98, 121].

The Normalized Information Distance, introduced by Vitanyi et al. in [122], uses Kolmogorov complexity and conditional Kolmogorov complexity to assess the difference, in terms of information contribution, between two strings.

Definition 3.1.3. *Let x and y be strings. Then the* ***Normalized Information Distance (NID)*** *is defined as*

$$\mathrm{NID}(x,y) = \frac{\max\{\mathbb{K}(x|y), \mathbb{K}(y|x)\}}{\max\{\mathbb{K}(x), \mathbb{K}(y)\}}.$$

Since Kolmogorov complexity, and hence NID, is not computable, Vitanyi et al. [123] proposed an approximation of NID based on data compression, the Normalized Compression Distance (NCD).

Definition 3.1.4. *Let C be a lossless compressor and let x and y two strings and xy their concatenation. In addition, let $|C(z)|$ be the size of the compression of the string z. Then the* ***Normalized Compression Distance (NCD)*** *between x and y is defined as*

$$\mathrm{NCD}(x,y) = \frac{|C(xy)| - \min\{|C(x)|, |C(y)|\}}{\max\{|C(x)|, |C(y)|\}}.$$

The main intuition behind the definition of NCD is that, if two strings x and y are similar, then the difference between the size $|C(xy)|$ of the compression of their concatenation and the size $|C(x)|$ of the compression of x is small. More accurately, assuming the knowledge of x (or of a compressed version $C(x)$ of x), then y may be obtained using only $|C(xy)| - |C(y)|$ bits.

Hence

$$\mathbb{K}(x|y) \approx |C(xy)| - |C(y)|$$

and the definition of NID comes from the fact that

$$\max\{|C(xy)| - |C(y)|, |C(xy)| - |C(x)|\} = |C(xy)| - \min\{|C(x)|, |C(y)|\}.$$

In [123] it has been proved that NCD is indeed an approximation of NID, whose accuracy depends on the characteristics of the adopted compressor. Unfortunately, practical use of NCD is not without issues. Since compression algorithms are often batch-based, they fail to recognize, and hence properly compress, distant identical patterns. Moreover, compression is a computationally intensive operation. For these reasons, an alternative information measure that is independent of the compressor choice has been investigated. To reach this goal, the compressor in NCD must be substituted for some other measure of complexity. To this aim, differential entropy has been considered and an entropic version of NID, named Entropic NID (ENID), has been defined in [119, 120].

3.1.2 ENID

Let us model the amplitudes of the expansion coefficients as a continuous function $w(t) : [0,1] \to [-1,1]$ – even if their amplitude exceeds 1, since expansion coefficients are bounded, it is always possible to perform such a normalization. Then the decreasing rearrangement $a(t) = |w|^*(t)$ of its absolute value $|w(t)|$ is a monotonically decreasing function $a(t) : [0,1] \to [0,1]$. An example of a is shown in Figure 3.1.

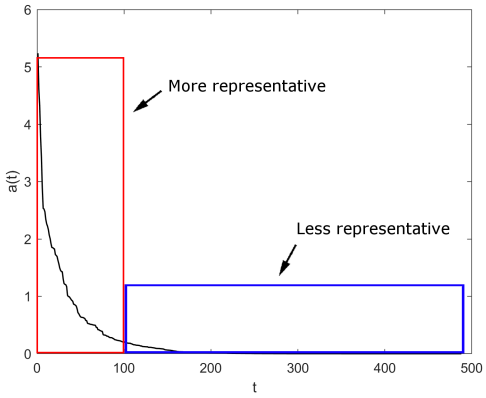


Figure 3.1: Rearranged absolute value of wavelet expansion coefficients using wavelet db2 of function Bumps. Difference between coefficients sources evidenced.

As explained in the previous sections, we are looking at the problem of coefficient selection as a problem of source separation. The set of coefficients representing the sources are the values that a assumes in two intervals $\Omega_1^\lambda = [0, \lambda]$ and $\Omega_2^\lambda = [\lambda, 1]$, $\lambda \in (0, 1)$. Let $E = E_a$ be the entropy of the whole set of coefficients, E_1^λ the entropy of a^* restricted to Ω_1^λ and E_2^λ the entropy of a^* restricted to Ω_2^λ. The information measure ENID is defined as follows.

Definition 3.1.5. *Given two subsets of expansion coefficients Ω_1^λ and Ω_2^λ, defined as before, the ENID between Ω_1^λ and Ω_2^λ is defined as*

$$\mathrm{ENID}(\lambda) = \frac{\big||E| - \min\{\lambda|E_1^\lambda|, (1-\lambda)|E_2^\lambda|\}\big|}{\max\{\lambda|E_1^\lambda|, (1-\lambda)|E_2^\lambda|\}}. \tag{3.1}$$

ENID compares the complexity of the whole set of coefficients to the minimum of the complexities of the two sub intervals Ω_1^λ and Ω_2^λ, normalized by the maximum of those complexities. This means that ENID measures the difference between the two sets of coefficients as the difference between the complexity of the whole set and the complexity of the simplest one of the two subsets. We claim that the source separation point is located where ENID is minimum: here both Ω_1^λ and Ω_2^λ are emitted from a single source. In terms of complexity, it occurs when they are most similar.

The separation point will be then identified by the value of λ for which $\mathrm{ENID}(\lambda)$ is minimum.

The following proposition ensures that ENID is well defined and always attains a minimum.

Proposition 3.1.6. *Let $a(t) : [0, 1] \to [0, 1]$ be a continuous, monotonically decreasing function and $E_a \neq 0$. Then the following properties hold.*

1. $\mathrm{ENID}(\lambda)$ *is well defined for* $\lambda \in [0, 1]$;
2. $0 \leq \mathrm{ENID}(\lambda) \leq 1$, $\forall \lambda \in [0, 1]$;

3. ENID(λ) *is continuous w.r.t* λ *and it attains a minimum in* $[0, 1]$.

Proof. (1) Let us observe that, from eq. (2.5), it is immediate that $E \leq 0$, $E_1^\lambda \leq 0$ and $E_2^\lambda \leq 0$. From Proposition 2.1.10 we know that, since $a(t)$ is monotonic,

$$E = \lambda E_1^\lambda + (1-\lambda)E_2^\lambda + B(\lambda),$$

where

$$B(\lambda) = -\lambda \ln(\lambda) - (1-\lambda)\ln(1-\lambda).$$

Since the signs of the entropies E, E_1^λ and E_2^λ are known, then

$$|E| + B(\lambda) = \lambda|E_1^\lambda| + (1-\lambda)|E_2^\lambda|.$$

As a consequence, it is not possible that both E_1^λ and E_2^λ are 0. Note that the hypothesis $E_a \neq 0$ is only needed in order to prove that ENID(λ) is well defined up to the border of $[0, 1]$. If this hypothesis is removed, it is still true that ENID(λ) is well defined in $(0, 1)$.

(2) The leftmost bound is trivial from the definition of ENID. For the rightmost bound, Proposition 2.1.10 and the relation $A + B = \max\{A, B\} + \min\{A, B\}$, it follows

$$\begin{aligned} |E| - \min\{\lambda|E_1^\lambda|, (1-\lambda)|E_2^\lambda|\} &= \max\{\lambda|E_1^\lambda|, (1-\lambda)|E_2^\lambda|\} - B(\lambda) \\ &\leq \max\{\lambda|E_1^\lambda|, (1-\lambda)|E_2^\lambda|\}. \end{aligned} \tag{3.2}$$

Dividing both members of eq. (3.2) by $\max\{\lambda|E_1^\lambda|, (1-\lambda)|E_2^\lambda|\}$ the thesis follows.

(3) Both E_1^λ and E_2^λ are continuous function w.r.t λ: if $\lambda_1 \to \lambda_2$, $\lambda_1 < \lambda_2$, then

$$\begin{aligned} |E_1^{\lambda_1} - E_1^{\lambda_2}| &= \int_{\lambda_1}^{\lambda_2} \ln(|a^{*\prime}(t)|)dt + \lambda_1 \ln(\lambda_1) - \lambda_2 \ln(\lambda_2) \\ &\to 0, \end{aligned}$$

and the same result holds for E_2^λ.

It is worth observing that, if f and g are two continuous functions, then both $(f \wedge g)(t) = \max\{f(t), g(t)\}$ and $(f \vee g)(t) = \min\{f(t), g(t)\}$ are well defined, continuous functions with respect to the variable t.

As a consequence, ENID(λ) is a continuous function w.r.t λ, and it attains both a maximum and a minimum in $[0, 1]$.

□

3.1.3 The negativity of entropy

While Shannon entropy is always positive, differential entropy can be negative – indeed, in the definition of ENID, it is crucial that the differential entropies involved are negative. As a consequence, even if Shannon entropy is the average number of bits needed to encode a random variable, differential entropy lacks such a clear interpretation. In this paragraph a motivation of why differential entropy can be considered a complexity measure is given.

Let $p : [0,1] \to \mathbb{R}^+$ be a probability density function, E its differential entropy, N the number of quantization bins, $\Delta = 1/N$ the bandwidth, p^Δ the discrete quantized version of p and E^Δ its Shannon entropy.

If Δ is sufficiently small, then

$$E \approx E^\Delta + \ln(\Delta),$$

in the sense that

$$\lim_{\Delta \to 0} \left[E - E^\Delta - \ln(\Delta)\right] = 0.$$

On the other hand, Shannon entropy is equal to the logarithm of N minus the Kullback-Leibler divergence of the quantized version p^Δ of p from a uniform distribution U, hence

$$\begin{aligned} E &\approx E^\Delta + \ln(\Delta) \\ &= \ln(N) - D_{KL}(p^\Delta || U) + \ln(\Delta) \\ &= -D_{KL}(p^\Delta || U). \end{aligned}$$

As a result, it is possible to conclude that, if $|E|$ is large, then p is complex in the sense that, for every sufficiently fine quantization, it strongly deviates from a uniform distribution.

In the continuous and deterministic case, the only function that has a uniform distribution (as defined by eq. (2.2)) is a straight line and, specifically, it is the identity function $y(t) = t$. Its differential entropy is zero, as it is the number of parameters needed to describe it, and therefore it is natural to consider it as the simplest function. Any other function is viewed as the composition of itself with the identity function and then we expect it to be more complex, i.e. to have a higher magnitude differential entropy.

From Proposition 2.2.4 we know that the entropy of f equals the Kullback-Leibler divergence between f and the uniform distribution (it suffices to consider $g(t) = t$ in eq. (2.11)):

$$E_f = -D_{KL}(f||U).$$

From the definition of Kullback-Leibler divergence if $f(t) = t$, then $E_f = 0$. Conversely, for Pinsker inequality, if $E_f = 0$ then $f^{*\prime}(t) \equiv 1$ and hence $f(t) = t$.

We conclude that, up to rearrangement, the only function whose range is $[0,1]$ and whose differential entropy is zero is the identity function: any other function has a higher complexity (that is, a lower differential entropy). The higher the magnitude of the differential entropy, the bigger the deviation fo $f(t)$ from the simplest function.

3.1.4 Implementation and numerical results

In this section the algorithm for coefficients selection with ENID will be outlined, and numerical results will be presented, as well as some comparisons with existing methods.

3.1.4.1 Algorithm formulation

The algorithm is structured as follows.
Input: Sequence of discretized values of a function u: $\{u_i\}_{i=1,\dots,M}$
Output: Approximation of u

Algorithm 1

1. Compute the discrete wavelet transform of $\{u_i\}_{i=1,\dots,M}$ and extract the detail coefficients $\{a_j\}_{j\in 1,\dots N}$
2. Sort the absolute values of the detail coefficients in decreasing order
3. For all $j = 1, \dots N$
   (a) Set $\lambda_j = j/N$
   (b) Using eq. (2.7), compute the approximation of $E_1^{\lambda_j}$ using the first j coefficients
   (c) Using eq. (2.7), compute the approximation of $E_2^{\lambda_j}$ using the las $N - j + 1$ coefficients
   (d) Compute ENID(λ_j) according to eq. (3.1)
4. Find the index $\tilde{N}$ that realizes the minimum of ENID. Preserve the first $\tilde{N}$ rearranged coefficients and set to zero the remaining ones
5. Invert the discrete wavelet transform

3.1.4.2 Experimental results

In order to evaluate the performances of ENID, the functions **Bumps**, **Heavy Sine** and **Blocks**, originally defined in [9], were considered, as well as a simple piece-wise polynomial function named **Poly**. The first three functions were provided by the Matlab function **wnoise**, while the latter is defined as follows

$$\text{Poly}(t) = \begin{cases} 0 & \text{if} \quad t \leq 1/8 \\ 8t - 1 & \text{if} \quad 1/8 < t \leq 1/4 \\ 3 - 8t & \text{if} \quad 1/4 < t \leq 3/8 \\ 2(4(t - 3/8) - 1/2)^2 & \text{if} \quad 3/8 < t \leq 5/8 \\ 1 - (6 - 8t)^{1/2} & \text{if} \quad 5/8 < t \leq 3/4 \\ 2(1 - (8t - 5)^{1/2})/3 & \text{if} \quad 3/4 < t \leq 7/8 \\ 0 & \text{if} \quad t > 7/8 \end{cases} . \tag{3.3}$$

The aforementioned functions are shown in fig. 3.3. In order to provide a more real example, two differents rows of Lena and Peppers test images, respectively shown in fig. 3.2(a) and fig. 3.2(b), have been chosen as test funtions. The two rows are shown in fig. 3.3 (e) and (f).

(a) (b)

Figure 3.2: $512 \times 512 \times 8$ bits Lena (a) and Pepper (b) image.

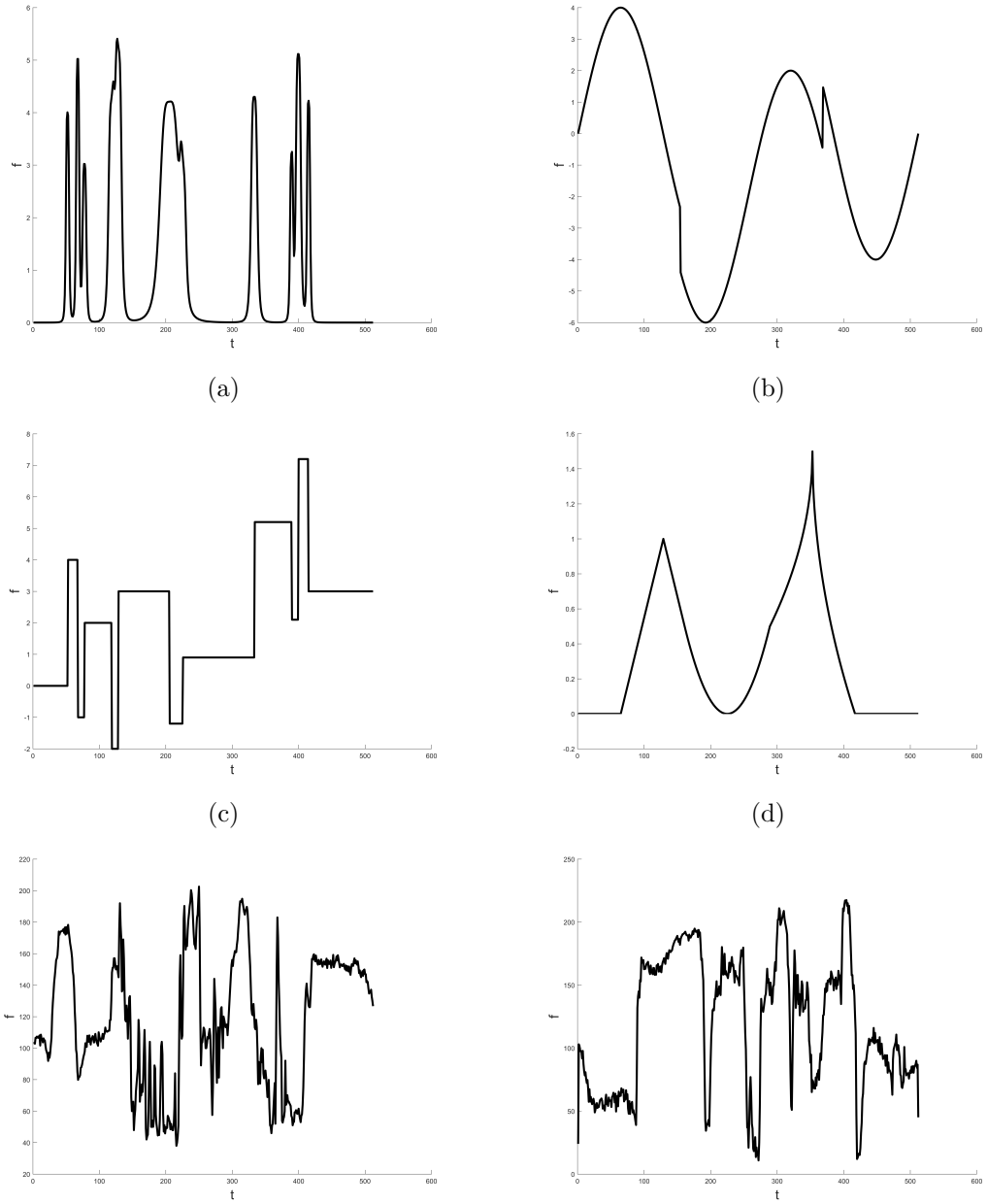


Figure 3.3: Bumps (a), Heavy sine (b), Blocks (c), Poly (d), row nr. 256 of Lena test image (e) and row nr. 256 of Pepper test image (f).

The functions were expanded in wavelet basis and only the detail coefficients have been considered for selection; the whole approximation coefficients have been retained for reconstruction. Numerical tests have been performed using

both stationary wavelet transform (SWT) and dyadic wavelet transform (DWT), 4 decomposition levels have been considered and Daubechies' wavelets of order 2 (db2), 4 (db4) and 12 (db12) and symlet wavelets of order 4 (sym4) and 12 (sym12) have been selected.

The proposed method has been compared with Donoho's universal threshold, matching pursuit algorithm and basis pursuit algorithm using Coifman's entropy cost function [13].

For comparison purposes, wavelet packet decomposition has been chosen for both matching pursuit and basis pursuit. Matching pursuit and Coifman's best basis search are both natively implemented in MatLab, respectively in the functions **wmpalg** and **besttree**. In the case of matching pursuit the maximum relative error ε in ℓ^2 norm must be given as input, and the maximum number of iterations must be set. In the following tests the matching pursuit has been run with $\varepsilon = 10^{-2}$ and $\varepsilon = 10^{-4}$, where the maximum number of iterations has been set equal to 500. In the case of thresholding, DWT has been considered and the parameter σ, which represents the strength of the noise in the denoising interpretation, here stands for the allowed risk taken in reconstruction. It is worth noting that Coifman's basis pursuit does not reduce the number of coefficients, since the leaves of the wavelet packet tree are always equal to the length of the original signal, even if the tree has been pruned by the basis selection procedure. Nevertheless it has been included in the comparative studies to provide best case situation, i.e. a method that should always exhibit perfect reconstruction and hence minimum error.

Mean Squared Error (MSE), Normalized Compression Distance (NCD) and compression ratio have been used as quality metrics. NCD has been computed using Matlab **gzip** function, which implements lossless compression in *.zip format. Although not referenced in the MatLab documentation, experimental evidence showed that **gzip** is not able to recognize identical patterns whose lengths exceeds 512 floating-point samples; hence, the lengths of the test functions have been set equal to 512.

Before showing numerical results, it is interesting to compare ENID with rearranged wavelet coefficients, both depicted in fig. 3.4. Figs. 3.4 (a) and (b) refer to Blocks, and in this case the behaviour of ENID closely matches rearranged coefficients, reaching the minimum (fig. 3.4 (a)) or the constant part (fig. 3.4 (b)) where sorted coefficients plot drops. Fig. 3.4 (c) and (d) refer to Heavysine. In this case, ENID minimum is located where the rearranged coefficients plot become almost constant. Finally, fig. 3.4 (e) and (f) refer to the 256th row of Lena image. Since the signal is less regular, the rearranged coefficients plot decays much slowly and it is difficult to tell to the naked eye where the coefficients become less significant. However, even in this case, ENID reaches its minimum in correspondence to the point where the monotonically decreasing behaviour of wavelet coefficients becomes almost flat.

Figures 3.5–3.10 show some qualitative results in terms of adopted metrics. In particular, it is worth observing that NCD plots present a clear breaking point that separates two distinct different trends, usually switching from a linear trend to a constant or different linear one, as it can be clearly seen in figs. 3.6 and 3.8. This breaking point represents the optimum point that ENID minimum should identify. Tables 3.1–3.6 allow for some comparative studies and the evaluation of the dependency on the wavelet functions. Due to the very low reconstruction errors, we found superfluous to compare approximations plots to

original functions.

Table 3.1 refers to Bumps. As it can be observed, if the considered wavelet has few vanishing moments, the proposed method is slightly worse than matching pursuit; on the contrary, if the wavelet has a high number of vanishing moments, the proposed method outperforms matching pursuit in terms of MSE and NCD, but matching pursuit still achieves a better compression ratio.

The behaviour of ENID on Bumps is illustrated in fig. 3.5 (DWT) and fig 3.6 (SWT). In both cases it is clear that the minimum of ENID corresponds to the point where NCD gets almost constant. The only exception is for wavelet function db12, as shown in fig. 3.6 (d), in which the minimum of ENID is located near NCD knee.

The results concerning Heavysine are reported in table 3.2. In this case, as fig. 3.7 shows, NCD does not get constant, but starts to stabilize on a linear trend long before approaching the location of the minimum of ENID.

A completely different case is represented by Blocks, as shown in fig. 3.8: both MSE and NCD drop abruptly. The behaviour of ENID closely matches NCD's one and in most of the cases the minimum of ENID is in correspondence to the breaking point in NCD. Indeed, the numerical results for Blocks, in table 3.3, show null MSE error and very low NCD error in several cases.

The function Poly exhibits performances similar to Blocks, as it can be seen in fig. 3.9. As 3.4 shows, the matching pursuit algorithm achieves better results in term of MSE, but the proposed method achieves better NCD scores, especially if the wavelet has a high number of vanishing moments.

Finally, fig. 3.10 refers to row no. 256 of Lena image (similar results have been achieved for row no. 256 of Pepper image). The most striking fact is that, since the input signal is not regular, Coifman's best basis procedure does not yield zero error, as reported in tables 3.5 and 3.6. Indeed, in this case matching pursuit almost always reaches its maximum number of iterations, achieving very poor compression ratios. For the NCD depicted in fig. 3.10, it is not immediate to identify a breaking point as in the previous cases; nonetheless, ENID always selects a point that corresponds to a trend change, albeit very small.

To conclude this section, a small remark concerning computational cost. Regarding eq. (2.7), assuming the cost of the logarithm computation is $O(1)$, it is possible to achieve $O(N)$ computational cost by updating the sum instead of computing it at each step. In addition the cost of the rearrangement must be accounted for, leading to a computational cost of $O(N \log_2(N))$. The proposed method is not faster than Donoho's thresholding, which is $O(N)$, but it is completely automatic, where in Donoho's method the threshold must be chosen. On the contrary, computational cost for the proposed method is comparable to matching pursuit's, which is $O(KN \log_2(N))$ in the case of wavelet packet dictionaries, where K is the number of iterations [8]. Even though it is possible that K equals N, it is usually much smaller. In our experiments K has been set equal to 500, which is close to N, but, except for the cases of Lena's and Pepper's row, it remains below 256.

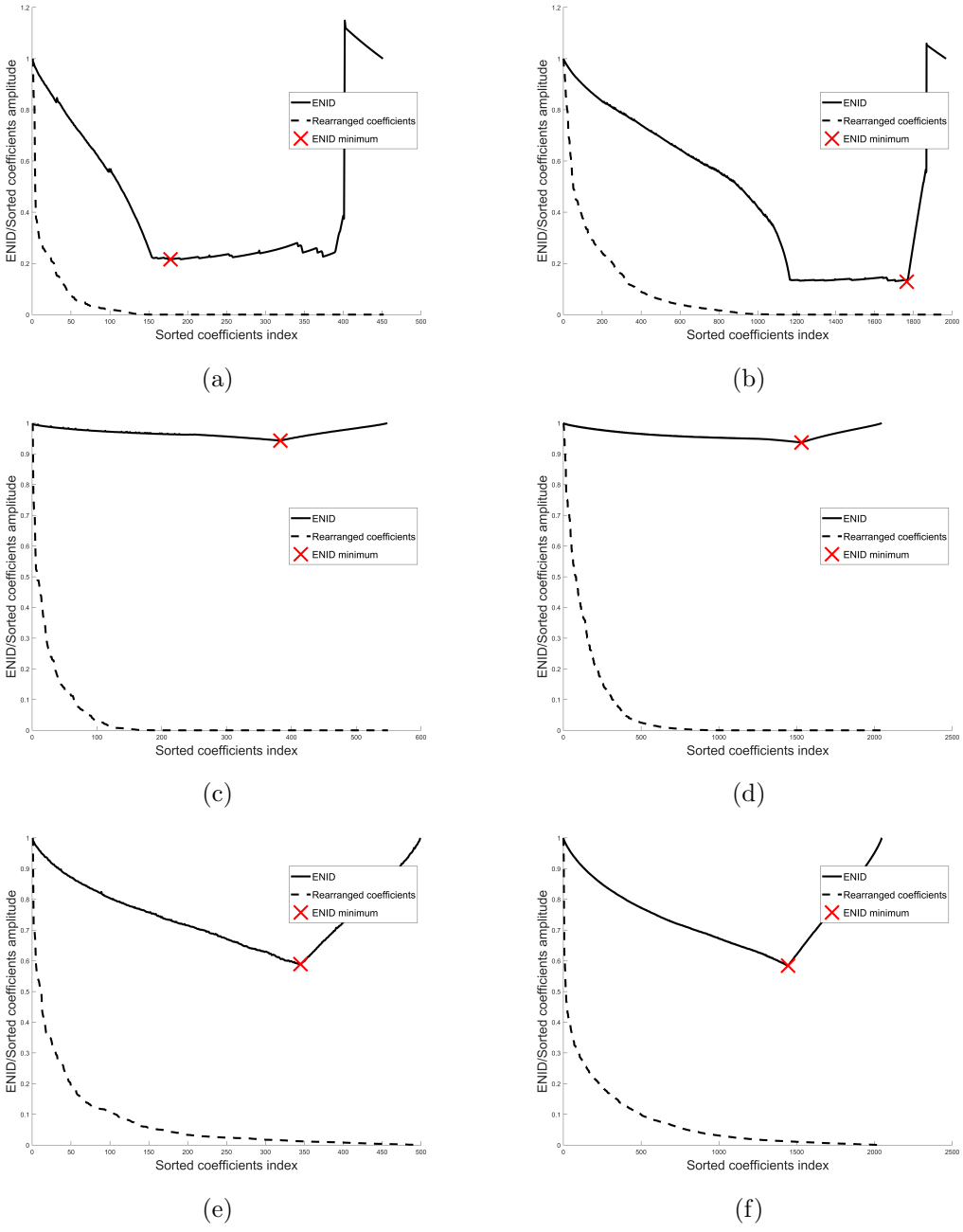


Figure 3.4: ENID and sorted coefficients. Blocks decomposed using DWT (a) and SWT (b) with sym4 as wavelet basis sym4, Heavysine decomposed using DWT (c) and SWT (d) with db12 as wavelet basis, row nr. 256 from image Lena decomposed using DWT (e) and SWT (f) with sym4 as wavelet basis. Sorted coefficients have been normalized for illustration purposes.

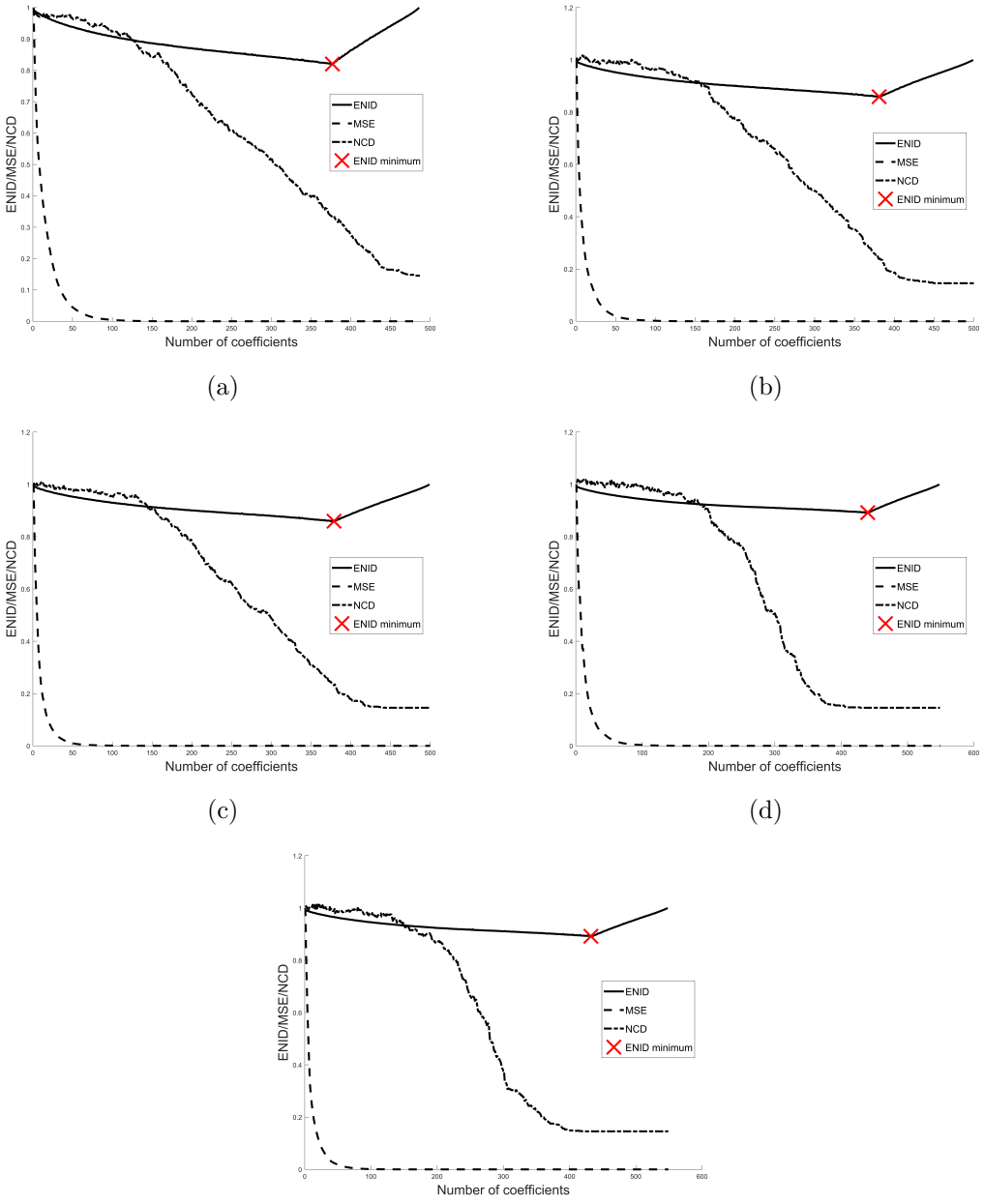


Figure 3.5: Function Bumps decomposed using DWT. ENID, MSE and NCD, for wavelet basis db2 (a), db4 (b), sym4 (c), db12 (d), sym12 (e). The minimum of ENID is marked by a cross. MSE has been normalized for illustration purposes.

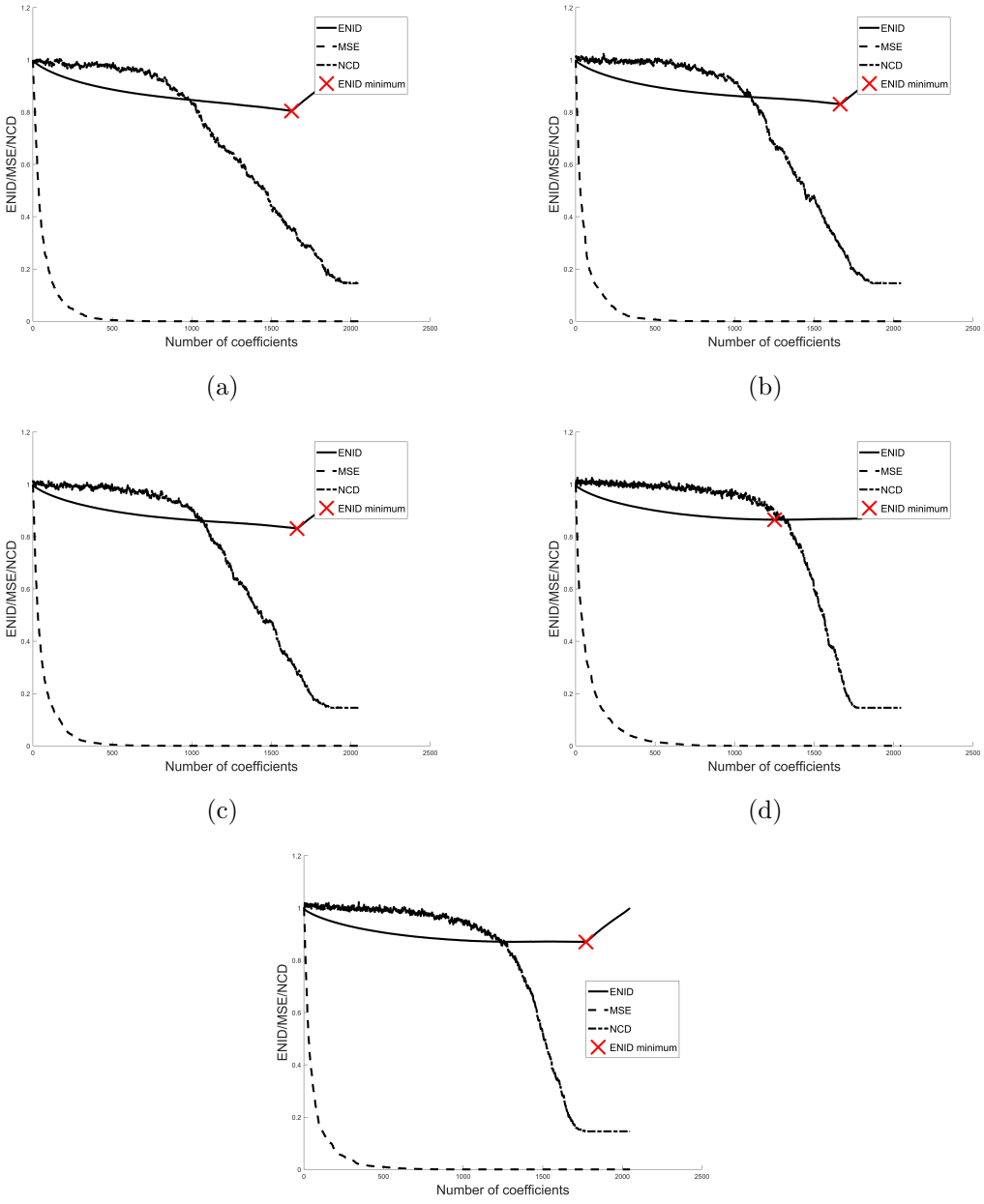


Figure 3.6: Function Bumps decomposed using SWT. ENID, MSE and NCD, in wavelet basis db2 (a), db4 (b), sym4 (c), db12 (d), sym12 (e). The minimum of ENID is marked by a cross. MSE has been normalized for illustration purposes.

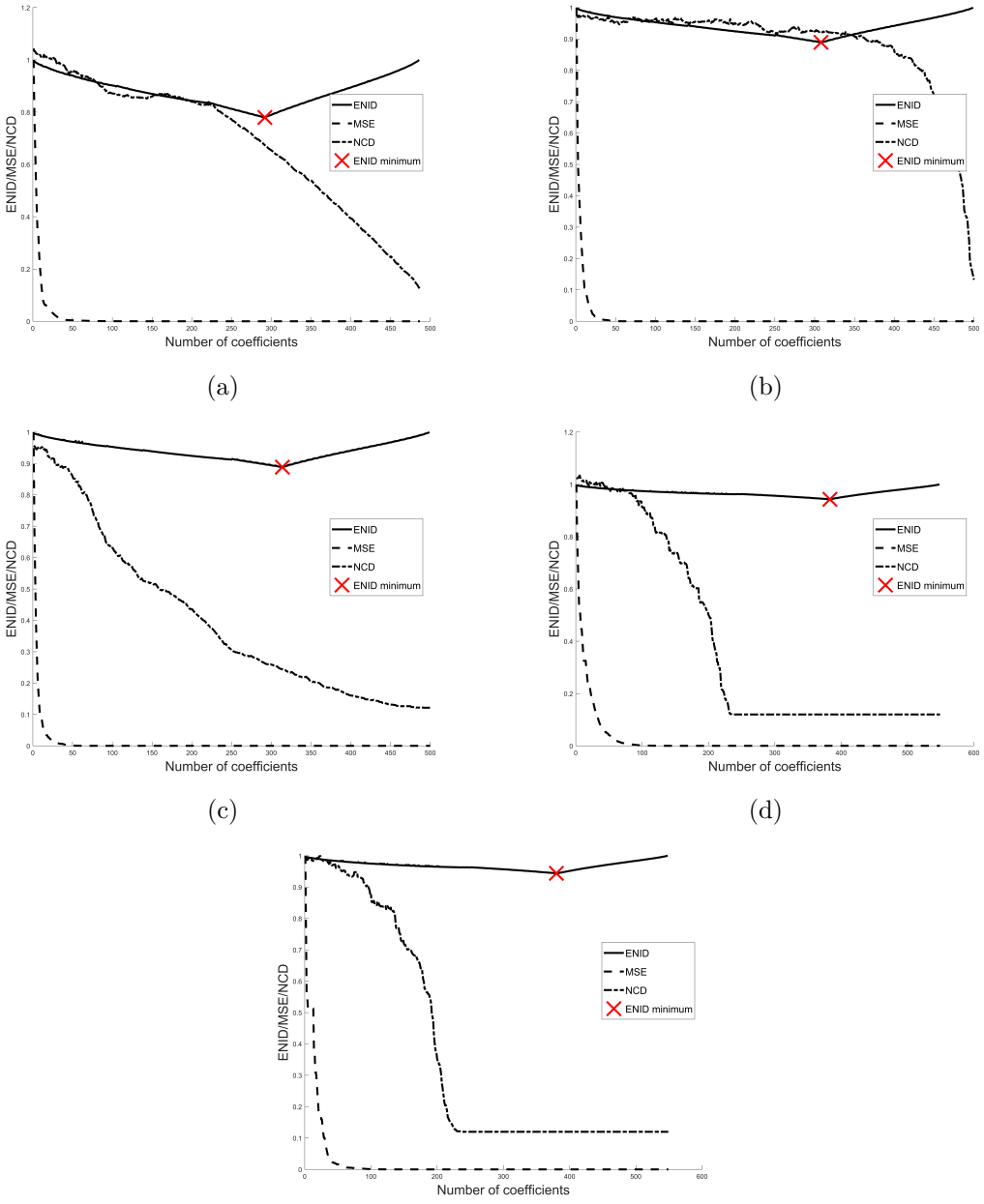


Figure 3.7: Function Heavy sine decomposed using DWT. ENID, MSE and NCD, in wavelet basis db2 (a), db4 (b), sym4 (c), db12 (d), sym12 (e). The minimum of ENID is marked by a cross. MSE has been normalized for illustration purposes.

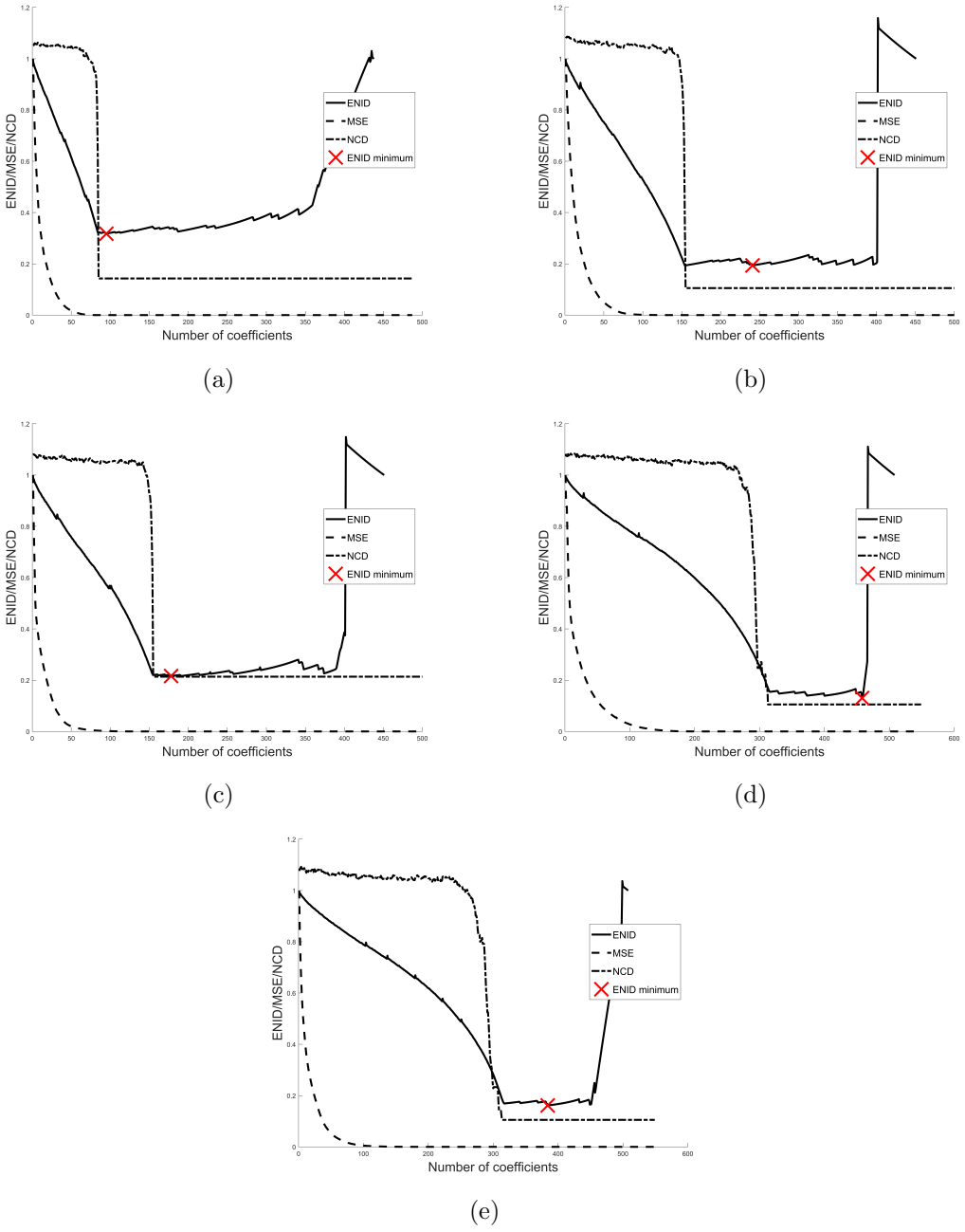


Figure 3.8: Function Blocks decomposed using DWT. ENID, MSE and NCD, in wavelet basis db2 (a), db4 (b), sym4 (c), db12 (d), sym12 (e). The minimum of ENID is marked by a cross. MSE has been normalized for illustration purposes.

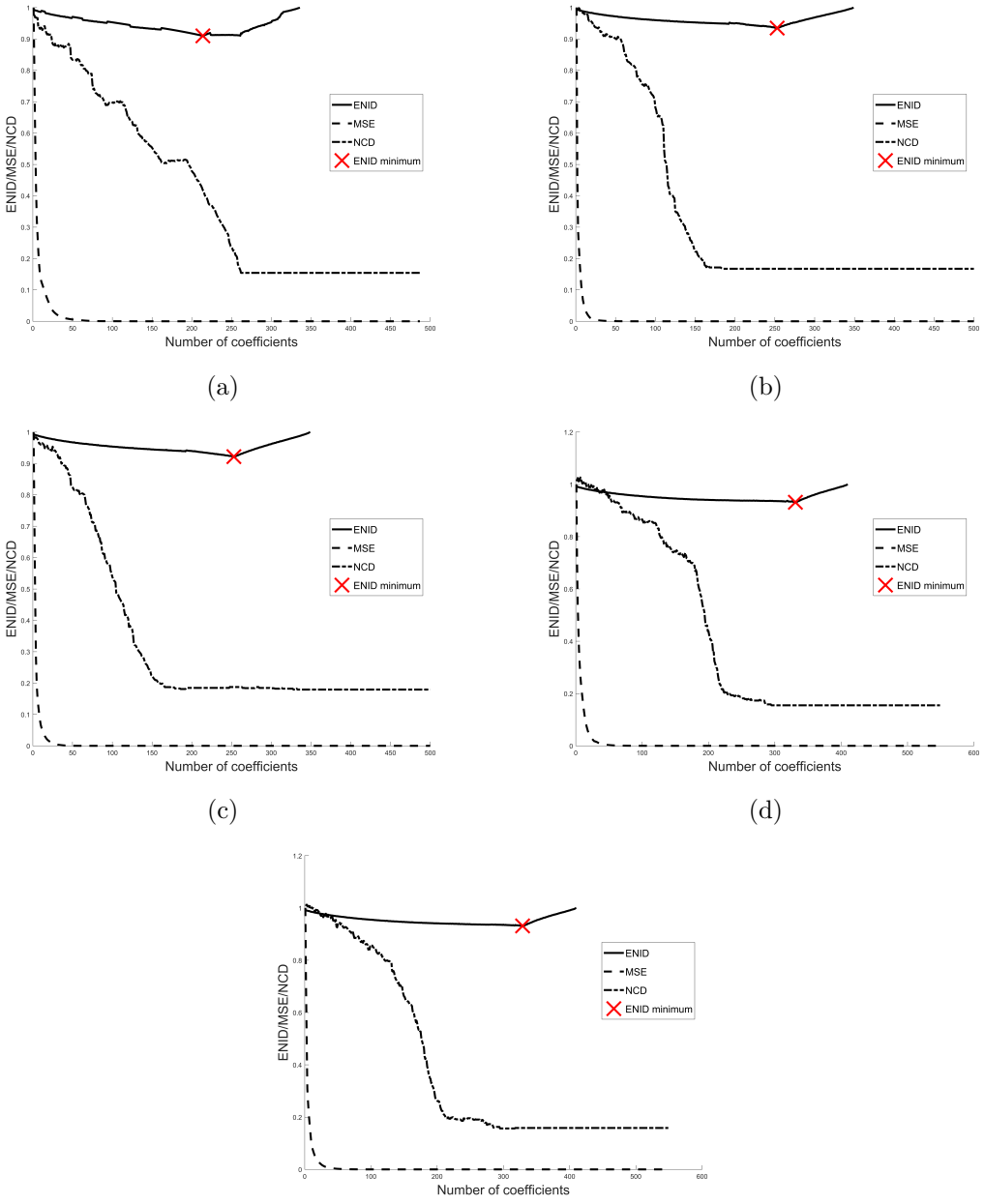


Figure 3.9: Function Poly decomposed using DWT. ENID, MSE and NCD, in wavelet basis db2 (a), db4 (b), sym4 (c), db12 (d), sym12 (e). The minimum of ENID is marked by a cross. MSE has been normalized for illustration purposes.

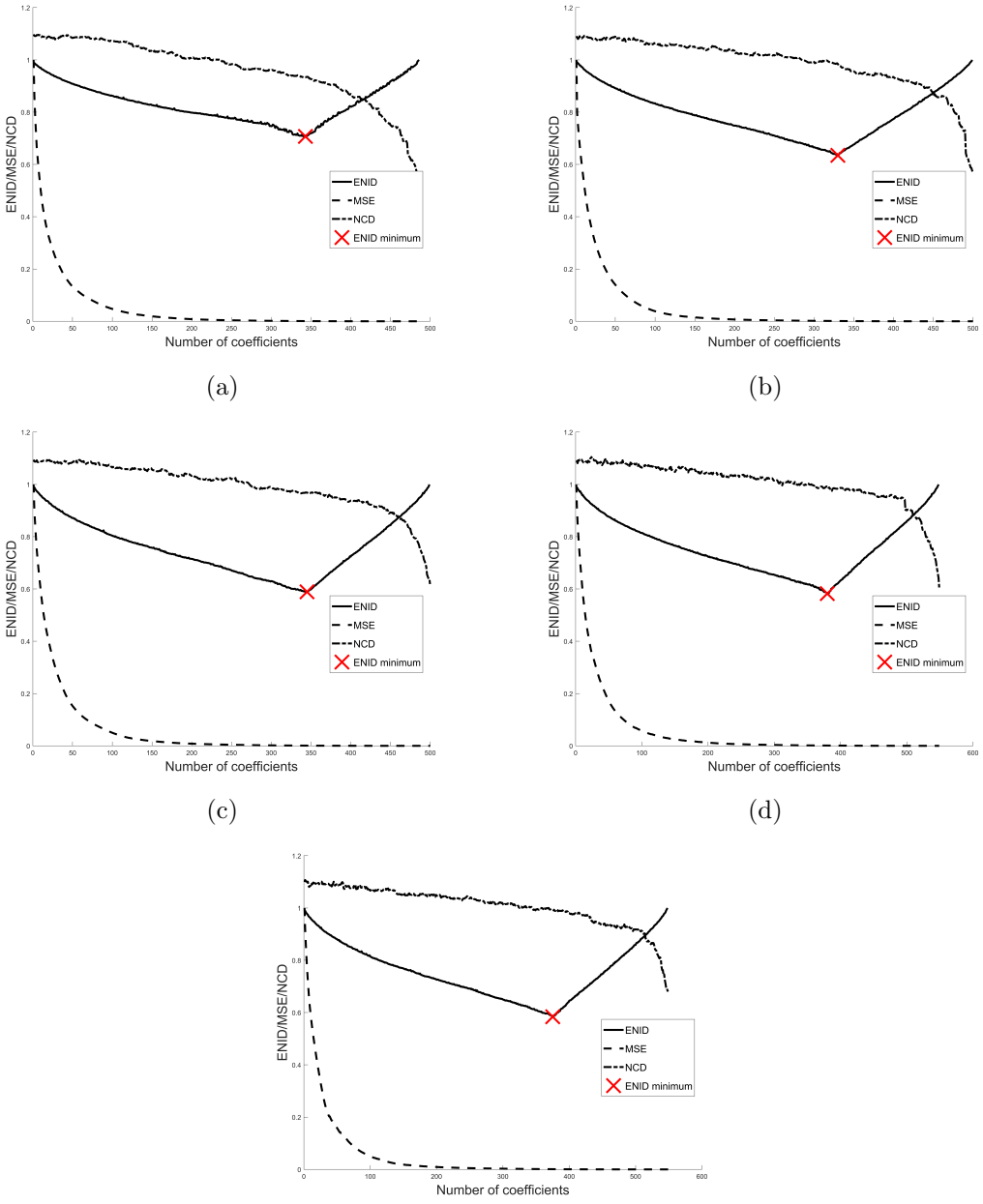


Figure 3.10: Row nr. 256 from image Lena.bmp, decomposed using DWT. ENID, MSE and NCD, in wavelet basis db2 (a), db4 (b), sym4 (c), db12 (d), sym12 (e). The minimum of ENID is marked by a cross. MSE has been normalized for illustration purposes.

bumps

method	MSE	NCD	$\tilde{N}$	N	ratio
DWT ENID					
db2	2.70676×10^{-11}	0.33702532	377	521	0.774127
db4	1.127×10^{-13}	0.23802001	381	538	0.762000
sym4	1.631×10^{-13}	0.23154008	379	538	0.758000
db12	0	0.14556962	440	602	0.801457
sym12	0	0.14556962	432	602	0.786885
SWT ENID					
db2	3.30726×10^{-11}	0.34862869	1628	2048	0.522461
db4	5.451×10^{-13}	0.28586498	1664	2048	0.531250
sym4	6.004×10^{-13}	0.27584388	1662	2048	0.530762
db12	$1.8985996390 \times 10^{-6}$	0.88751290	1252	2048	0.430664
sym12	0	0.14556962	1771	2048	0.557373
Matching Pursuit, $\varepsilon = 10^{-2}$					
db2,	3.17×10^{-14}	0.19883966	463	512	0.904297
db4,	2.77×10^{-14}	0.19620253	423	512	0.826172
sym4,	3.01×10^{-14}	0.19831224	421	512	0.822266
db12,	2.83×10^{-14}	0.18411362	368	512	0.718750
sym12,	3.09×10^{-14}	0.19568648	366	512	0.714844
Matching Pursuit, $\varepsilon = 10^{-4}$					
db2	3.17×10^{-14}	0.19883966	463	512	0.904297
db4	2.77×10^{-14}	0.19620253	423	512	0.826172
sym4	3.01×10^{-14}	0.19831224	421	512	0.822266
db12	2.83×10^{-14}	0.18411362	368	512	0.718750
sym12	3.09×10^{-14}	0.19568648	366	512	0.714844
Basis Pursuit, Entropy cost function					
db2	0	0.14556962	512	512	1.000000
db4	0	0.14556962	512	512	1.000000
sym4	0	0.14556962	512	512	1.000000
db12	0	0.14556962	512	512	1.000000
sym12	0	0.14556962	512	512	1.000000
Donoho Threshold, $\sigma = 10^{-2}$					
db2	$5.00187274816 \times 10^{-5}$	0.82235108	164	521	0.336756
db4	$3.69351005305 \times 10^{-5}$	0.91322530	155	538	0.310000
sym4	$3.95281050496 \times 10^{-5}$	0.90615224	146	538	0.292000
db12	$3.67218228157 \times 10^{-5}$	0.93397177	170	602	0.309654
sym12	$4.26536272669 \times 10^{-5}$	0.94656489	145	602	0.264117

Table 3.1: Function Bumps: numerical results. Here N and $\tilde{N}$ stand, respectively, for the total and for the selected number of coefficients.

heavy sine

method	MSE	NCD	$\tilde{N}$	N	compression ratio
DWT ENID					
db2	$2.843989180 \times 10^{-7}$	0.67355372	292	521	0.599589
db4	8.17×10^{-14}	0.92533333	308	538	0.616000
sym4	7.69×10^{-14}	0.24363992	314	538	0.628000
db12	0	0.11989529	383	602	0.697632
sym12	0	0.11989529	380	602	0.692168
SWT ENID					
db2	8.2339381×10^{-9}	0.45218244	1436	2048	0.475586
db4	5.20×10^{-14}	0.17393535	1441	2048	0.476807
sym4	4.45×10^{-14}	0.18374741	1450	2048	0.479004
db12	0	0.11989529	1531	2048	0.498779
sym12	0	0.11989529	1508	2048	0.493164
Matching Pursuit, $\varepsilon = 10^{-2}$					
db2,	1.315345×10^{-10}	0.16761217	500	512	0.976563
db4,	9.47×10^{-14}	0.26223946	310	512	0.605469
sym4,	9.46×10^{-14}	0.25812712	312	512	0.609375
db12,	8.81×10^{-14}	0.18402427	186	512	0.363281
sym12,	9.43×10^{-14}	0.20110055	178	512	0.347656
Matching Pursuit, $\varepsilon = 10^{-4}$					
db2	1.315345×10^{-10}	0.16761217	500	512	0.976563
db4	9.47×10^{-14}	0.26223946	310	512	0.605469
sym4	9.46×10^{-14}	0.25812712	312	512	0.609375
db12	8.81×10^{-14}	0.18402427	186	512	0.363281
sym12	9.43×10^{-14}	0.20110055	178	512	0.347656
Basis Pursuit, Entropy cost function					
db2	0	0.11989529	512	512	1.000000
db4	0	0.11989529	512	512	1.000000
sym4	0	0.12146597	512	512	1.000000
db12	0	0.11989529	512	512	1.000000
sym12	0	0.11989529	512	512	1.000000
Donoho Threshold, $\sigma = 10^{-2}$					
db2	$3.51985710987 \times 10^{-5}$	0.91442652	79	521	0.082136
db4	$8.5603459941 \times 10^{-6}$	0.86119734	51	538	0.102000
sym4	$1.15627777225 \times 10^{-5}$	0.85135737	51	538	0.102000
db12	$1.87709331753 \times 10^{-5}$	0.88571429	110	602	0.200364
sym12	$1.92629878363 \times 10^{-5}$	0.85162150	101	602	0.183971

Table 3.2: Function Heavy sine: numerical results. Here N and $\tilde{N}$ stand, respectively, for the total and for the selected number of coefficients.

method	MSE	blocks NCD	$\tilde{N}$	N	compression ratio
DWT ENID					
db2	0	0.14285714	95	521	0.195072
db4	8.17×10^{-14}	0.92533333	308	538	0.616000
sym4	0	0.21359223	178	538	0.356000
db12	0	0.10526316	458	602	0.834244
sym12	0	0.10526316	384	602	0.699454
SWT ENID					
db2	0	0.14141414	858	2048	0.334473
db4	0	0.10526316	1689	2048	0.537354
sym4	0	0.21568627	1766	2048	0.556152
db12	0	0.10526316	1803	2048	0.565186
sym12	0	0.10526316	1770	2048	0.557129
Matching Pursuit, $\varepsilon = 10^{-2}$					
db2	0	0.23636364	120	512	0.234375
db4	0	0.31451613	201	512	0.392578
sym4	0	0.31092437	201	512	0.392578
db12	7.14×10^{-14}	0.46938776	341	512	0.666016
sym12	5.98×10^{-14}	0.52727273	334	512	0.652344
Matching Pursuit, $\varepsilon = 10^{-4}$					
db2	0	0.23636364	120	512	0.234375
db4	0	0.31451613	201	512	0.392578
sym4	0	0.31092437	201	512	0.392578
db12	7.14×10^{-14}	0.46938776	341	512	0.666016
sym12	5.98×10^{-14}	0.52727273	334	512	0.652344
Basis Pursuit, Entropy cost function					
db2	0	0.14851485	512	512	1.000000
db4	0	0.19230769	512	512	1.000000
sym4	0	0.24074074	512	512	1.000000
db12	0	0.14432990	512	512	1.000000
sym12	0	0.22330097	512	512	1.000000
Donoho Threshold, $\sigma = 10^{-2}$					
db2	$1.0955317647 \times 10^{-6}$	0.61797753	83	521	0.170431
db4	0.0000140847160919	1.04502974	136	538	0.272000
sym4	$1.15627777225 \times 10^{-5}$	0.85135737	130	538	0.260000
db12	$1.84362927559 \times 10^{-5}$	1.05063291	207	602	0.377049
sym12	$2.42945737869 \times 10^{-5}$	1.04580607	188	602	0.342441

Table 3.3: Function Blocks: numerical results. Here N and $\tilde{N}$ stand, respectively, for the total and for the selected number of coefficients.

method	MSE	poly NCD	$\tilde{N}$	N	compression ratio
DWT ENID					
db2	1.0085577×10^{-9}	0.41747573	214	521	0.439425
db4	8.17×10^{-14}	0.92533333	308	538	0.616000
sym4	7.69×10^{-14}	0.24363992	314	538	0.628000
db12	0	0.15492958	331	602	0.602914
sym12	0	0.15849387	329	602	0.599271
SWT ENID					
db2	2.03952×10^{-11}	0.31894484	1059	2048	0.383545
db4	0	0.16041848	1272	2048	0.435547
sym4	0	0.17493473	1257	2048	0.431885
db12	6.75317×10^{-11}	0.67239102	1111	2048	0.396240
sym12	1.4050×10^{-12}	0.43431221	1172	2048	0.411133
Matching Pursuit, $\varepsilon = 10^{-2}$					
db2	0	0.16550523	285	512	0.556641
db4	2.2×10^{-15}	0.20399306	189	512	0.369141
sym4	2.2×10^{-15}	0.20701454	189	512	0.369141
db12	2.1×10^{-15}	0.20936170	255	512	0.498047
sym12	2.0×10^{-15}	0.20357751	243	512	0.474609
Matching Pursuit, $\varepsilon = 10^{-4}$					
db2	0	0.16550523	285	512	0.556641
db4	2.2×10^{-15}	0.20399306	189	512	0.369141
sym4	2.2×10^{-15}	0.20701454	189	512	0.369141
db12	2.1×10^{-15}	0.20936170	255	512	0.498047
sym12	2.0×10^{-15}	0.20357751	243	512	0.474609
Basis Pursuit, Entropy cost function					
db2	0	0.15283843	512	512	1.000000
db4	0	0.16232639	512	512	1.000000
sym4	0	0.17458946	512	512	1.000000
db12	0	0.15968586	512	512	1.000000
sym12	0	0.16797215	512	512	1.000000
Donoho Threshold, $\sigma = 10^{-2}$					
db2	$2.53009043443 \times 10^{-5}$	0.93621134	9	521	0.018480
db4	$1.20045866224 \times 10^{-5}$	0.96110211	15	538	0.030000
sym4	$1.87855132736 \times 10^{-5}$	0.94880174	14	538	0.028000
db12	$2.27763969526 \times 10^{-5}$	0.99106674	20	602	0.036430
sym12	$2.12478553095 \times 10^{-5}$	0.98794549	15	602	0.027322

Table 3.4: Function Poly: numerical results. Here N and $\tilde{N}$ stand, respectively, for the total and for the selected number of coefficients.

lena, row 256

method	MSE	NCD	$\tilde{N}$	N	compression ratio
DWT ENID					
db2	0.2156460285186768	0.93314425	343	521	0.704312
db4	0.2633713483810425	0.98438662	330	538	0.660000
sym4	0.1673672348260880	0.96792453	345	538	0.690000
db12	0.2432090938091278	0.99144027	380	602	0.692168
sym12	0.2152678966522217	0.99292628	375	602	0.683060
SWT ENID					
db2	0.1003059070127614	0.97454407	1607	2048	0.517334
db4	0.2739858613136300	0.99152230	1454	2048	0.479980
sym4	0.3139664331285880	0.99664929	1445	2048	0.477783
db12	0.2696727245520483	0.98151571	1481	2048	0.486572
sym12	0.2532048743354862	0.99037037	1465	2048	0.482666
Matching Pursuit, $\varepsilon = 10^{-2}$					
db2	0.0000637789170411	0.51706037	500	512	0.976563
db4	2.2×10^{-15}	0.20399306	189	512	0.369141
sym4	2.2×10^{-15}	0.20701454	189	512	0.369141
db12	$3.72049982833 \times 10^{-5}$	0.77294118	500	512	0.976563
sym12	$2.168963182945 \times 10^{-4}$	0.86951814	500	512	0.976563
Matching Pursuit, $\varepsilon = 10^{-4}$					
db2	$6.37789170411 \times 10^{-5}$	0.51706037	500	512	0.976563
db4	$6.25276274551 \times 10^{-5}$	0.59328649	500	512	0.976563
sym4	$1.488888005312 \times 10^{-4}$	0.69251887	500	512	0.976563
db12	$3.72049982833 \times 10^{-5}$	0.77294118	500	512	0.976563
sym12	$2.168963182945 \times 10^{-4}$	0.86951814	500	512	0.976563
Basis Pursuit, Entropy cost function					
db2	6.77351×10^{-11}	0.54187545	512	512	1.000000
db4	1.003098×10^{-10}	0.54797441	512	512	1.000000
sym4	2.232684×10^{-10}	0.58256275	512	512	1.000000
db12	1.241587×10^{-10}	0.57988981	512	512	1.000000
sym12	4.171552×10^{-10}	0.67707006	512	512	1.000000
Donoho Threshold, $\sigma = 10^{-2}$					
db2	$8.817740991 \times 10^{-7}$	0.56959459	482	521	0.989733
db4	$1.317942946 \times 10^{-7}$	0.57592722	498	538	0.996000
sym4	$7.640361446 \times 10^{-7}$	0.63839286	498	538	0.996000
db12	$4.6369686970 \times 10^{-6}$	0.70678466	543	602	0.989071
sym12	$1.3792741811 \times 10^{-6}$	0.71073378	544	602	0.990893

Table 3.5: Row nr. 256 from image Lena.bmp: numerical results. Here N and $\tilde{N}$ stand, respectively, for the total and for the selected number of coefficients.

peppers, row 256

method	MSE	NCD	$\tilde{N}$	N	compression ratio
DWT ENID					
db2	0.4838123321533203	0.98183224	332	521	0.681725
db4	0.6010890007019043	1.01890990	335	538	0.670000
sym4	0.4522691369056702	1.00631501	341	538	0.682000
db12	0.5208664536476135	1.02250920	382	602	0.695811
sym12	0.7736942768096924	1.02958580	363	602	0.661202
SWT ENID					
db2	0.2300025783155537	0.99335057	1547	2048	0.502686
db4	0.3891914271080494	1.00892193	1464	2048	0.482422
sym4	0.3568777709917778	1.00815419	1441	2048	0.476807
db12	0.5747633061547710	1.02524128	1483	2048	0.487061
sym12	0.6056243619198243	1.01552680	1446	2048	0.478027
Matching Pursuit, $\varepsilon = 10^{-2}$					
db2	$2.495201450555 \times 10^{-4}$	0.62122386	500	512	0.976563
db4	$2.695755256539 \times 10^{-4}$	0.77592166	500	512	0.976563
sym4	$7.09654443748 \times 10^{-5}$	0.67453505	500	512	0.976563
db12	$1.994069236916 \times 10^{-4}$	0.83225458	500	512	0.976563
sym12	$2.541948466722 \times 10^{-4}$	0.78615299	500	512	0.976563
Matching Pursuit, $\varepsilon = 10^{-4}$					
db2	$2.495201450555 \times 10^{-4}$	0.62122386	500	512	0.976563
db4	$2.695755256539 \times 10^{-4}$	0.77592166	500	512	0.976563
sym4	$7.09654443748 \times 10^{-5}$	0.67453505	500	512	0.976563
db12	$1.994069236916 \times 10^{-4}$	0.83225458	500	512	0.976563
sym12	$2.541948466722 \times 10^{-4}$	0.78615299	500	512	0.976563
Basis Pursuit, Entropy cost function					
db2	7.97793×10^{-11}	0.55726257	512	512	1.000000
db4	1.017099×10^{-10}	0.57771469	512	512	1.000000
sym4	2.412758×10^{-10}	0.61481481	512	512	1.000000
db12	1.435862×10^{-10}	0.62551160	512	512	1.000000
sym12	4.456918×10^{-10}	0.69516028	512	512	1.000000
Donoho Threshold, $\sigma = 10^{-2}$					
db2	$1.574077686 \times 10^{-7}$	0.62284928	486	521	0.997947
db4	2.3918947×10^{-9}	0.59541470	498	538	0.996000
sym4	$2.6939262625 \times 10^{-6}$	0.63795337	498	538	0.996000
db12	$8.527463251 \times 10^{-7}$	0.67507886	547	602	0.996357
sym12	$1.470099278 \times 10^{-7}$	0.73106738	547	602	0.996357

Table 3.6: Row nr. 256 from image Pepper.bmp: numerical results. Here N and $\tilde{N}$ stand, respectively, for the total and for the selected number of coefficients.

3.2 Kullback-Leibler based information distances

In the previous Section 3.1, ENID has been proposed as a replacement of NID in the task of choosing the optimal set of projection coefficients. In particular,

ENID approximates NCD by replacing the length of the compressed data with their average length (λE_1^λ and $(1-\lambda)E_2^\lambda$), where differential entropy is considered instead of Shannon entropy. This choice avoids the quantization step required in the computation of Shannon entropy and relies on the fact that, under suitable normalization hypotheses, differential entropy is a complexity measure. In this section this approach is expanded, providing an estimation of NID that uses Kullback-Leibler divergence. An interpretation of this information measure is given and a proper model to study its applicability is developed.

Let us recall the definition of NID. Given two strings a and b, the Normalized Information Distance between them is defined as

$$\mathrm{NID}(a,b) = \frac{\max\{\mathbb{K}(a|b), \mathbb{K}(b|a)\}}{\max\{\mathbb{K}(a), \mathbb{K}(b)\}},$$

where $\mathbb{K}(a)$ is the Kolmogorov complexity of a and $\mathbb{K}(a|b)$ is the conditional Kolmogorov complexity of a given b, that is the length of the minimum program that outputs a given b as an input. Conditional Kolmogorov complexity $\mathbb{K}(a|b)$ between a and b measures how much information in b can be used to describe a. In [124] a variation of conditional Kolmogorov complexity, called algorithmic relative complexity has been proposed, inspired by Kullback-Leibler divergence.

Kullback-Leibler divergence can be rewritten as the difference of two terms:

$$D_{KL}(X||Y) = E_{X,Y} - E_X;$$

the first term is the cross-entropy $E_{X,Y} = -\int p_X \ln(p_Y) dy$ between X and Y and the second is the entropy of X. In the discrete case, the cross-entropy measures how many bits per symbol are necessary to encode X assuming it is distributed as Y. The difference between it and the entropy of X is the amount of exceeding bits used to encode X, mistakenly assuming it is distributed as Y.

Similarly, algorithmic relative complexity between a and b is defined as the algorithmic cross complexity $\mathbb{K}(a \oplus b)$ between a and b minus the Kolmogorov complexity of a:

$$\mathbb{K}(a||b) = \mathbb{K}(a \oplus b) - \mathbb{K}(a).$$

Algorithmic cross complexity is defined as the length of the minimum program that outputs a by reusing parts of the *code of the minimum program that outputs* b. This definition mimics the definition of cross entropy: what determines the discrepancy between a and b is how much the code (or the encoding, when referred to cross entropy) developed for b is capable of encoding a.

In [124] algorithmic relative complexity has been considered for classification tasks, proving itself competitive with NCD. Being based on Kolmogorov complexity, algorithmic relative complexity is non-computable; an approximation procedure has been proposed, where the process of compressing a using a dictionary developed for b is implemented. It should be noted that a similar concept has been proposed in [125], where a compression-based approximation of Kullback-Leibler divergence has been used in language recognition and classification tasks. The same approach has been implemented successfully in [126],

where it has been proposed to use cross-entropy in place of conditional Kolmogorov complexity. Thanks to the approach proposed in this thesis, a direct approximation of Kullback-Leibler divergence in time domain can be formulated, avoiding compression-based approximations.

Our proposal is to use Kullback-Leibler divergence in place of conditional Kolmogorov complexity. In addition to the relation between Kullback-Leibler divergence and coding error for discrete codes, this choice is justified by its interpretation as the entropy of the composition of two functions. Indeed, as a consequence to the fact that $E_{g^{-1}\circ f} = -D(f||g)$, and because of Pinsker inequality, it follows that the Kullback-Leibler divergence also measures the difference between two functions in the sense of composition: if the composition of two functions is the identity, then the Kullback-Leibler divergence is zero; conversely, if their composition is not the identity, then the Kullback-Leibler divergence is not zero.

Making a parallel with the discrete setting, $g^{-1} \circ f$ is the "interpretation" of the code given by f according to g. If the "interpretation" of the code is perfect, than the composition equals the identity and its entropy is zero.

In our framework we limit ourselves to functions defined from the unit interval to the unit interval.

There are two information distances that can be derived from Kullback-Leibler divergence that we propose. The first one is inspired by the un-normalized form of NID, as proposed by Vitanyì et al. in [122], and it consists of the maximum between $D_{KL}(f||g)$ and $D_{KL}(g||f)$.

Definition 3.2.1 (KLID)**.** *Let $f, g : [0,1] \to [0,1]$ be measurable, monotone functions. Then the **Kullback-Leibler Information Distance** (KLID) between f and g is defined as follows.*

$$\mathrm{KLID}(f,g) = \max\{D_{KL}(f||g), D_{KL}(g||f)\}.$$

The choice of an un-normalized distance is due to the lack of a meaningful upper bound for Kullback-Leibler divergence in terms of entropies.

The second information distance we propose is derived from the Jeffrey divergence, which is the symmetrization of Kullback-Leibler divergence obtained by adding together $D_{KL}(f||g)$ and $D_{KL}(g||f)$.

Definition 3.2.2 (JID)**.** *Let $f, g : [0,1] \to [0,1]$ be measurable, monotone functions. Then the **Jeffrey Information Distance** (JID) between f and g is defined as follows.*

$$\mathrm{JID}(f,g) = D_{KL}(f||g) + D_{KL}(g||f).$$

We will show that both information distances, similarly to ENID, can be used to automatically select projection coefficients in a basis.

3.3 Approximation of Kullback-Leibler divergence

Let $f, g : [0,1] \to [0,1]$ be two strictly monotone functions. As seen in Chapter 2, the functional form of Kullback-Leibler divergence is itself complicated to

manage (both analitically and numerically), due to the presence of the inverse of g:

$$D_{KL}(f||g) = \int_0^1 \ln\left(\frac{g'(g^{-1}(f(t)))}{f'(t)}\right) dt.$$

Nonetheless, it is possible to provide an approximation for $D_{KL}(f||g)$.

In the following we will denote the integral mean value of a function $f : \Omega \to \mathbb{R}$ using the expectation notation:

$$\mathbb{E}[f] = \frac{1}{|\Omega|}\int_\Omega f(t)dt.$$

Proposition 3.3.1. *Let $f(t), g(t) : [0,1] \to [0,1]$ be strictly increasing monotone functions and assume $C_g = \sup_{s\in[0,1]} |g''(s)|/|g'(s)|^2$, $C_g < +\infty$. Then*

$$D_{KL}(f||g) = E_g - E_f + \int_0^1 \int_{g(t)}^{f(t)} \frac{g''(g^{-1}(y))}{|g'(g^{-1}(y))|^2} dydt$$

and

$$D_{KL}(f||g) = |E_g - E_f| + R_g|\mathbb{E}[f-g]|,$$

with

$$|R_g| \leq C_g.$$

Proof.
Recall that

$$\begin{aligned} D_{KL}(f||g) &= \int_0^1 \ln\left(\frac{g'(g^{-1}(f(t)))}{f'(t)}\right) dt \\ &= E_{f,g} - E_f \end{aligned}$$

and let us focus on the cross-entropy term

$$E_{f,g} = \int_0^1 \ln(g'(g^{-1}(f(t))))dt.$$

Considering the first order Taylor expansion of $\ln(g'(g^{-1}(y + \Delta y)))$ with integral remainder, we get

$$\ln(g'(g^{-1}(y+\Delta y))) = \ln(g'(g^{-1}(y))) + \int_y^{y+\Delta y} \frac{g''(g^{-1}(y'))}{|g'(g^{-1}(y'))|^2} dy'.$$

Set $y = g(y)$ and $\Delta y = f(t) - g(t)$ and integrate with respect to t. Then, adding and subtracting $g(t)$,

$$\begin{aligned}
E_{f,g} &= \int_0^1 \ln\left(g'(g^{-1}(f(t)))\right) dt \\
&= \int_0^1 \ln\left(g'(g^{-1}(g(t)+f(t)-g(t)))\right) dt \\
&= \int_0^1 \left[\ln\left(g'(t)\right) + \int_{g(t)}^{f(t)} \frac{g''(g^{-1}(y))}{|g'(g^{-1}(y))|^2} dy\right] dt \\
&= E_g + \int_0^1 \int_{g(t)}^{f(t)} \frac{g''(g^{-1}(y))}{|g'(g^{-1}(y))|^2} dydt.
\end{aligned}$$

As a consequence,

$$D_{KL}(f||g) = E_g - E_f + \int_0^1 \int_{g(t)}^{f(t)} \frac{g''(g^{-1}(y))}{|g'(g^{-1}(y))|^2} dydt.$$

Note that

$$-C_g \int_{g(t)}^{f(t)} dy \leq \int_{g(t)}^{f(t)} \frac{g''(g^{-1}(y))}{|g'(g^{-1}(y))|^2} dy \leq C_g \int_{g(t)}^{f(t)} dy$$

and hence

$$-C_g \left|\int_0^1 \int_{g(t)}^{f(t)} dydt\right| \leq D_{KL}(f||g) - |E_g - E_f| \leq C_g \left|\int_0^1 \int_{g(t)}^{f(t)} dydt\right|.$$

As a result,

$$|D_{KL}(f||g) - |E_g - E_f|| \leq C_g \left|\mathbb{E}[f-g]\right|.$$

□

The quality of this approximation depends on the constant C_g: as an example, consider the function

$$g(t) = \frac{(1+\beta t)^{-\gamma} - (1+\beta)^{-\gamma}}{1-(1+\beta)^{-\gamma}},$$

with $\beta, \gamma > 0$.

Then,

$$\begin{aligned}
\left|\frac{g''(t)}{|g'(t)|^2}\right| &= \left|1-\frac{1}{\gamma}\right| \left[1-(1+\beta)^{-\gamma}\right] (1+\beta t)^{-\gamma} \\
&\leq \left|1-\frac{1}{\gamma}\right| \left((1+\beta)^{\gamma}-1\right). \\
&= C_g.
\end{aligned}$$

If the constants C_g and C_f are finite, the proposed approximation for the Kullback-Leibler divergence can be used to approximate the Kullback-Leibler Information Distance :

$$\begin{aligned}\mathrm{KLID}(f,g) &= \max\{D_{KL}(f||g), D_{KL}(g||f)\} \\ &= \max\{|E_f - E_g| + R_g|\mathbb{E}[f-g]|, |E_g - E_f| + R_f|\mathbb{E}[g-f]|\} \\ &= |E_f - E_g| + \max\{R_g, R_f\}|\mathbb{E}[f-g]| \\ &= |E_f - E_g| + R\,|\mathbb{E}[f-g]|\,, \end{aligned} \tag{3.4}$$

where $|R| \leq \max\{C_g, C_f\}$.
Similary, it is possible to approximate the Jeffrey Information Distance:

$$\begin{aligned}\mathrm{JID}(f,g) &= D_{KL}(f||g) + D_{KL}(g||f) \\ &= \left| \int_0^1 \left[\int_{g(t)}^{f(t)} \frac{g''(g^{-1}(y))}{|g'(g^{-1}(y))|^2} - \frac{f''(f^{-1}(y))}{|f'(f^{-1}(y))|^2} dy \right] dt \right| \\ &= R'|\mathbb{E}[f-g]|, \end{aligned} \tag{3.5}$$

where $|R'| \leq C_g + C_f$.

3.3.1 Connection with Kolmogorov complexity

The main rationale behind the proposed methods is that, since the Kullback-Leibler divergence between two functions f and g is equivalent to the entropy of $g^{-1} \circ f$, and since differential entropy is a measure of complexity, then Kullback-Leibler itself is a measure that accounts for the complexity of the composition $g^{-1} \circ f$.

In the following, a connection with Kolmogorov complexity will be delineated more precisely.

Consider a differentiable, strictly decreasing function $g(t) : [0,1] \to [0,1]$ piece-wise defined such that $g(t) = \alpha_1 t^{\gamma_1} + \beta_1$ if $t < 1/2$ and $g(t) = \alpha_2 t^{\gamma_2} + \beta_2$ otherwise.

For any $\lambda \in [0,1]$, define $g_1^\lambda(t)$ and $g_2^\lambda(t)$ to be, respectively, the part of g before and after λ, properly normalized to be from the unit interval onto itself:

$$\begin{aligned} g_1^\lambda(t) &= \frac{g(t) - g(\lambda)}{g(0) - g(\lambda)} \\ g_1^\lambda(t) &= \frac{g(t) - g(0)}{g(\lambda) - g(0)}. \end{aligned}$$

From Proposition 2.2.4, Kullback-Leibler divergence of g_1^λ from g_2^λ is the negative of the differential entropy of the composition $(g_2^\lambda)^{-1} \circ g_1^\lambda$. If $\lambda = 1/2$, then the aforementioned composition is still in the form $\alpha t^\gamma + \beta$, for some α, β and γ. On the other hand, if $\lambda \neq 1/2$, the composition is itself defined as a piece-wise function, with each piece requiring a different definition. As a result the composition is "more complex" than in the case $\lambda = 1/2$.

The notion of complexity in the previous paragraph can be made rigorous by referring to Kolmogorov complexity.

In particular, we will delineate a connection between $D(g_1^\lambda||g_2^\lambda)$ and the Kolmogorov complexity of a discretization of $(g_2^\lambda)^{-1} \circ g_1^\lambda$.

Let $f : [0,1] \to [0,1]$ be a monotonic function. Given a sufficiently fine quantization Δ composed of N bins, with f^Δ denoting the corresponding quantized version of f and E_f^Δ denoting the Shannon entropy of f^Δ, it holds that

$$E_f \approx E_f^\Delta - \ln(N).$$

Let U be a uniform random variable attaining values in $[0,1]$ and let $\{f_i^\Delta\}_{i=1,\dots,n}$ be a vector of n values sampled from the random variable $f^\Delta(U)$.

Since it is known that f is monotonic it is possible, for any n, to consider the increasing rearrangement $\{(f^\Delta)_i^*\}_{i=1,\dots,n}$ of the sequence $\{f_i^\Delta\}_{i=1,\dots,n}$ and, if n is large enough, $\{(f_i^\Delta)^*\}_{i=1,\dots,n}$ can be considered as approximating a uniform discretization of f^Δ. It is possible to consider the rearrangement $\{(f^\Delta)_i^*\}_{i=1,\dots,n}$ instead of $\{f^\Delta{}_i\}_{i=1,\dots,n}$, if their difference in terms of complexity is correctly accounted for.

Indeed, given a string b of length n, whose minimum program is p, it is possible to define a program q that outputs any given rearrangement b^* of b by detailing which particular permutation corresponds to the rearrangement. But any permutation of n elements can be encoded in a program using an amount of bits proportional to $n\log(n)$ by simply storing it as a vector of numbers. Hence, the difference between p and q, is equal to $n\log(n)$ bits.

As a consequence [98],

$$\begin{aligned} E_f^\Delta &= \lim_{n\to+\infty} \frac{1}{n}\mathbb{E}\left[\mathbb{K}(\{f^\Delta{}_i\}_{i=1,\dots,n})\right] \\ &= \lim_{n\to+\infty} \frac{1}{n}\mathbb{E}\left[\mathbb{K}(\{(f^\Delta)_i^*\}_{i=1,\dots,n}) + n\log(n)\right] \\ &= \lim_{n\to+\infty} \frac{1}{n}\mathbb{E}\left[\mathbb{K}(\{(f^\Delta)_i^*\}_{i=1,\dots,n})\right] + \log(n), \end{aligned}$$

where $\mathbb{K}(b)$ is the Kolmogorov complexity of b.

If both N and n are large enough, then $\{(f^\Delta)_i^*\}_{i=1,\dots,n}$ approximates a discretization of f on an uniform grid and

$$E_f \approx \frac{1}{n}\mathbb{E}\left[\mathbb{K}(\{(f^\Delta)_i^*\}_{i=1,\dots,n})\right] + \ln(n) - \ln(N).$$

Now we take $f = (g_2^\lambda)^{-1} \circ g_1^\lambda$. If $\lambda = 1/2$ the composition $(g_2^\lambda)^{-1} \circ g_1^\lambda$ is a simple power law, and the Kolmogorov complexity of any of its uniform discretizations is small, as it can be produced as the output of a program that depends on 4 parameters only: α, β, γ and the number of elements in the uniform discretization of $(g_2^\lambda)^{-1} \circ g_1^\lambda$. On the contrary, if $\lambda \neq 1/2$, such a short description is not possible, as it is necessary to store in the program the parameters describing both g_1^λ and g_2^λ, as well as the breaking point of the piecewise definition.

3.3.2 Algorithms for coefficients selection

Both KLID and JID are employed to automatically find the optimal set of reconstruction coefficients, in the sense of information contribution. As in the case of ENID, the problem is seen as a source separation problem, where our aim is to find the point that separates the "most representative" coefficients from the "less representative" coefficients. We remark that the process is completely automatic and no parameter setting is required.

As for ENID, in order to find the separation point, the coefficients curve $a(t)$ is partitioned in two different sets, one comprising the coefficients before a certain point λ and the other comprising the coefficients after said point. If both coefficient sets are emitted from a single source, then their distance in terms of complexity will be minimal. On the other hand, if one of the coefficient sets contains elements from more than one source, its distance from the other set is higher, in terms of complexity. Hence, the separation point is the λ that minimizes the chosen information measure.

Since KLID and JID compare functions defined from the unit interval onto itself, the two curves must be normalized to functions from the unit interval to the unit interval. Assuming $a(t) : [0,1] \to \mathbb{R}^+$, is the strictly decreasing coefficients curve, the two normalized curves $a_1^\lambda(t)$ and $a_2^\lambda(t)$ are defined as

$$a_1^\lambda(t) = \frac{a(\lambda t) - a(\lambda)}{a(0) - a(\lambda)} \quad t \in [0, \lambda]$$

and

$$a_2^\lambda(t) = \frac{a(\lambda + (1-\lambda)t) - a(1)}{a(\lambda) - a(1)} \quad t \in [\lambda, 1].$$

Then, for any λ, the chosen information distance (KLID or JID) is computed using the appropriate approximation. The optimal point is detected as the λ realizing the minimum between all the values of $\mathrm{KLID}(a_1^\lambda, a_2^\lambda)$ (or $\mathrm{JID}(a_1^\lambda, a_2^\lambda)$).

The algorithms are structured as follows:

Algorithm 2 KLID selection method

Input: Sequence of discretized values of a function u: $\{u_i\}_{i=1,\dots,M}$
Output: Approximation of u

1. Compute the discrete wavelet transform of $\{u_i\}_{i=1,\dots,M}$ and extract the detail coefficients $\{a_j\}_{j\in 1,\dots N}$
2. Sort the absolute values of the detail coefficients in decreasing order
3. For all $j = 1, \dots N$
   (a) Define $\lambda_j = j/N$
   (b) For $1 \le i < j$, $a_1^i = (a_i - a_j)/(a_1 - a_j)$
   (c) For $j \le i \le N$, $a_2^i = (a_i - a_N)/(a_j - a_N)$
   (d) Define $a_1 = \{a_1^i\}_{i=1,\dots,N}$
   (e) Define $a_2 = \{a_2^i\}_{i=1,\dots,N}$
   (f) Compute the approximation of $E_1 = E_{a_1}$ using eq. (2.7)
   (g) Compute the approximation of $E_2 = E_{a_2}$ using eq. (2.7)
   (h) Compute $K(j) = \mathrm{KLID}(a_1, a_2)$, using the approximation $\mathrm{KLID}(a_1, a_2) \approx |E_1 - E_2|$, according to eq. (3.4)
4. Find the index $\tilde{N}$ that realizes the minimum of the vector K. Preserve the first $\tilde{N}$ rearranged coefficients and set to zero the remaining ones
5. Invert the discrete wavelet transform

Algorithm 3 JID selection method

Input: Sequence of discretized values of a function u: $\{u_i\}_{i=1,\dots,M}$
Output: Approximation of u

1. Compute the discrete wavelet transform of $\{u_i\}_{i=1,\dots,M}$ and extract the detail coefficients $\{a_j\}_{j\in 1,\dots N}$
2. Sort the absolute values of the detail coefficients in decreasing order
3. For all $j = 1, \dots N$
   (a) Define $\lambda_j = j/N$
   (b) For $1 \leq i < j$, $a_1^i = (a_i - a_j)/(a_1 - a_j)$
   (c) For $j \leq i \leq N$, $a_2^i = (a_i - a_N)/(a_j - a_N)$
   (d) Define $a_1 = \{a_1^i\}_{i=1,\dots,N}$
   (e) Define $a_2 = \{a_2^i\}_{i=1,\dots,N}$
   (f) Compute $J(j) = \mathrm{JID}(a_1, a_2)$, using the approximation $\mathrm{JID}(a_1, a_2) \approx |\mathbb{E}[a_1 - a_2]|$, according to eq. (3.5)
4. Find the index $\tilde{N}$ that realizes the minimum of the vector J. Preserve the first $\tilde{N}$ rearranged coefficients and set to zero the remaining ones
5. Invert the discrete wavelet transform

Remark 3.3.2. *The fact that JID can be approximated as the* absolute value of the mean*, rather than the* mean of the absolute value *greatly simplifies the computation of Algorithm 3. Since a_1 and a_2 have different lengths, to compute the mean of the absolute value of their difference some form of interpolation must be used. On the other hand, the mean of the difference can be computed as the difference of the individual means.*

Questions might arise whether the proposed approximations are accurate enough to preserve the source-discrimination capacity of KLID and JID.

Notwithstanding that the proposed approximations are proportional to the actual information measures if the constants C_g and C_f are finite, the following example shows with a simple, explicit argument, that this discrimination capacity between sources is maintained if the proposed approximation of Kullback-Leibler divergence is used in the estimation of JID and KLID. Let us consider the simple case in which the curve of coefficients $a(t)$ is composed by two linear segments: given $\tilde{t}, \tilde{a} \in [0, 1]$, then

$$a(t) = \begin{cases} 1 - (1 - \tilde{a})\dfrac{t}{\tilde{t}} & t < \tilde{t} \\ -\dfrac{\tilde{a}}{1 - \tilde{t}}(t - \tilde{t}) + \tilde{a} & t \geq \tilde{t} \end{cases}.$$

If $\lambda = \tilde{t}$, then both $a_1^\lambda(t) \equiv t$ and $a_2^\lambda(t) \equiv t$ and, as a consequence, $E_{a_1^\lambda} = E_{a_2^\lambda} = 0$ and $\mathbb{E}[a_1^\lambda - a_2^\lambda] = 1/2 - 1/2 = 0$.

On the other hand, suppose $\lambda > \tilde{t}$. Then, $a_2^\lambda(t) \equiv t$ and $E_{a_2^\lambda} = 0$ while $a_2^\lambda(t) \neq t$, $E_{a_1^\lambda} < 0$ and $|E_{a_1^\lambda} - E_{a_2^\lambda}| > 0$. In addition, $\mathbb{E}[a_2^\lambda] = 1/2$ but, since $a_1^\lambda(t) < t$, then $\mathbb{E}[a_1^\lambda] < 1/2$ and $|\mathbb{E}[a_1^\lambda - a_2^\lambda]| > 0$. As a consequence the only minimum for both KLID and JID is located at $\lambda = \tilde{t}$.

3.3.3 Experimental results

Performances of both KLID and JID have been studied on the same test functions that have been used for ENID and which are shown in fig. 3.3: the three functions **Bumps**, **Heavy sine** and **Blocks**, provided by the Matlab functions **wnoise**, and the piece-wise polynomial function **Poly**, defined in eq. 3.3.

The test functions have been discretized using 512 points and then decomposed using dyadic wavelet transform (DWT); 4 levels of decomposition have been considered using Daubechies' wavelet of order 2 (db2), 4 (db4) and 12 (db12) and symlets wavelets of order 4 (sym4) and 12 (sym12). Only the detail coefficients have been considered for selection, while the approximation coefficients have been retained. The detail coefficients have been selected by means of KLID selection method (Algorithm 2), and by means of JID selection method (Algorithm 3).

Preliminary numerical experiments showed that the presence of plateaus in the coefficients curves worsened the effectiveness of KLID and JID selection methods. Usually plateaus are not present but we noticed that, if the rearranged wavelet coefficients decay too fast (such as for Blocks or Poly), the tail of the coefficients curve assumes small values that range from 10^{-12} to 10^{-16}, and they tend to arrange themselves in plateaus. In KLID approximation this phenomenon results in zeros inside the logarithm in the computation of the entropies, leading to infinities; on the contrary, in JID it has been observed that the presence of these plateaus tends to generate an oscillating "sawtooth - like" behaviour which leads to undesired minima.

A simple heuristic has been implemented to avoid these irregularities, consisting in deleting duplicate values from the computations: in this way the plateaus, and the consequent infinities and "sawtooth-like" behaviours are eliminated. This deletion procedure does not affect noticeably the behaviour of the approximation of KILD and JID when computed far from plateaus.

Reconstruction errors and compression ratios are reported in table 3.7. The reconstruction error has been evaluated using mean square error (MSE) and normalized compression distance (NCD). The latter has been computed as detailed in Subsection 3.1.4. Both KLID and JID perform well if the test function is sufficiently regular (Bumps and Heavy Sine), while they give less stable results if the test function is less regular (Blocks and Poly). In particular, in the cases of Bumps and Heavy Sine the numbers of coefficients selected by KLID and JID are similar, while in the cases of Blocks and Poly the number of coefficients selected by KLID is usually much smaller than that selected by JID.

In figs. 3.11 - 3.14 the plots of KLID, JID, MSE error and NCD error are shown for each test functions and for each selected wavelets basis.

Fig. 3.11 refers to the Bumps. It can be observed that, for each wavelet, the point selected by JID is usually in correspondence to the first change of trend in

the NCD error. On the other hand, the point selected by KLID is closer to the second change of trend, where NCD error tends to get constant. This behaviour is especially evident in figs. 3.11 (d) and (e), where the trend of KLID matches closely NCD error.

Similar results can be observed in fig. 3.12 for the function Blocks. In this case the behaviours of KLID and JID are inverted, with respect to the function Bumps: the number of coefficients selected by KLID is very low, whereas, in figs. 3.12 (b), (c) and (d), in correspondence to the number of coefficients selected by JID, NCD error suddenly drops to a constant value. As it can observed in figs. 3.12 (a) and (d), JID has multiple local minima and the selected one does not correspond to the NCD error. Nevertheless, in both figs. 3.12 (a) and (d), it can be observed that the minima are close to the knee of MSE curve.

The results concerning the function Heavy Sine vary significantly depending on the chosen wavelet basis. In the cases of db2 and sym4 wavelet bases, depicted in figs. 3.13 (a) and (c), the minima of both KLID and JID correspond to slight changes of trend in NCD error, but they are far from the major change of trend, as it is realized around 250 selected coefficients. Similar considerations can be made if db4 is used: as shown in fig. 3.13 (b), the minima of KLID and JID are far from the knee of NCD error – which takes place toward the end of the curve – but they are still relatively closer to the knee of MSE curve. The results get significantly better if wavelet db12 or sym12 are considered (figs. 3.13 (d) and (e)): in these cases the number of coefficients selected by JID correspond almost exactly to the point where the plateau in NCD error begins. KLID is less precise and, even if its minimum is close to the beginning of the plateau as well when db12 is used (fig. 3.13 (d)), it is located farther away when sym12 is employed (fig. 3.13 (e)) .

Function Poly revealed to be a challenge for the proposed method. As it can be observed from fig. 3.14 (a) - (e), KLID always selects a very small number of coefficients, and no correspondence to NCD behaviour can be found. On the other hand JID, even if severely oscillating, presents several local minima that are very close to the knee of NCD error, and the selected one is always in the proximity of the beginning of the plateau in NCD error.

Opposite to ENID, both KLID and JID tend to select less coefficients: as a consequence mean square error (MSE) is higher, but compression ratios are lower. Indeed, concerning compression rates, the only case in which ENID performs better than KLID and JID is the function Blocks: for the other functions, KLID and JID automatic selection methods individuate a set of coefficients that achieves better compression ratios than ENID.

In addition it has been observed that KLID and JID minima are in correspondence to changes of NCD error profile, providing empirical evidence of their relation with NID and, ultimately, with Kolmogorov complexity.

Wavelet	KLID				JID			
	MSE	NCD	$\widetilde{N}$	ratio	MSE	NCD	$\widetilde{N}$	ratio
Bumps								
db2	$1.98153297 \times 10^{-8}$	0.59	293	0.56	$1.9426688359 \times 10^{-6}$	0.75	221	0.42
db4	1.422005×10^{-10}	0.56	314	0.58	$9.78183682 \times 10^{-8}$	0.77	238	0.44
sym4	2.633171×10^{-10}	0.55	305	0.56	$6.521811699 \times 10^{-7}$	0.85	212	0.41
db12	10^{-15}	0.15	448	0.74	8.6521×10^{-12}	0.50	351	0.58
sym12	7×10^{-15}	0.17	427	0.70	$1.3629480336 \times 10^{-6}$	0.90	228	0.37
Blocks								
db2	$2.32438746011071 \times 10^{-2}$	1.05	41	0.08	$8.4487539059420 \times 10^{-3}$	1.05	56	0.11
db4	$2.02094379678080 \times 10^{-2}$	1.07	45	0.08	$5.797934558 \times 10^{-7}$	1.02	160	0.29
sym4	$1.95333658944421 \times 10^{-2}$	1.08	45	0.08	2.2185246×10^{-9}	1	182	0.34
db12	7×10^{-15}	1.08	63	0.1	0	0.39	348	0.56
sym12	$1.85522869679502 \times 10^{-2}$	1.06	65	0.11	$4.697544142048 \times 10^{-4}$	1.06	140	0.23
Heavy Sine								
db2	$1.69415690016 \times 10^{-5}$	0.95	80	0.15	$1.9866792048 \times 10^{-6}$	0.89	124	0.23
db4	1.195265×10^{-10}	0.95	160	0.29	8.036645×10^{-10}	0.96	134	0.25
sym4	9.68818×10^{-11}	0.57	162	0.3	9.056830×10^{-10}	0.64	131	0.24
db12	0	0.12	300	0.50	2.49×10^{-14}	0.21	273	0.45
sym12	0	0.12	469	0.78	0	0.12	302	0.5
Poly								
db2	$4.21907475861 \times 10^{-5}$	0.94	34	0.06	1.75587×10^{-11}	0.18	282	0.54
db4	$3.10741245453 \times 10^{-5}$	0.97	41	0.8	1.4918522×10^{-9}	0.75	116	0.21
sym4	$6.03920969335 \times 10^{-5}$	0.96	38	0.07	1.8075×10^{-12}	0.37	152	0.28
db12	$3.65137763392\ 10^{-5}$	0.99	73	0.12	0	0.37	316	0.52
sym12	$6.29647532687 \times 10^{-5}$	1	63	0.1	3.96955×10^{-11}	0.56	220	0.36

Table 3.7: Coefficients selection using KLID and JID: reconstruction errors and compression rates.

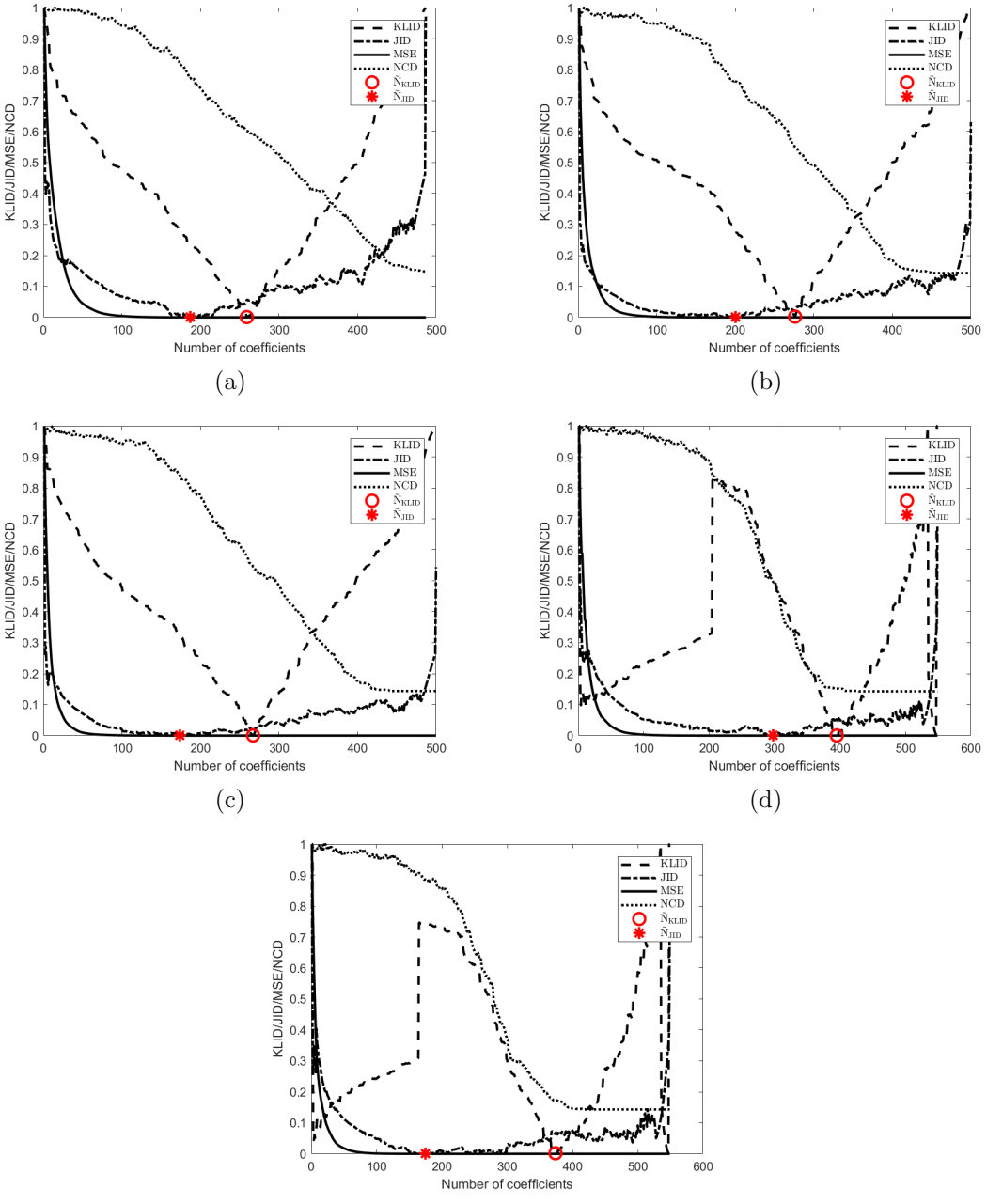


Figure 3.11: Function Bumps decomposed using DWT. KLID, JID, MSE and NCD for wavelet basis db2 (a), db4 (b), sym4 (c), db12 (d), sym12 (e). The minimum of KLID is marked by a circle and the minimum of JID is marked by a diamond. The graphs have been normalized for illustration purposes.

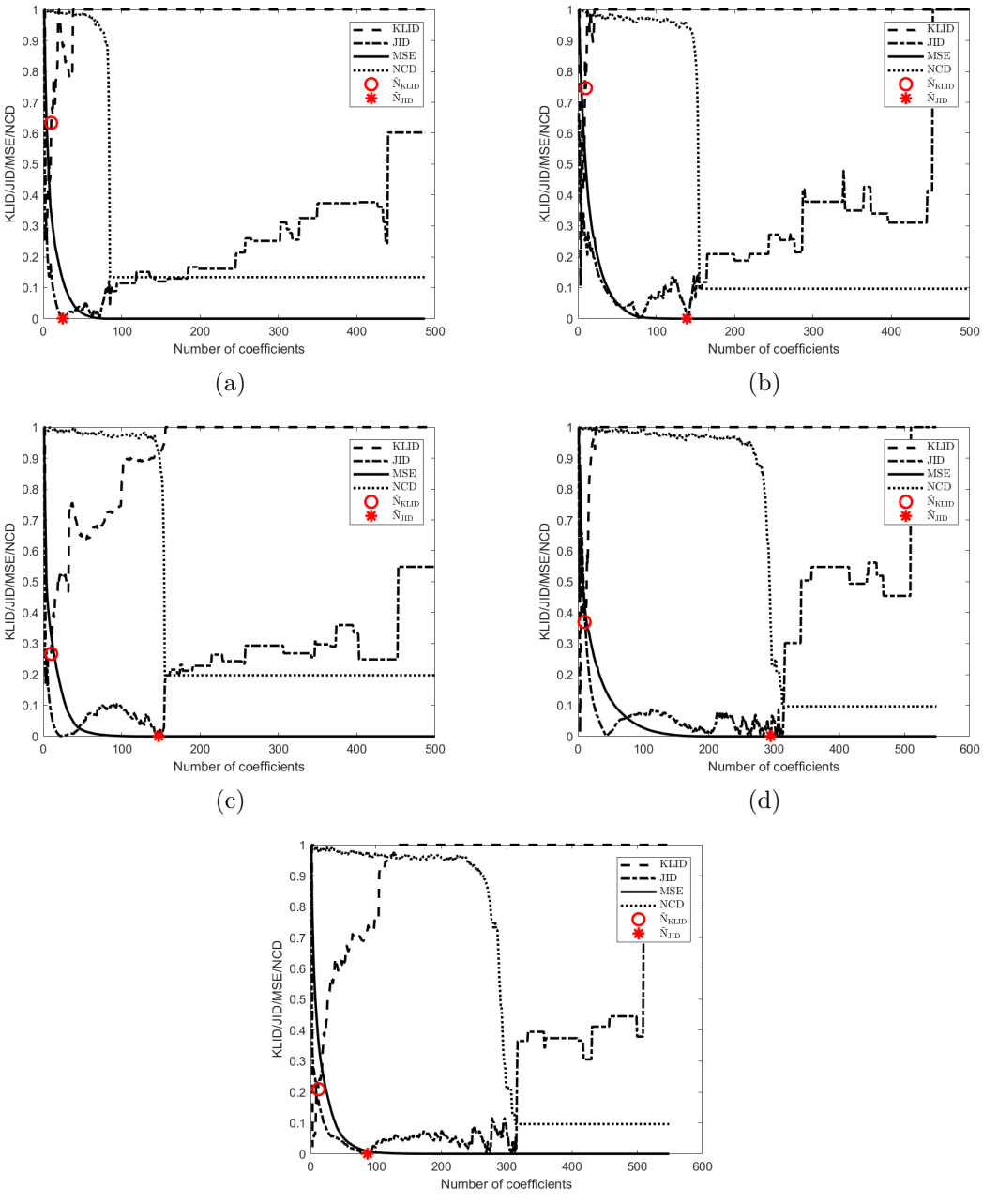
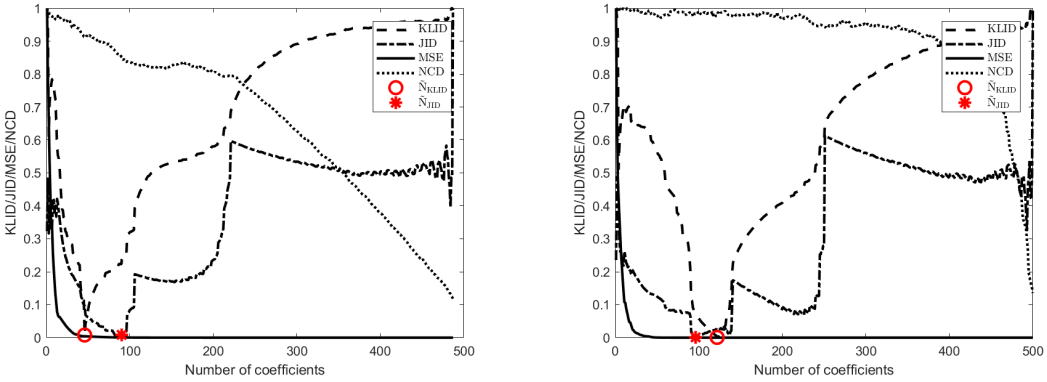
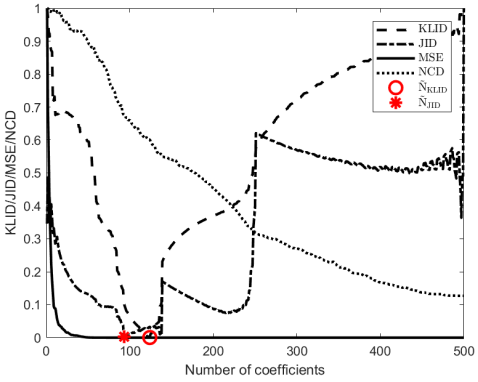


Figure 3.12: Function Blocks decomposed using DWT. KLID, JID, MSE and NCD for wavelet basis db2 (a), db4 (b), sym4 (c), db12 (d), sym12 (e). The minimum of KLID is marked by a circle and the minimum of JID is marked by a diamond. The graphs have been normalized for illustration purposes.

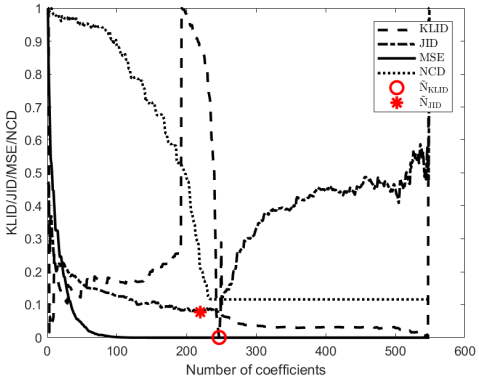


(a)

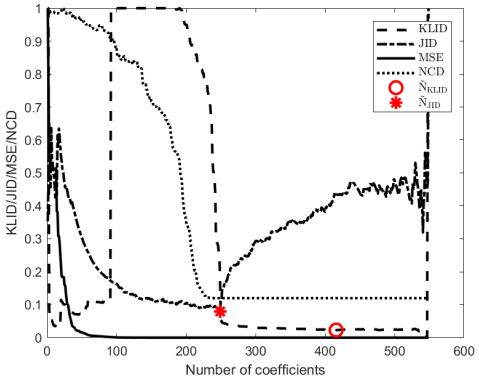
(b)



(c)



(d)



(e)

Figure 3.13: Function Heavy Sine decomposed using DWT. KLID, JID, MSE and NCD for wavelet basis db2 (a), db4 (b), sym4 (c), db12 (d), sym12 (e). The minimum of KLID is marked by a circle and the minimum of JID is marked by a diamond. The graphs have been normalized for illustration purposes.

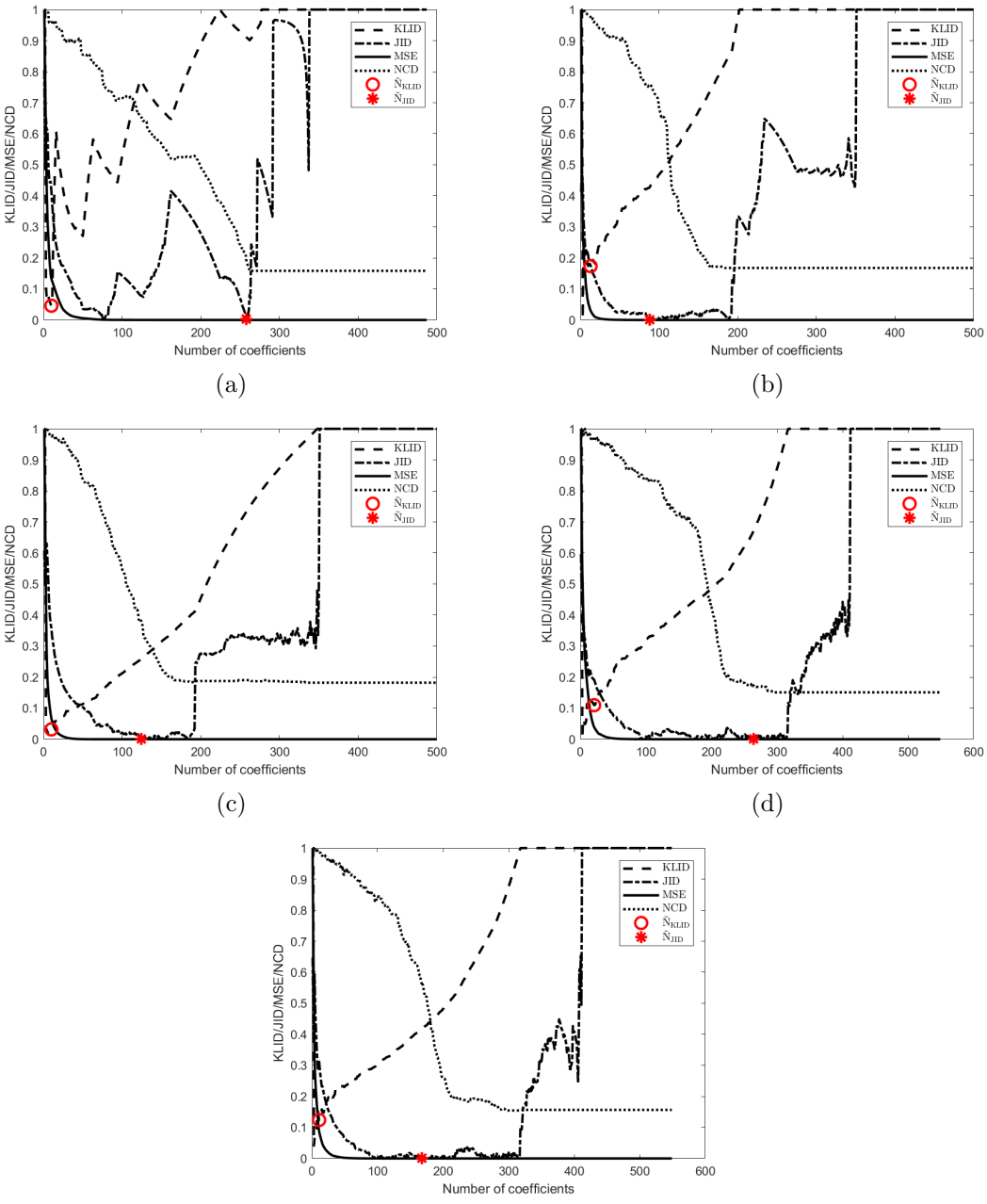


Figure 3.14: Function Poly decomposed using DWT. KLID, JID, MSE and NCD, in wavelet basis db2 (a), db4 (b), sym4 (c), db12 (d), sym12 (e). The minimum of KLID is marked by a circle and the minimum of JID is marked by a diamond. The graphs have been normalized for illustration purposes.

3.3.4 Noisy case

Whereas the approximation provided by ENID coefficients selection has been observed to be so accurate that even additive noise is correctly reconstructed, the approximations provided by JID and KLID coefficients selection tend to adapt and to select fewer coefficients if additive noise is added. Indeed, numerical experiments show that the number of selected coefficients is close to the optimum and is comparable with state of the art methods.

As in the noiseless case, the functions have been discretized using 512 points but, for these numerical tests, they have been normalized to have maximum absolute value equal to 2, for the sake of fair comparisons. Subsequently, gaussian noise with variances ranging from 0.1 to 1 has been added. Since gaussian noise is unbounded, normalization needs to occur before noise is added, otherwise spikes in the noise realization might results in an excessive shrink of all the underlying function. The considered wavelet are the same as in the noiseless case.

To further investigate the possible use of the proposed method for denoising purposes, stationary wavelet transform (SWT) has been considered for decomposition as well. The shift-invariance property of SWT has proved to be very effective in denoising, reducing Gibbs oscillations [83].

The proposed methods have been compared with other state-of-the art thresholding techniques: Donoho universal threshold, both using the real noise variance and estimating it from the data; SURE thresholding and Minimax thresholding. They are implemented in the matlab function **wden**. Furthermore, for each experiment, the optimal threshold has been considered for comparison. The optimum threshold has been selected by picking the one giving the minimum MSE among all possible thresholds. For each case, this optimum threshold realizes the best possible result using hard thresholding.

Numerical results are presented in tables 3.8-3.15 and depicted in figures 3.15 - 3.22. In each of those tables, for space purposes, Donoho threshold with real noise variance has been denoted by **D (R)**, while Donoho threshold with estimated noise variance has been denoted by **D (A)**. The optimal threshold error has been denoted by **Opt.**.

Figures 3.15 and 3.16 refer to, respectively, the function Blocks decomposed using DWT and SWT. In both cases the MSE of the reconstructions selected by JID and KLID are generally better than those provided by SURE and Minimax thresholding method. In particular, if SWT is employed, this is always true.

As it can be observed in fig. 3.17, in the case of Bumps, if DWT decomposition is considered, the proposed methods always performs better than Minimax. On the other hand, the comparison with SURE depends on the wavelet, with the proposed methods outperforming SURE if wavelets db2 or sym4 are considered. On the other hand, if SWT decomposition is considered, the proposed methods always perform better than SURE and Minimax, as shown in fig. 3.18. The only exception is if excessive noise is present and the wavelet has a high number of vanishing moments (figs. 3.18 (d) and (e), $\sigma = 1$).

With regard to Heavy Sine decomposed using DWT, whose numerical results are shown in fig. 3.19, KLID and JID performs better than both SURE and Minimax if the noise is low ($\sigma = 0.1$) or if db2 is selected. On the other hand, if SWT is employed, the proposed methods always performs better than SURE and Minimax and are close to Donoho and to the optimum, as shown in fig.

3.20.

In the case of the function Poly, whose results are shown in figures 3.21 and 3.22, the comparison is slightly less favorable: using DWT decomposition KLID and JID perform better than Minimax in every case, but they are not better than SURE; using SWT decomposition the proposed methods perform always better than Minimax and compare favourably to SURE in the case of db2 while achieving similar results in the other cases.

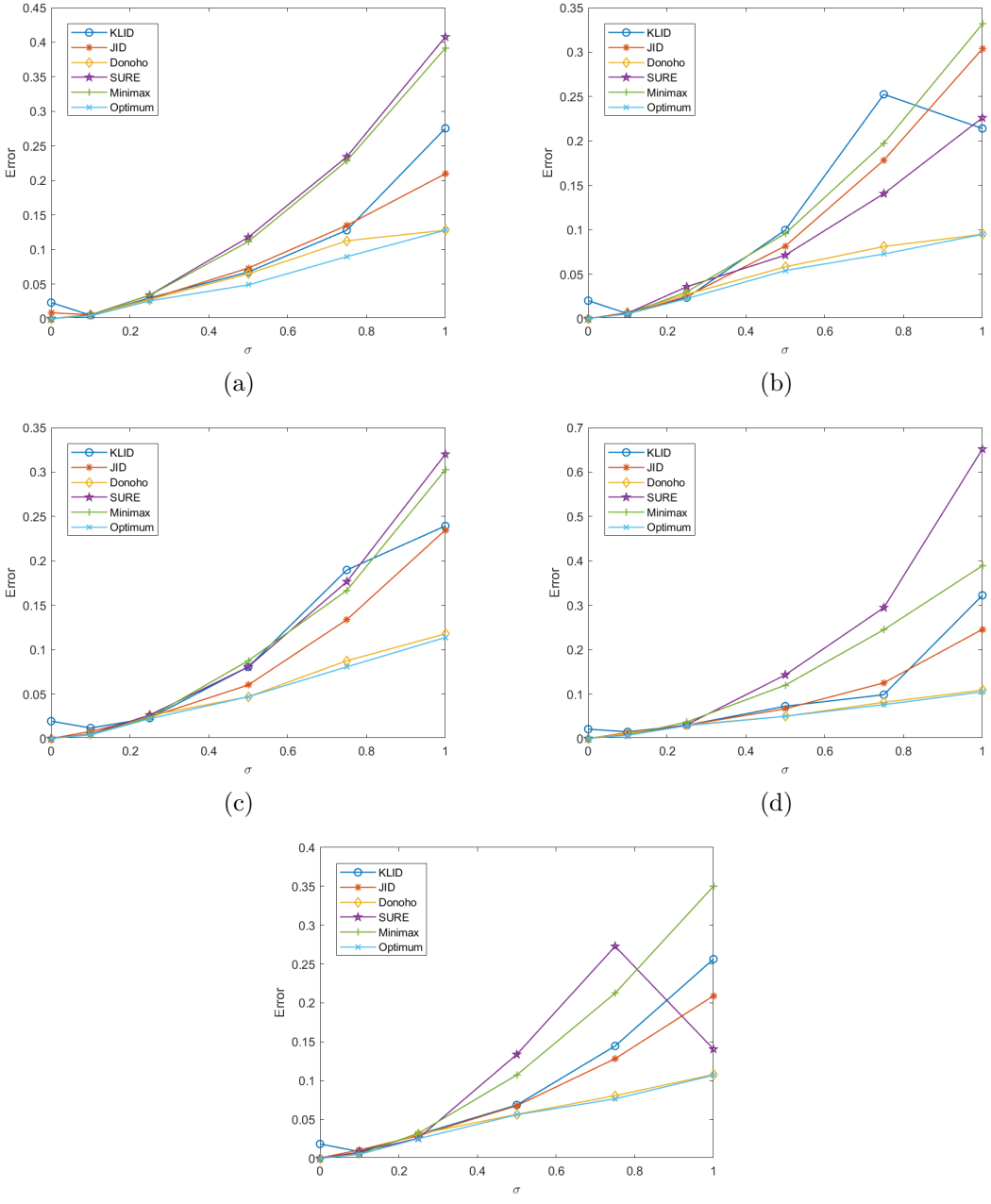


Figure 3.15: Function Blocks decomposed using DWT, noisy case. Error plots for methods KLID, JID, Donoho, SURE, Minimax for wavelet basis db2 (a), db4 (b), sym4 (c), db12 (d), sym12 (e), along with the optimal error. The horizontal axis is the variance σ of the noise, the vertical axis is the magnitude of the errors.

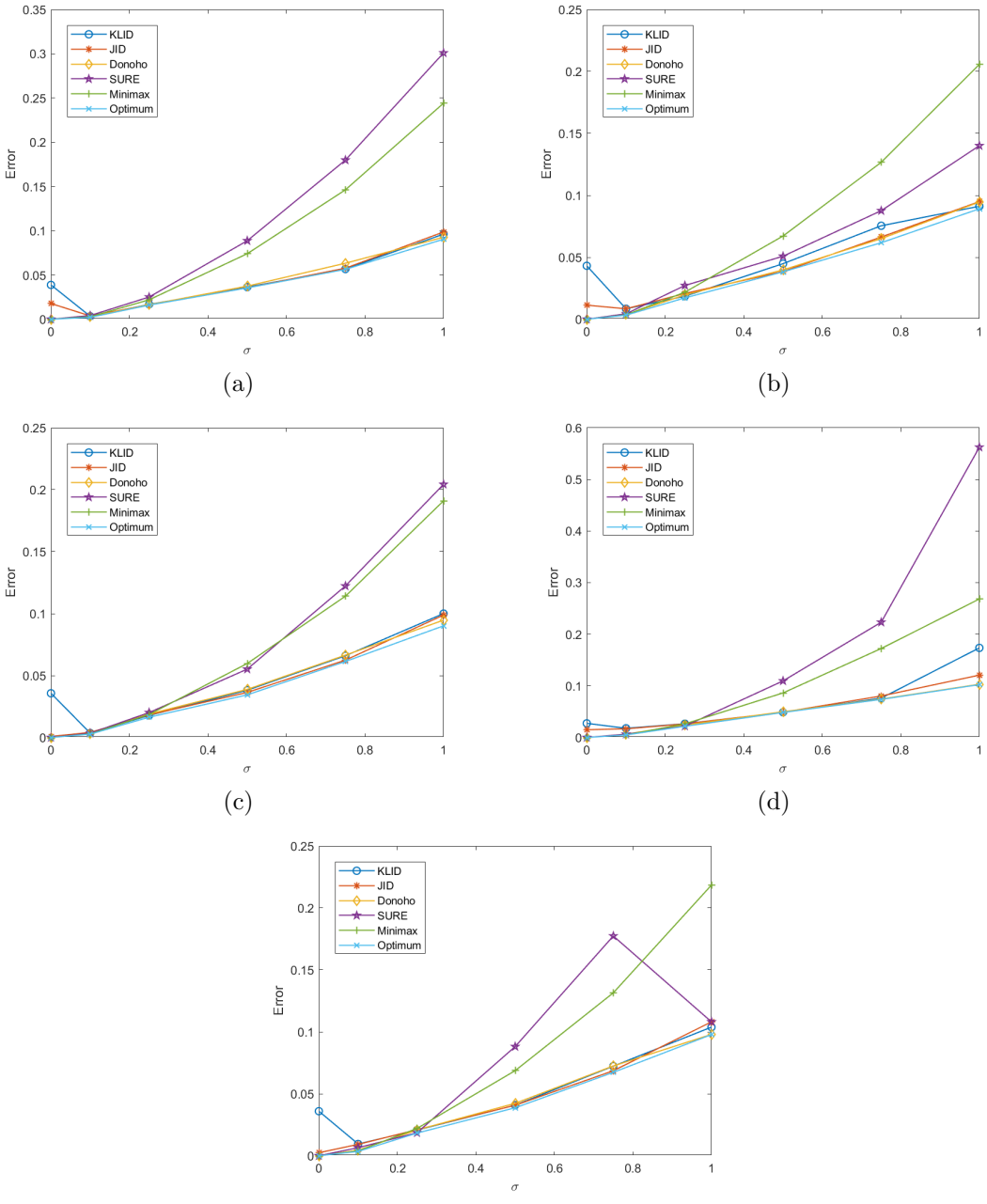


Figure 3.16: Function Blocks decomposed using SWT, noisy case. Error plots for methods KLID, JID, Donoho, SURE, Minimax for wavelet basis db2 (a), db4 (b), sym4 (c), db12 (d), sym12 (e), along with the optimal error. The horizontal axis is the variance σ of the noise, the vertical axis is the magnitude of the errors.

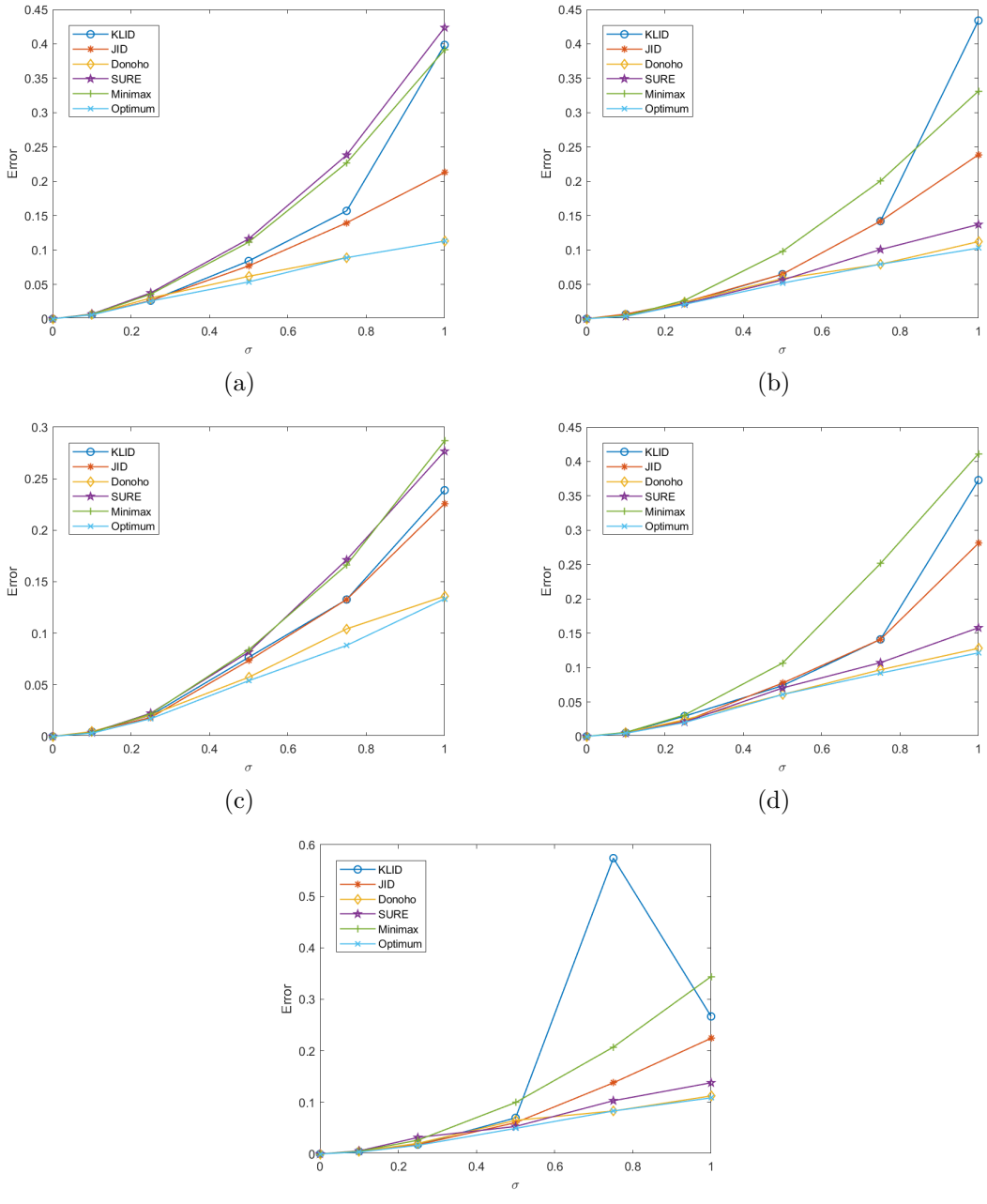


Figure 3.17: Function Bumps decomposed using DWT, noisy case. Error plots for methods KLID, JID, Donoho, SURE, Minimax for wavelet basis db2 (a), db4 (b), sym4 (c), db12 (d), sym12 (e), along with the optimal error. The horizontal axis is the variance σ of the noise, the vertical axis is the magnitude of the errors.

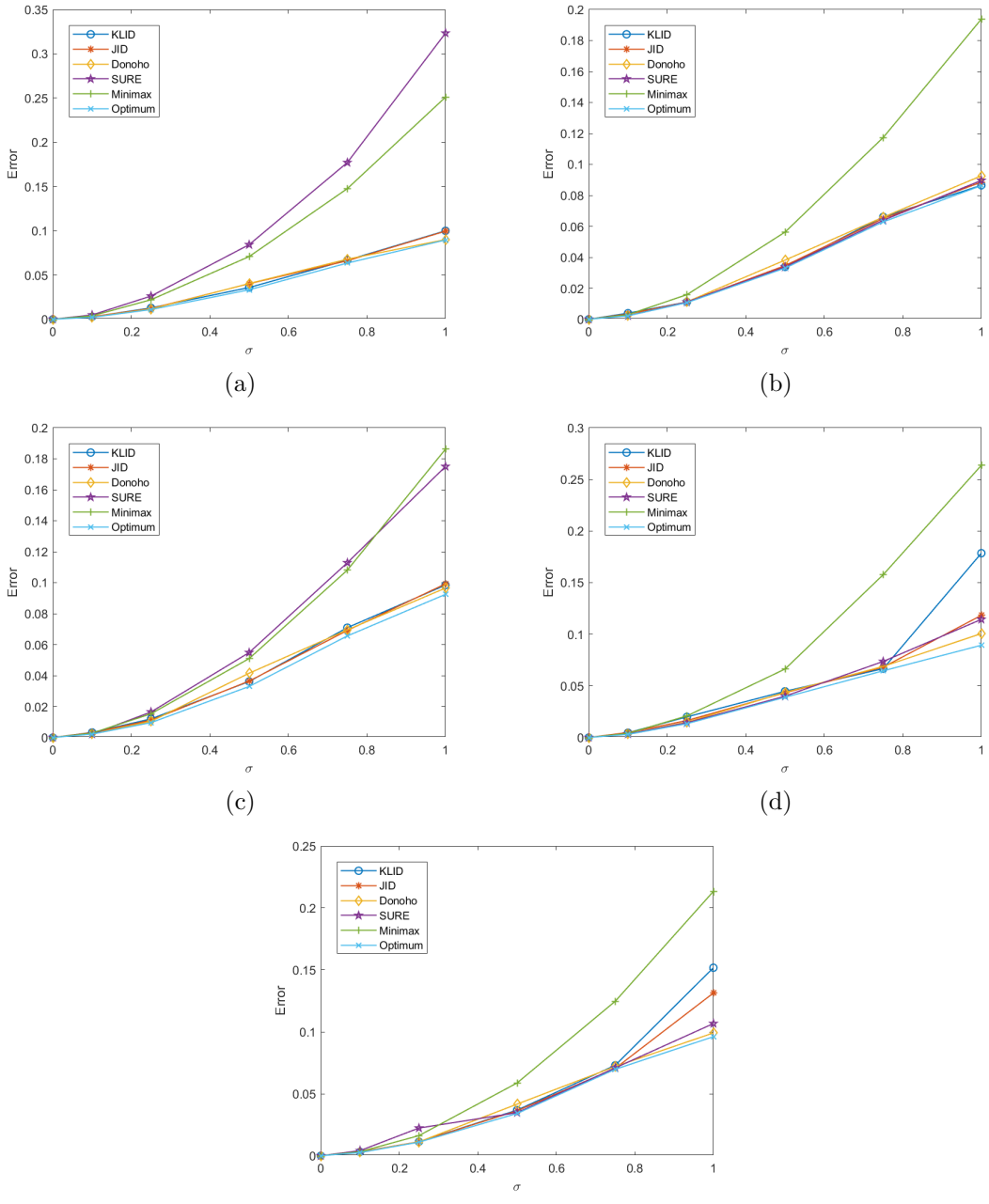


Figure 3.18: Function Bumps decomposed using SWT, noisy case. Error plots for methods KLID, JID, Donoho, SURE, Minimax for wavelet basis db2 (a), db4 (b), sym4 (c), db12 (d), sym12 (e), along with the optimal error. The horizontal axis is the variance σ of the noise, the vertical axis is the magnitude of the errors.

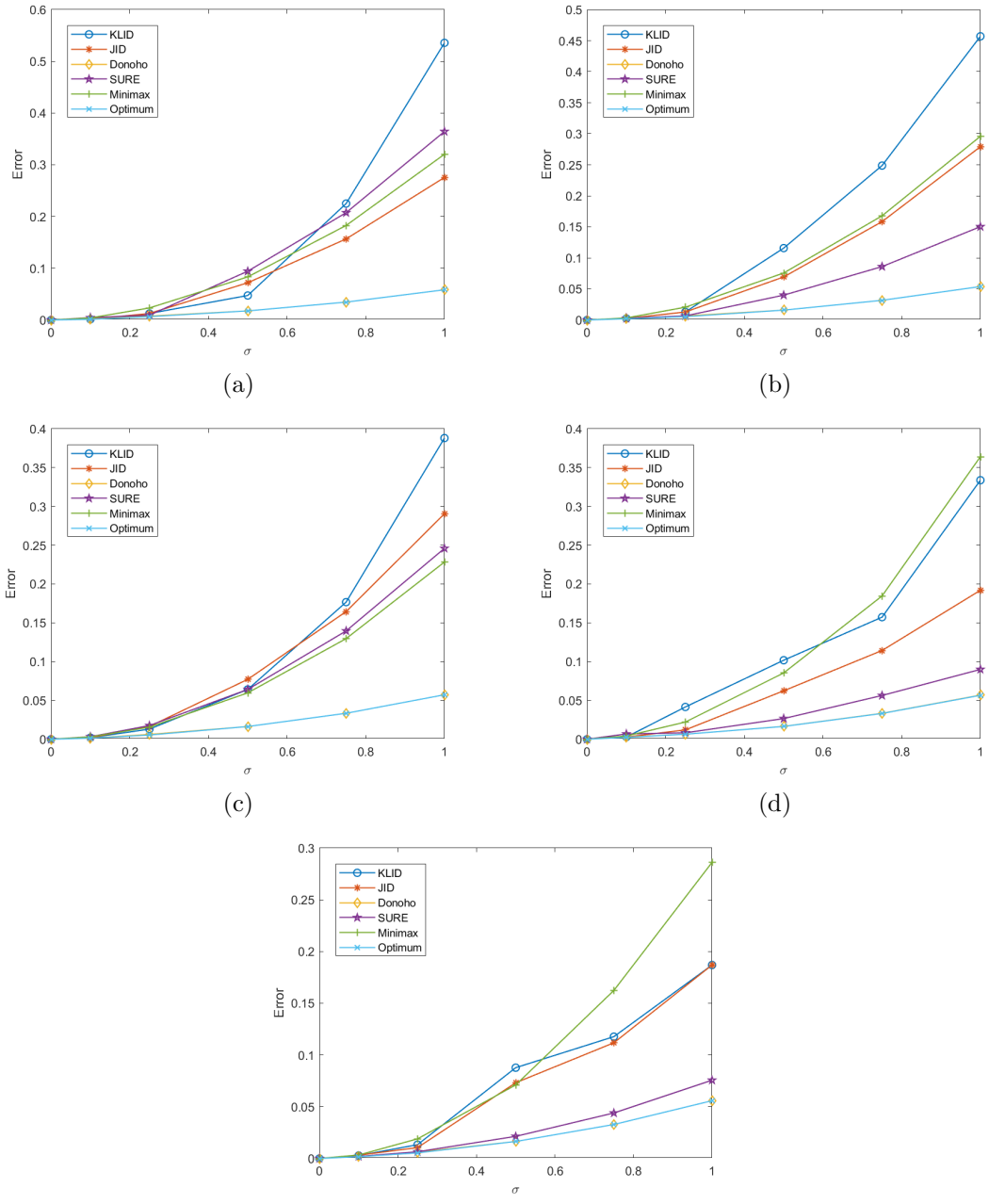


Figure 3.19: Function Heavy Sine decomposed using DWT, noisy case. Error plots for methods KLID, JID, Donoho, SURE, Minimax for wavelet basis db2 (a), db4 (b), sym4 (c), db12 (d), sym12 (e), along with the optimal error. The horizontal axis is the variance σ of the noise, the vertical axis is the magnitude of the errors.

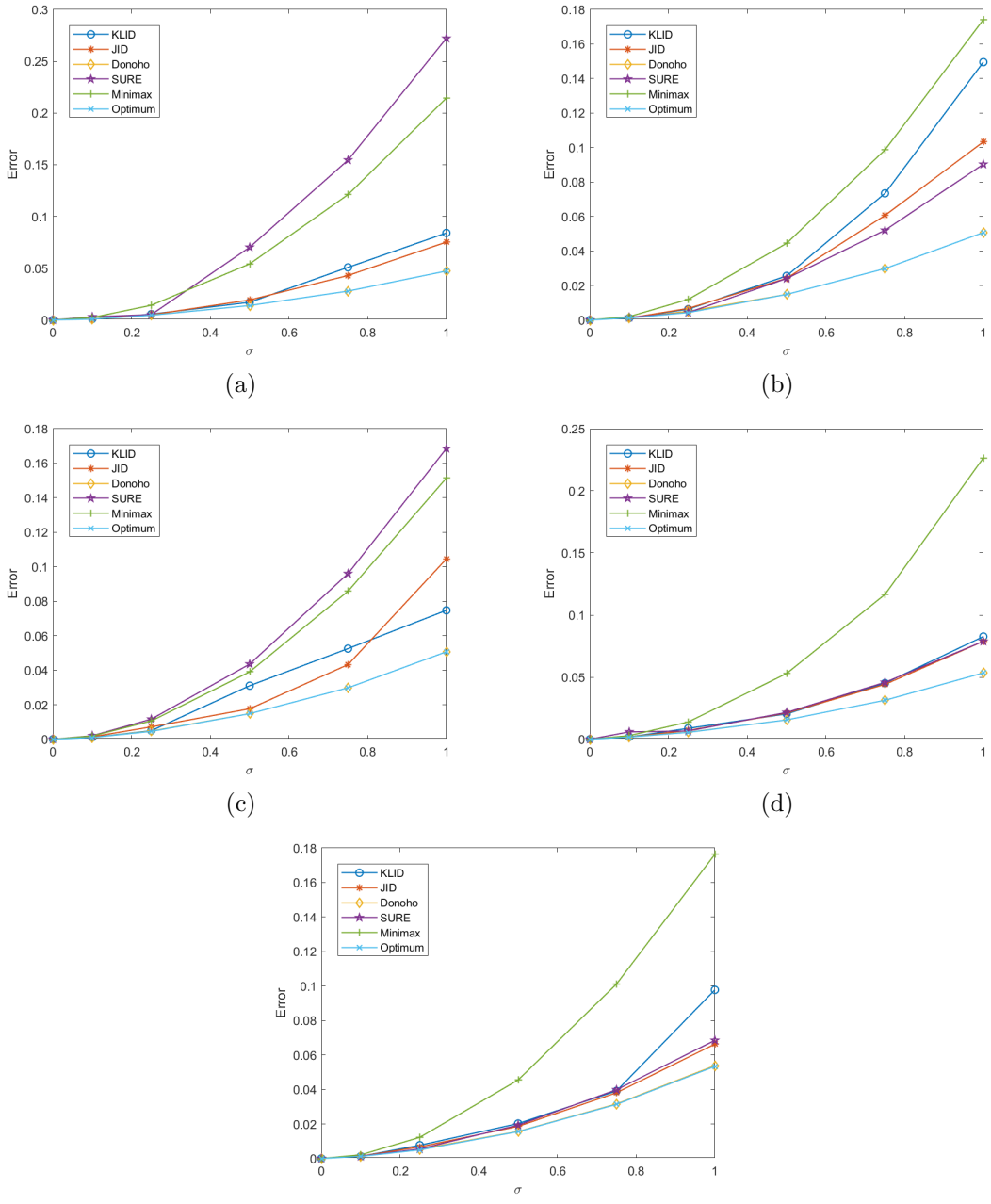


Figure 3.20: Function Heavy Sine decomposed using SWT, noisy case. Error plots for methods KLID, JID, Donoho, SURE, Minimax for wavelet basis db2 (a), db4 (b), sym4 (c), db12 (d), sym12 (e), along with the optimal error. The horizontal axis is the variance σ of the noise, the vertical axis is the magnitude of the errors.

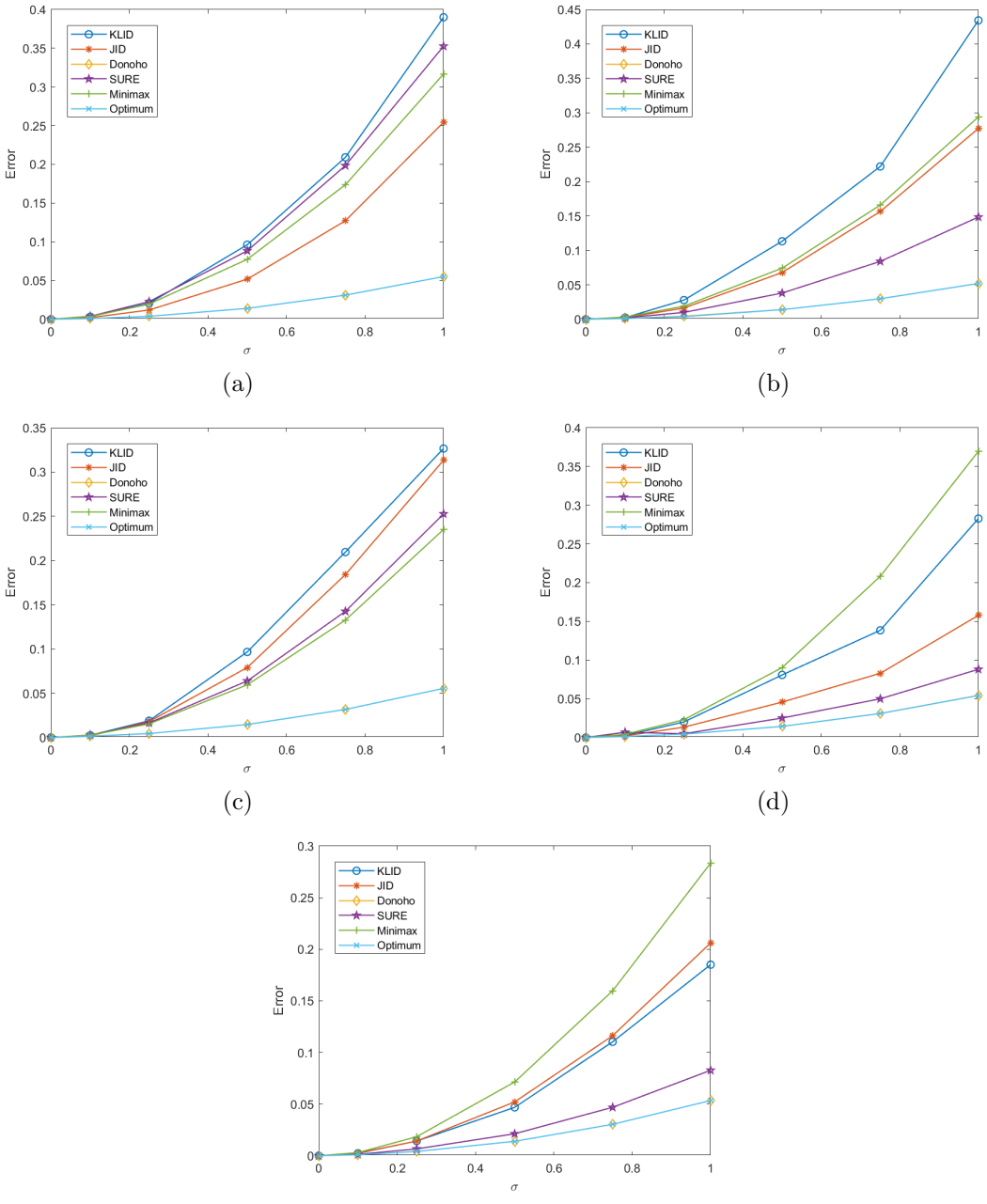


Figure 3.21: Function Poly decomposed using DWT, noisy case. Error plots for methods KLID, JID, Donoho, SURE, Minimax for wavelet basis db2 (a), db4 (b), sym4 (c), db12 (d), sym12 (e), along with the optimal error. The horizontal axis is the variance σ of the noise, the vertical axis is the magnitude of the errors.

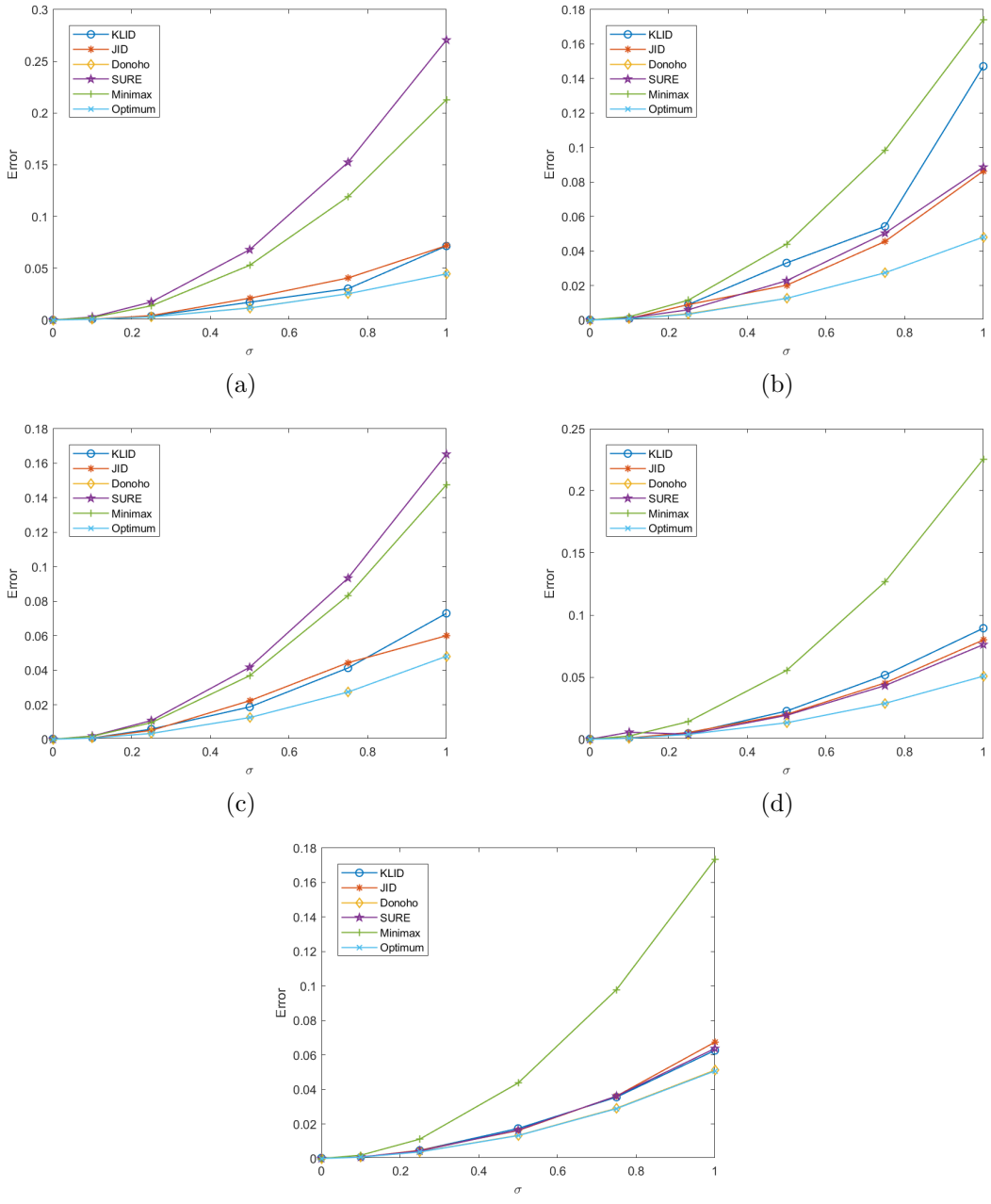


Figure 3.22: Function Poly decomposed using SWT, noisy case. Error plots for methods KLID, JID, Donoho, SURE, Minimax for wavelet basis db2 (a), db4 (b), sym4 (c), db12 (d), sym12 (e), along with the optimal error. The horizontal axis is the variance σ of the noise, the vertical axis is the magnitude of the errors.

To better analyse the performances of KLID and JID in the noisy case,

Bumps and Blocks have been selected and MSE plots with respect to number of coefficients for different noise strengths are shown in fig. 3.23 for DWT decomposition using wavelet db2, with the position of the different methods marked on it. It is worth noting that, in the noiseless case (fig. 3.23 (a)), Donoho threshold with oracle noise strength selects all the coefficients. On the other hand, if noise the strength σ is automatically estimated (using MAD, median absolute deviation [127]), the selected number of coefficients is similar to the proposed methods'. When noise is moderate (figs. 3.23 (b), (c) and (d)), MSE presents a minimum, representing the optimum reconstruction error, whose location is clearly separated from 0. On the other hand, if noise is too strong (figs. 3.23 (e) and (f)), the minimum of MSE is located in 0, meaning that no details is worth preserving. In both cases the proposed methods select a number of coefficients that are very close to the optimum. If the noise is moderate (figs. 3.23 (b), (c) and (d)) the number of coefficients selected are closer to the optimum than when noise is stronger (figs. 3.23 (e) and (f)). The proposed methods always perform better than SURE or Minimax.

In fig. 3.25 the behaviour of KLID and JID is shown, along with MSE; the minima of the proposed methods are marked. It is noteworthy that, apart for some numerical instabilities in the very first and very last coefficients (cfr. figs. 3.25 (d), (e) and (f)) that might produce false minima, the global minima of KLID and JID are clearly identified.

The same considerations hold for fig. 3.24, where Bumps is decomposed using SWT with db2 wavelet. It can be noted in fig. 3.26 that in the SWT case, due to the greater number of small coefficients with respect to DWT, numerical instabilities around the last coefficients are more present.

The performances of the proposed methods for Blocks, for different noise strengths, are depicted in fig. 3.29 for the case of DWT decomposition using wavelet *db*2. It has already been observed that, for Blocks, KLID and JID selects too few coefficients if no noise is present (fig. 3.29 (a)). However, when noise is added, the selected number of coefficients is close to the optimum (figs. 3.29 (b), (c), (d), (e) and (f)), as in the case of Bumps.

In fig. 3.27 (a) it can be seen that KLID and JID show less smooth profiles if no noise is present, are less smooth and Blocks is selected rather Bumps. This is due to the large amount of small coefficients. In addition, the behaviour of KLID and JID is the same that has been observed for Bumps.

The same can be observed in fig. 3.30, when SWT decomposition with wavelet db2 is considered. Moreover, fig. 3.28 depicts MSE values provided by KLID and JID for Blocks for different noise strengths. As for Bumps, numerical error toward the first and the last coefficients is more present in this case than when DWT is considered.

As a last comment, if SWT is employed, the number of selected coefficients are less than or equal to the optimum. It means that the reconstruction error is due to the oversmoothing of the underlying function, and not to the presence of residual noise.

The results of the denoising of Bumps for KLID, JID and Donoho thresholds using SWT are shown in fig. 3.31, while in fig. 3.32 the outcomes of the denoising of Heavy Sine using KLID, JID and Donoho thresholds with SWT decomposition are depicted.

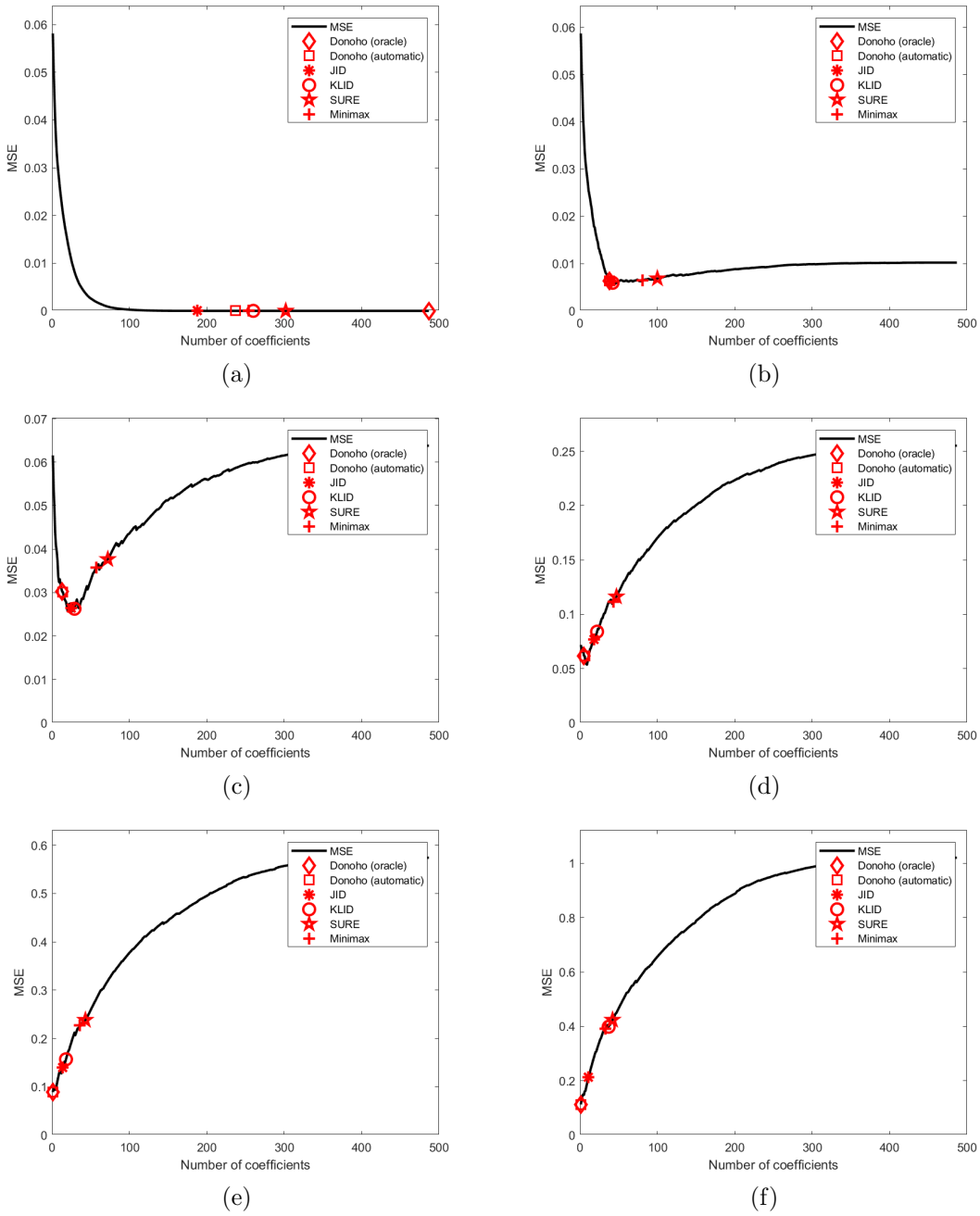


Figure 3.23: Function Bumps, decomposed using DWT with db2 wavelet: approximation MSE comparison with existing methods for noise variance 0 (a), 0.1 (b), 0.25 (c), 0.5 (d), 0.75 (e), 1 (f).

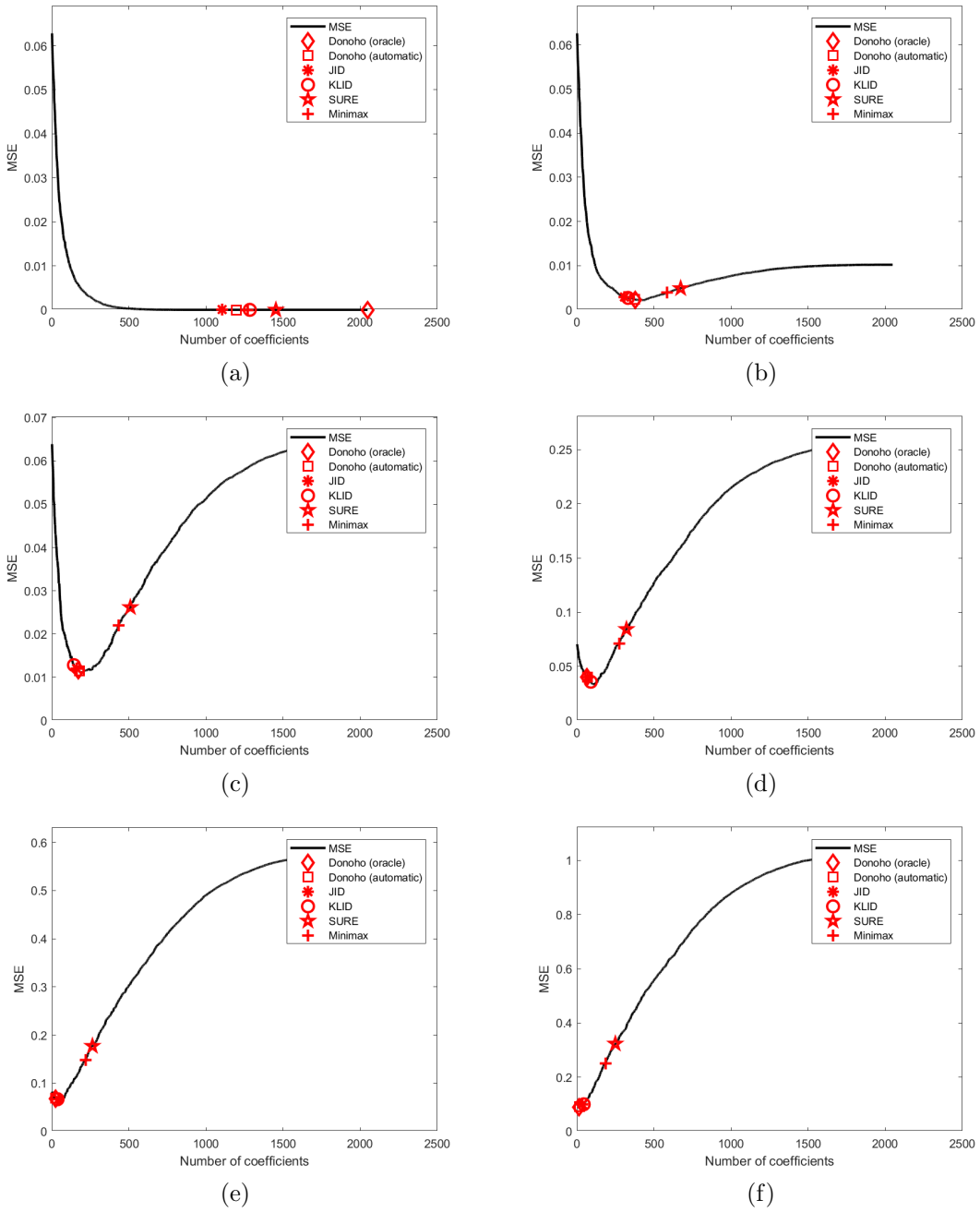


Figure 3.24: Function Bumps, decomposed using SWT with db2 wavelet: approximation MSE comparison with existing methods for noise variance 0 (a), 0.1 (b), 0.25 (c), 0.5 (d), 0.75 (e), 1 (f).

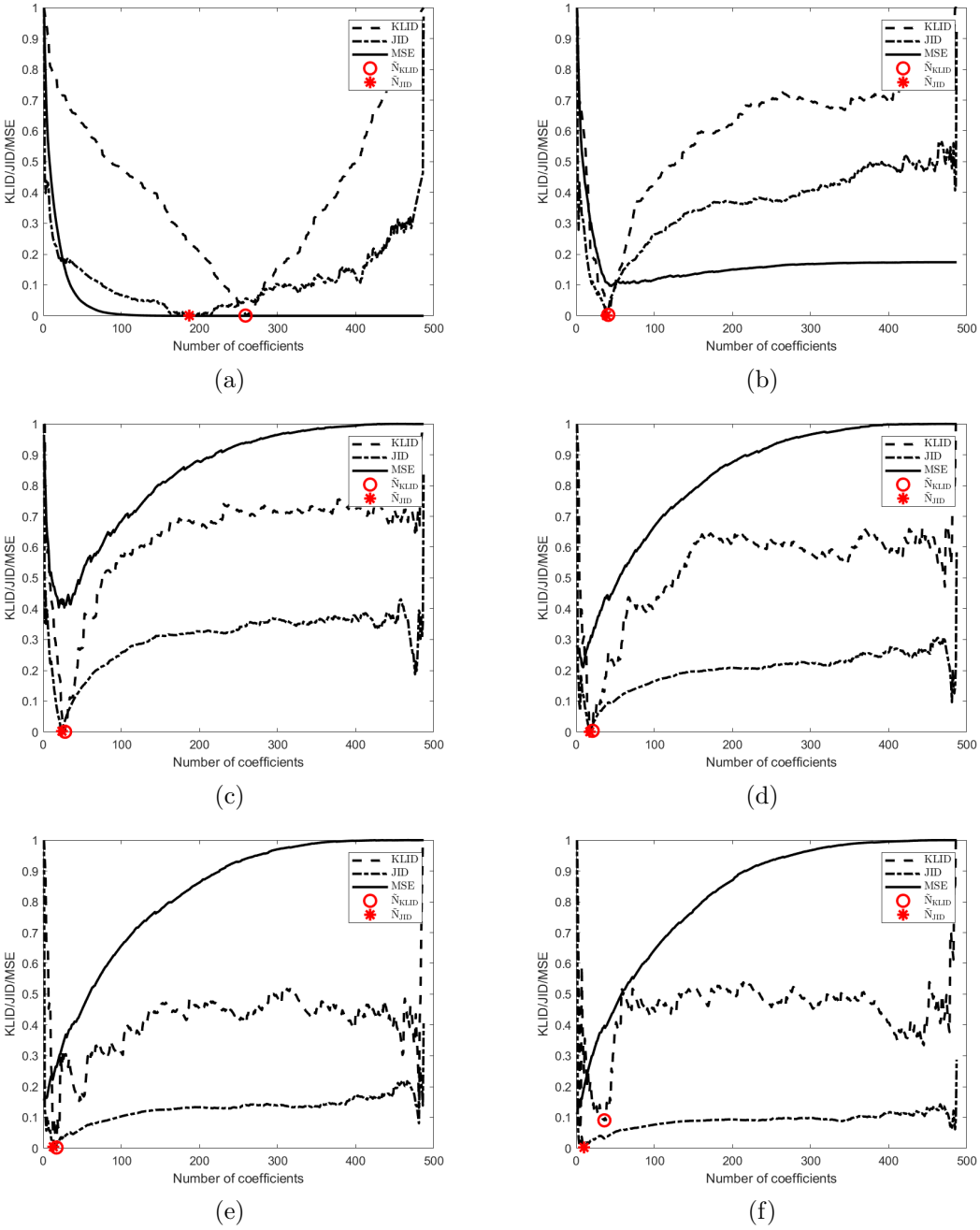


Figure 3.25: Function Bumps, decomposed using DWT with db2 wavelet: MSE, KLID and JID graphs for noise variance 0 (a), 0.1 (b), 0.25 (c), 0.5 (d), 0.75 (e), 1 (f). The plots have been rescaled for illustration purposes.

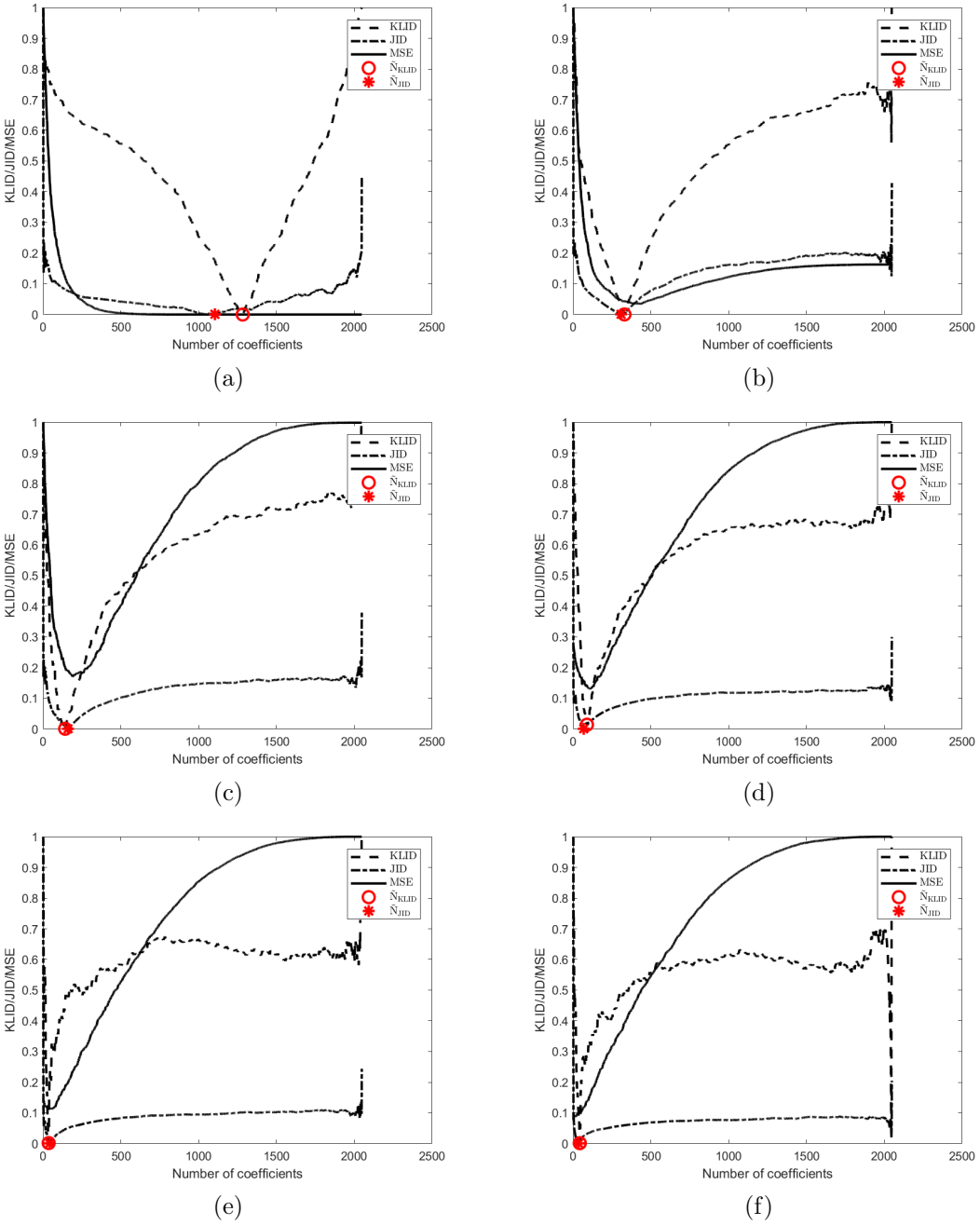


Figure 3.26: Function Bumps, decomposed using SWT with db2 wavelet: MSE, KLID and JID graphs for noise variance 0 (a), 0.1 (b), 0.25 (c), 0.5 (d), 0.75 (e), 1 (f). The plots have been rescaled for illustration purposes.

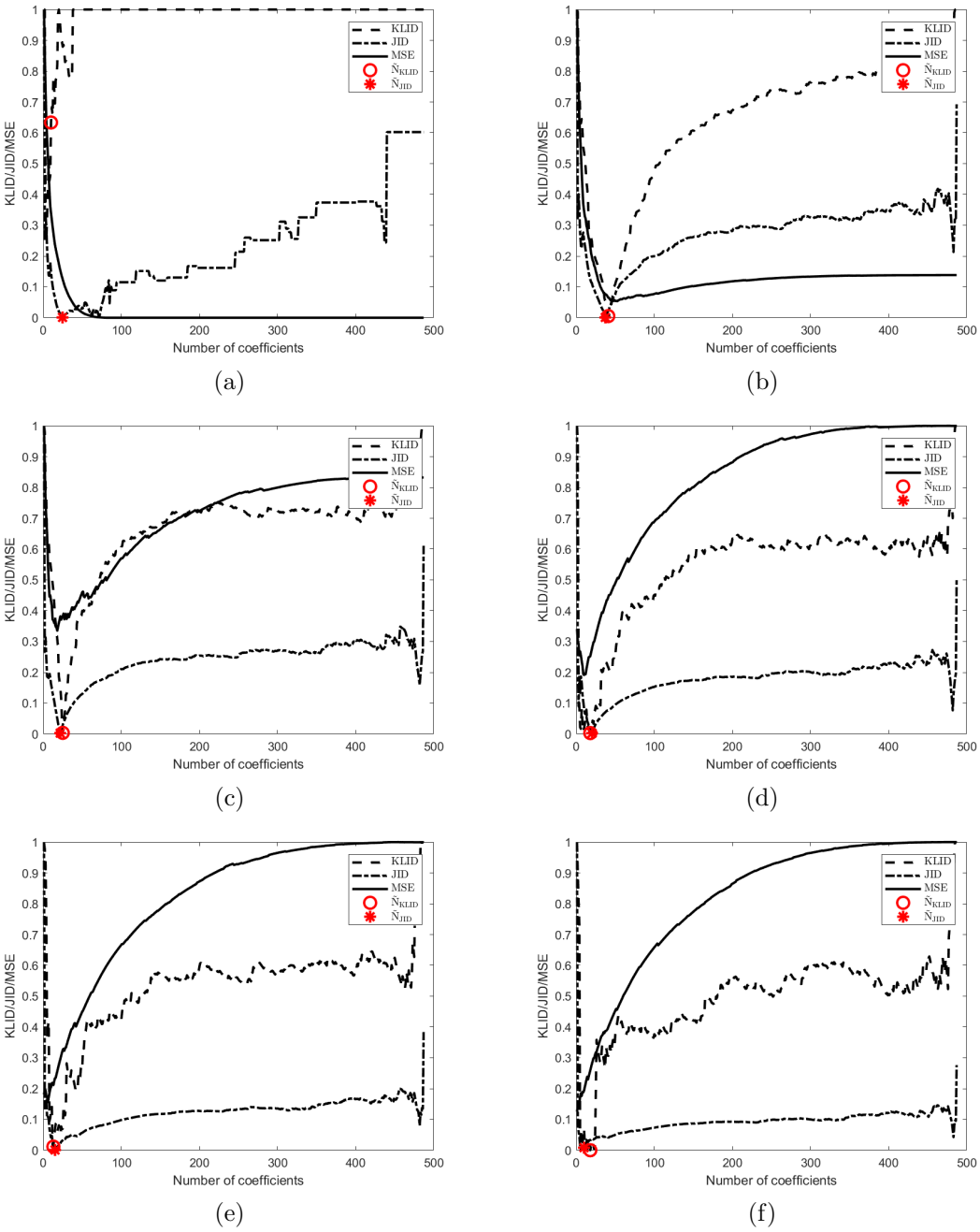


Figure 3.27: Function Blocks, decomposed using DWT with db2 wavelet: MSE, KLID and JID graphs for noise variance 0 (a), 0.1 (b), 0.25 (c), 0.5 (d), 0.75 (e), 1 (f). The plots have been rescaled for illustration purposes.

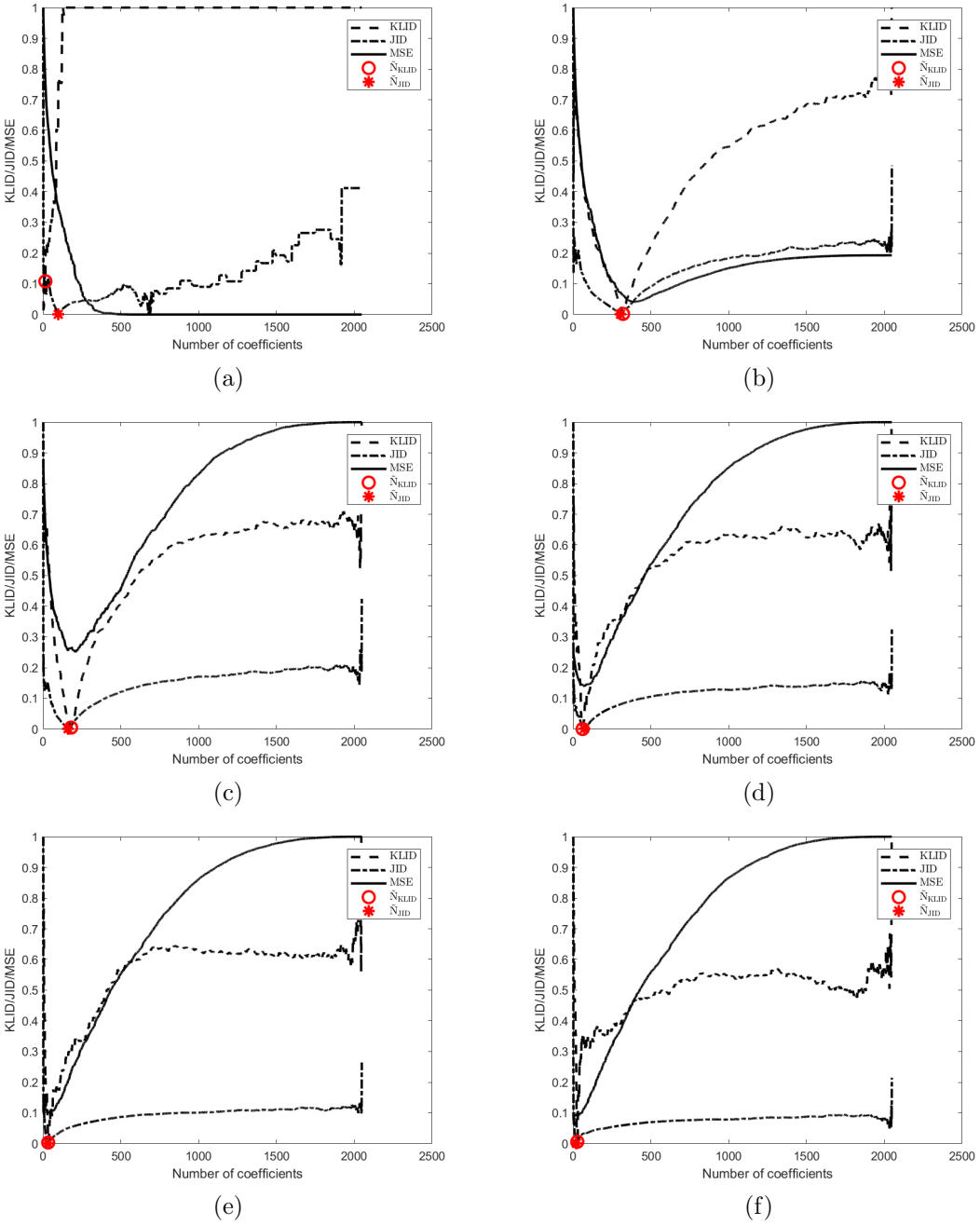


Figure 3.28: Function Blocks, decomposed using SWT with db2 wavelet: MSE, KLID and JID graphs for noise variance 0 (a), 0.1 (b), 0.25 (c), 0.5 (d), 0.75 (e), 1 (f). The plots have been rescaled for illustration purposes.

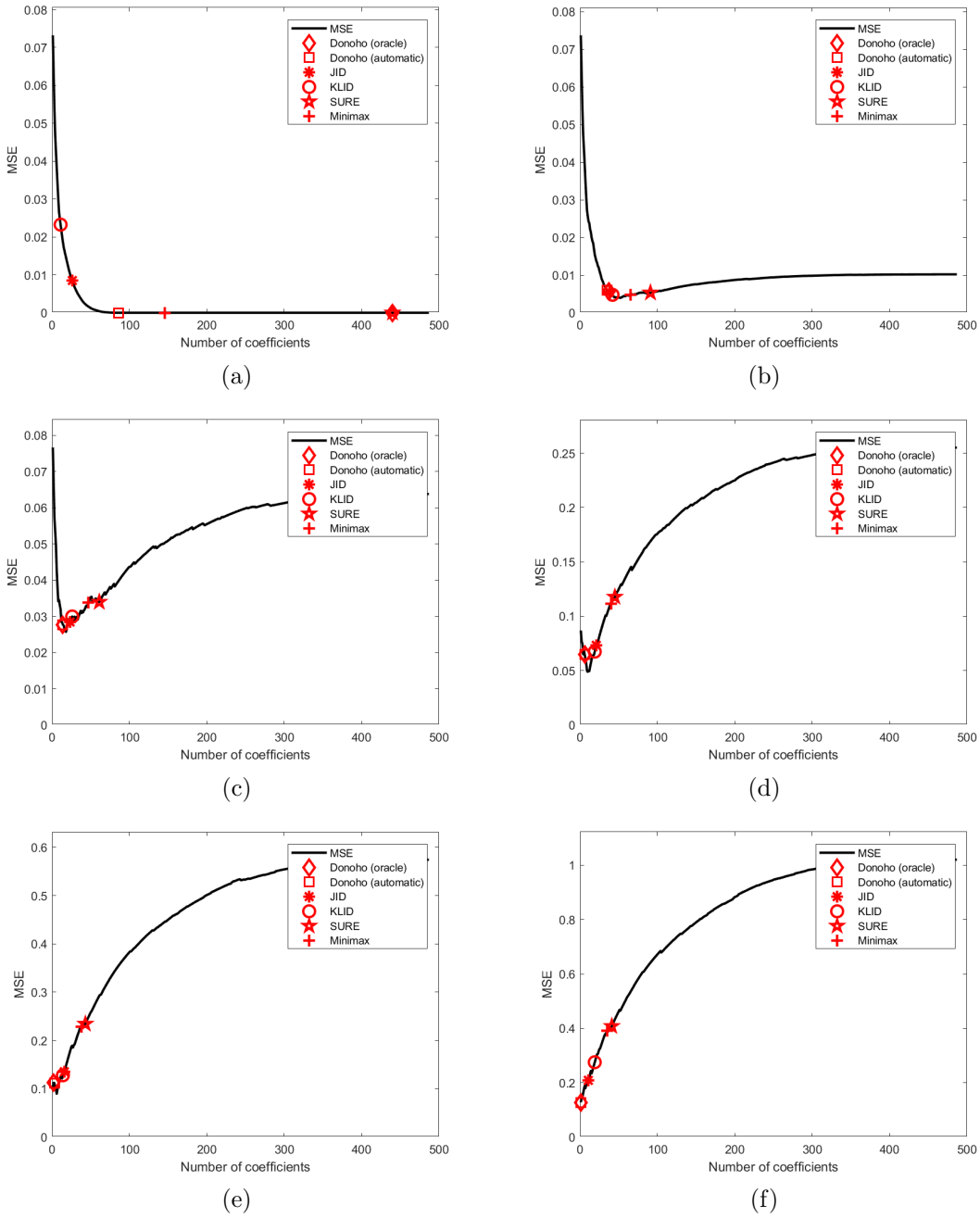


Figure 3.29: Function Blocks, decomposed using DWT with db2 wavelet: comparison with existing methods through MSE curve for noise variance 0 (a), 0.1 (b), 0.25 (c), 0.5 (d), 0.75 (e), 1 (f)

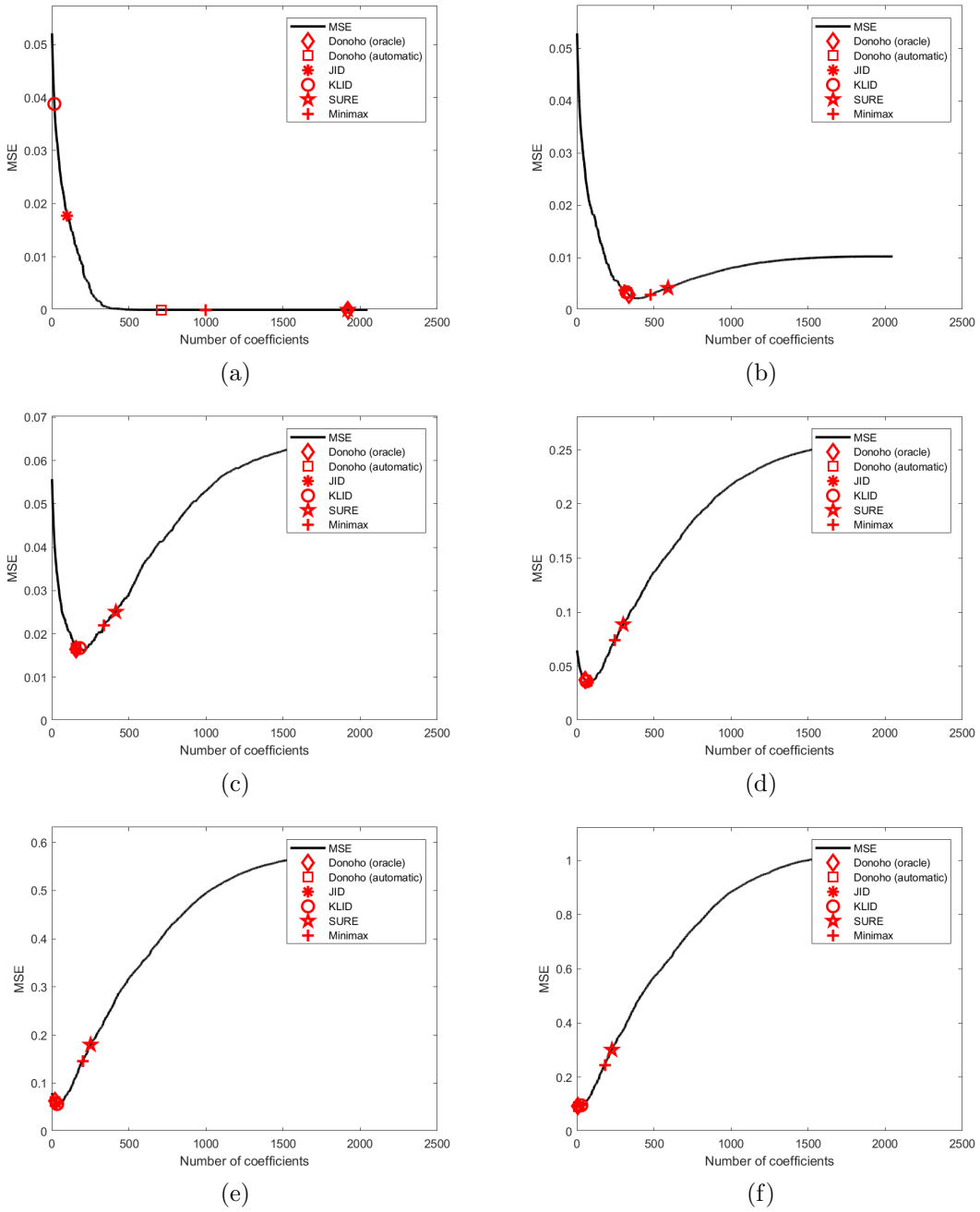


Figure 3.30: Function Blocks, decomposed using SWT with db2 wavelet: comparison with existing methods through MSE curve for noise variance 0 (a), 0.1 (b), 0.25 (c), 0.5 (d), 0.75 (e), 1 (f)

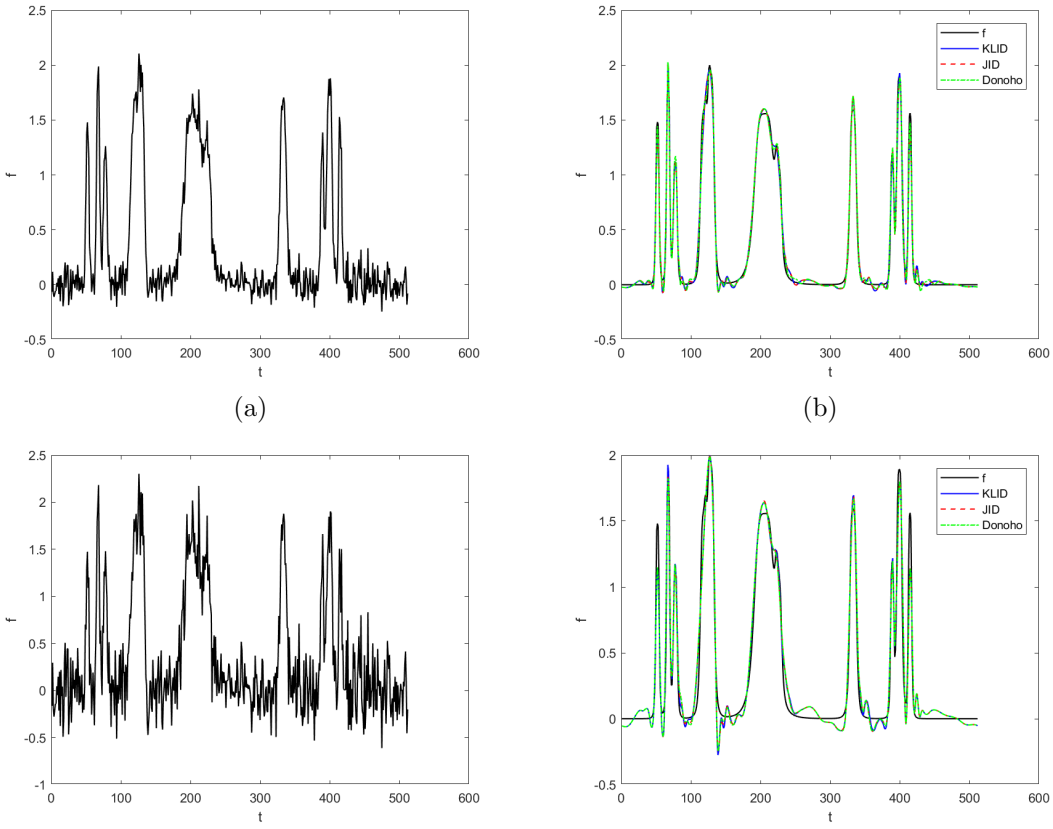


Figure 3.31: Function Bumps corrupted with additive noise, decomposed using SWT with db4 wavelet, denoised with KLID, JID and Donoho thresholds. Corrupted and denoised function for noise strength $\sigma = 0.1$ (a & b) and $\sigma = 0.25$ (c & d).

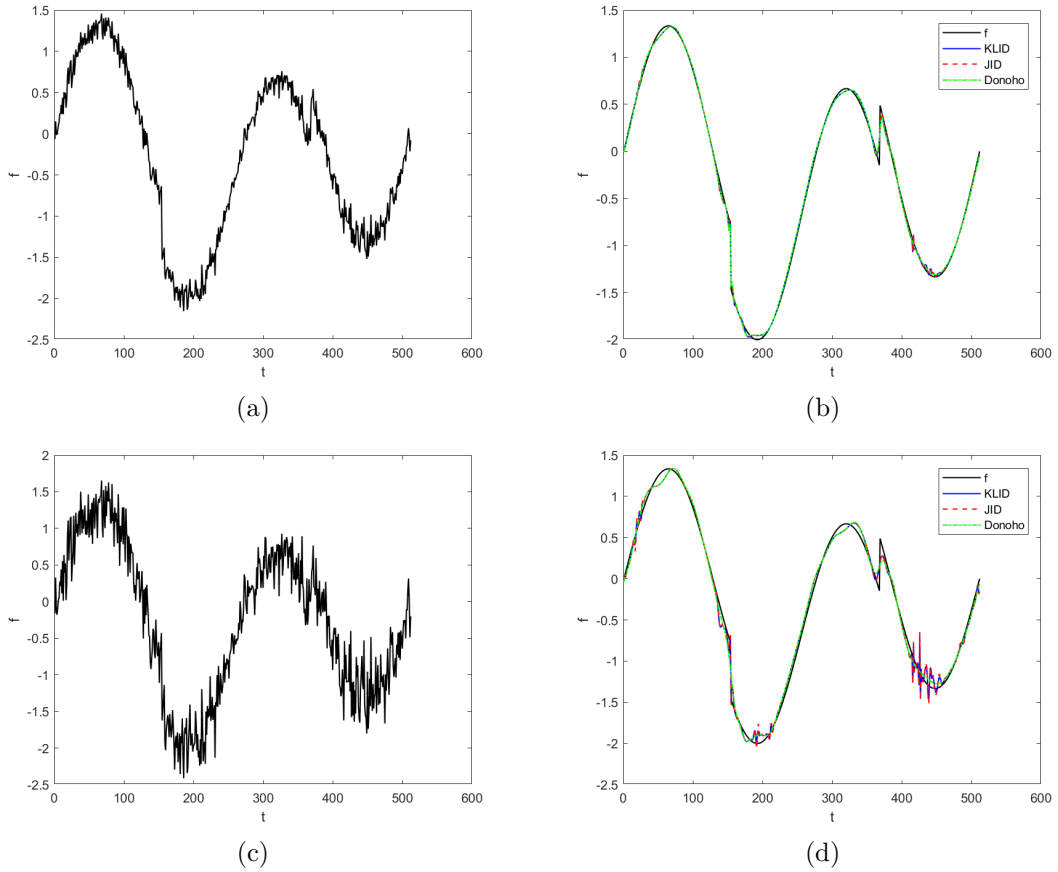


Figure 3.32: Function Heavy Sine corrupted with additive noise, decomposed using SWT with db4 wavelet, denoised with KLID, JID and Donoho thresholds. Corrupted and denoised function for noise strength $\sigma = 0.1$ (a & b) and $\sigma = 0.25$ (c & d).

	MSE						
σ	KLID	JID	D (R)	D (E)	SURE	Minimax	Opt.
	db2						
0.1	4.77×10^{-3}	5.56×10^{-3}	5.68×10^{-3}	5.92×10^{-3}	5.32×10^{-3}	4.74×10^{-3}	3.91×10^{-3}
0.25	2.99×10^{-2}	2.84×10^{-2}	2.76×10^{-2}	2.77×10^{-2}	3.39×10^{-2}	3.37×10^{-2}	2.57×10^{-2}
0.5	7.42×10^{-2}	7.31×10^{-2}	6.48×10^{-2}	6.48×10^{-2}	1.18×10^{-1}	1.11×10^{-1}	4.88×10^{-2}
0.75	1.28×10^{-1}	1.35×10^{-1}	1.17×10^{-1}	1.12×10^{-1}	2.34×10^{-1}	2.28×10^{-1}	8.96×10^{-2}
1	2.75×10^{-1}	2.1×10^{-1}	1.28×10^{-1}	1.28×10^{-1}	4.08×10^{-1}	3.91×10^{-1}	1.28×10^{-1}
	db4						
0.1	5.57×10^{-3}	5.51×10^{-3}	7.09×10^{-3}	7.65×10^{-3}	6.08×10^{-3}	5.27×10^{-3}	5.27×10^{-3}
0.25	2.35×10^{-2}	2.50×10^{-2}	2.73×10^{-2}	2.73×10^{-2}	3.59×10^{-2}	3.01×10^{-2}	2.25×10^{-2}
0.5	9.98×10^{-2}	8.17×10^{-2}	5.83×10^{-2}	5.83×10^{-2}	7.14×10^{-2}	9.58×10^{-2}	5.41×10^{-2}
0.75	2.52×10^{-1}	1.78×10^{-1}	8.11×10^{-2}	8.11×10^{-2}	1.40×10^{-1}	1.97×10^{-1}	7.27×10^{-2}
1	2.14×10^{-1}	3.03×10^{-1}	9.49×10^{-2}	9.49×10^{-2}	2.26×10^{-1}	3.31×10^{-1}	9.49×10^{-2}
	sym4						
0.10	1.18×10^{-2}	7.94×10^{-3}	5.13×10^{-3}	6.92×10^{-3}	5.09×10^{-3}	4.63×10^{-3}	4.05×10^{-3}
0.25	2.29×10^{-2}	2.40×10^{-2}	2.57×10^{-2}	2.58×10^{-2}	2.64×10^{-2}	2.39×10^{-2}	2.23×10^{-2}
0.50	8.06×10^{-2}	6.03×10^{-2}	4.70×10^{-2}	4.70×10^{-2}	8.06×10^{-2}	8.76×10^{-2}	4.70×10^{-2}
0.75	1.90×10^{-1}	1.34×10^{-1}	8.75×10^{-2}	8.75×10^{-2}	1.77×10^{-1}	1.67×10^{-1}	8.06×10^{-2}
1.00	2.39×10^{-1}	2.35×10^{-1}	1.18×10^{-1}	1.18×10^{-1}	3.20×10^{-1}	3.02×10^{-1}	1.14×10^{-1}
	db12						
0.10	1.50×10^{-2}	1.35×10^{-2}	1.16×10^{-2}	1.33×10^{-2}	7.95×10^{-3}	7.29×10^{-3}	7.13×10^{-3}
0.25	3.01×10^{-2}	3.01×10^{-2}	3.03×10^{-2}	3.00×10^{-2}	3.11×10^{-2}	3.73×10^{-2}	2.89×10^{-2}
0.50	7.26×10^{-2}	6.69×10^{-2}	5.00×10^{-2}	5.68×10^{-2}	1.43×10^{-1}	1.20×10^{-1}	5.00×10^{-2}
0.75	9.86×10^{-2}	1.25×10^{-1}	8.15×10^{-2}	7.63×10^{-2}	2.95×10^{-1}	2.45×10^{-1}	7.63×10^{-2}
1.00	3.22×10^{-1}	2.46×10^{-1}	1.09×10^{-1}	1.09×10^{-1}	6.52×10^{-1}	3.89×10^{-1}	1.05×10^{-1}
	sym12						
0.10	8.70×10^{-3}	1.08×10^{-2}	6.52×10^{-3}	8.00×10^{-3}	7.97×10^{-3}	5.78×10^{-3}	5.14×10^{-3}
0.25	3.06×10^{-2}	2.91×10^{-2}	3.10×10^{-2}	3.25×10^{-2}	2.59×10^{-2}	3.23×10^{-2}	2.53×10^{-2}
0.50	6.87×10^{-2}	6.77×10^{-2}	5.67×10^{-2}	5.67×10^{-2}	1.34×10^{-1}	1.07×10^{-1}	5.61×10^{-2}
0.75	1.45×10^{-1}	1.28×10^{-1}	8.07×10^{-2}	8.07×10^{-2}	2.73×10^{-1}	2.13×10^{-1}	7.66×10^{-2}
1.00	2.56×10^{-1}	2.09×10^{-1}	1.08×10^{-1}	1.08×10^{-1}	1.41×10^{-1}	3.50×10^{-1}	1.07×10^{-1}

Table 3.8: Reconstruction error for function Blocks decomposed using DWT. D (R) and D (E) stand for Donoho universal threshold with, respectively, real noise variance and estimated noise variance, Opt. stands for optimum threshold and σ is the noise variance.

	MSE						
σ	KLID	JID	D (R)	D (E)	SURE	Minimax	Opt.
	db2						
0.10	3.35×10^{-3}	3.74×10^{-3}	2.87×10^{-3}	3.28×10^{-3}	4.17×10^{-3}	2.83×10^{-3}	2.17×10^{-3}
0.25	1.67×10^{-2}	1.63×10^{-2}	1.64×10^{-2}	1.67×10^{-2}	2.51×10^{-2}	2.20×10^{-2}	1.59×10^{-2}
0.50	3.63×10^{-2}	3.56×10^{-2}	3.74×10^{-2}	3.74×10^{-2}	8.87×10^{-2}	7.41×10^{-2}	3.54×10^{-2}
0.75	5.67×10^{-2}	5.77×10^{-2}	6.33×10^{-2}	6.16×10^{-2}	1.80×10^{-1}	1.46×10^{-1}	5.64×10^{-2}
1.00	9.62×10^{-2}	9.86×10^{-2}	9.27×10^{-2}	9.13×10^{-2}	3.01×10^{-1}	2.44×10^{-1}	9.04×10^{-2}
	db4						
0.10	8.45×10^{-3}	8.42×10^{-3}	3.79×10^{-3}	4.10×10^{-3}	4.38×10^{-3}	3.54×10^{-3}	3.29×10^{-3}
0.25	1.86×10^{-2}	2.08×10^{-2}	1.96×10^{-2}	1.98×10^{-2}	2.72×10^{-2}	2.18×10^{-2}	1.72×10^{-2}
0.50	4.49×10^{-2}	3.85×10^{-2}	3.98×10^{-2}	3.97×10^{-2}	5.09×10^{-2}	6.72×10^{-2}	3.83×10^{-2}
0.75	7.55×10^{-2}	6.64×10^{-2}	6.51×10^{-2}	6.51×10^{-2}	8.76×10^{-2}	1.27×10^{-1}	6.18×10^{-2}
1.00	9.12×10^{-2}	9.50×10^{-2}	9.46×10^{-2}	9.46×10^{-2}	1.40×10^{-1}	2.06×10^{-1}	8.92×10^{-2}
	sym4						
0.10	3.53×10^{-3}	4.08×10^{-3}	3.22×10^{-3}	4.05×10^{-3}	3.50×10^{-3}	2.93×10^{-3}	2.73×10^{-3}
0.25	1.78×10^{-2}	1.83×10^{-2}	1.90×10^{-2}	2.00×10^{-2}	2.01×10^{-2}	1.86×10^{-2}	1.64×10^{-2}
0.50	3.82×10^{-2}	3.64×10^{-2}	3.88×10^{-2}	3.98×10^{-2}	5.52×10^{-2}	5.98×10^{-2}	3.44×10^{-2}
0.75	6.60×10^{-2}	6.24×10^{-2}	6.65×10^{-2}	6.98×10^{-2}	1.22×10^{-1}	1.14×10^{-1}	6.14×10^{-2}
1.00	1.00×10^{-1}	9.89×10^{-2}	9.47×10^{-2}	9.54×10^{-2}	2.04×10^{-1}	1.91×10^{-1}	9.01×10^{-2}
	db12						
0.10	1.78×10^{-2}	1.68×10^{-2}	6.51×10^{-3}	8.67×10^{-3}	6.43×10^{-3}	5.02×10^{-3}	4.96×10^{-3}
0.25	2.67×10^{-2}	2.64×10^{-2}	2.43×10^{-2}	2.39×10^{-2}	2.20×10^{-2}	2.63×10^{-2}	2.17×10^{-2}
0.50	4.88×10^{-2}	4.90×10^{-2}	4.98×10^{-2}	4.90×10^{-2}	1.10×10^{-1}	8.63×10^{-2}	4.85×10^{-2}
0.75	7.62×10^{-2}	8.04×10^{-2}	7.45×10^{-2}	7.48×10^{-2}	2.23×10^{-1}	1.72×10^{-1}	7.38×10^{-2}
1.00	1.74×10^{-1}	1.21×10^{-1}	1.03×10^{-1}	1.05×10^{-1}	5.62×10^{-1}	2.68×10^{-1}	1.02×10^{-1}
	sym12						
0.10	9.49×10^{-3}	9.19×10^{-3}	4.40×10^{-3}	5.52×10^{-3}	6.48×10^{-3}	3.87×10^{-3}	3.49×10^{-3}
0.25	2.07×10^{-2}	2.11×10^{-2}	2.08×10^{-2}	2.10×10^{-2}	1.85×10^{-2}	2.20×10^{-2}	1.83×10^{-2}
0.50	4.08×10^{-2}	4.08×10^{-2}	4.23×10^{-2}	4.23×10^{-2}	8.81×10^{-2}	6.87×10^{-2}	3.87×10^{-2}
0.75	7.23×10^{-2}	6.86×10^{-2}	7.25×10^{-2}	7.30×10^{-2}	1.77×10^{-1}	1.32×10^{-1}	6.73×10^{-2}
1.00	1.04×10^{-1}	1.08×10^{-1}	9.79×10^{-2}	9.87×10^{-2}	1.08×10^{-1}	2.18×10^{-1}	9.78×10^{-2}

Table 3.9: Reconstruction error for function Blocks decomposed using SWT. D (R) and D (E) stand for Donoho universal threshold with, respectively, real noise variance and estimated noise variance, Opt. stands for optimum threshold and σ is the noise variance.

	MSE						
σ	KLID	JID	D (R)	D (E)	SURE	Minimax	Opt.
	db2						
0.10	5.94×10^{-3}	6.20×10^{-3}	6.36×10^{-3}	6.36×10^{-3}	6.84×10^{-3}	6.40×10^{-3}	5.67×10^{-3}
0.25	2.63×10^{-2}	2.65×10^{-2}	3.02×10^{-2}	3.01×10^{-2}	3.77×10^{-2}	3.57×10^{-2}	2.58×10^{-2}
0.50	8.40×10^{-2}	7.69×10^{-2}	6.17×10^{-2}	6.17×10^{-2}	1.16×10^{-1}	1.11×10^{-1}	5.36×10^{-2}
0.75	1.57×10^{-1}	1.40×10^{-1}	8.89×10^{-2}	8.89×10^{-2}	2.38×10^{-1}	2.27×10^{-1}	8.89×10^{-2}
1.00	3.99×10^{-1}	2.13×10^{-1}	1.13×10^{-1}	1.13×10^{-1}	4.24×10^{-1}	3.91×10^{-1}	1.13×10^{-1}
	db4						
0.10	6.62×10^{-3}	6.80×10^{-3}	5.08×10^{-3}	6.26×10^{-3}	3.51×10^{-3}	4.35×10^{-3}	3.44×10^{-3}
0.25	2.30×10^{-2}	2.40×10^{-2}	2.40×10^{-2}	2.40×10^{-2}	2.18×10^{-2}	2.67×10^{-2}	2.09×10^{-2}
0.50	6.49×10^{-2}	6.49×10^{-2}	5.84×10^{-2}	5.84×10^{-2}	5.64×10^{-2}	9.81×10^{-2}	5.18×10^{-2}
0.75	1.42×10^{-1}	1.42×10^{-1}	7.93×10^{-2}	7.93×10^{-2}	1.01×10^{-1}	2.01×10^{-1}	7.93×10^{-2}
1.00	4.34×10^{-1}	2.39×10^{-1}	1.12×10^{-1}	1.12×10^{-1}	1.37×10^{-1}	3.31×10^{-1}	1.03×10^{-1}
	sym4						
0.10	4.57×10^{-3}	4.67×10^{-3}	4.67×10^{-3}	5.48×10^{-3}	3.68×10^{-3}	3.59×10^{-3}	3.08×10^{-3}
0.25	2.04×10^{-2}	1.83×10^{-2}	2.07×10^{-2}	2.19×10^{-2}	2.24×10^{-2}	2.18×10^{-2}	1.71×10^{-2}
0.50	7.67×10^{-2}	7.35×10^{-2}	5.73×10^{-2}	6.43×10^{-2}	8.18×10^{-2}	8.39×10^{-2}	5.39×10^{-2}
0.75	1.33×10^{-1}	1.33×10^{-1}	1.04×10^{-1}	1.04×10^{-1}	1.71×10^{-1}	1.66×10^{-1}	8.82×10^{-2}
1.00	2.39×10^{-1}	2.26×10^{-1}	1.36×10^{-1}	1.48×10^{-1}	2.77×10^{-1}	2.87×10^{-1}	1.33×10^{-1}
	db12						
0.10	5.31×10^{-3}	5.35×10^{-3}	5.51×10^{-3}	5.47×10^{-3}	4.63×10^{-3}	5.83×10^{-3}	4.50×10^{-3}
0.25	2.94×10^{-2}	2.21×10^{-2}	2.42×10^{-2}	2.39×10^{-2}	2.08×10^{-2}	3.10×10^{-2}	2.01×10^{-2}
0.50	7.38×10^{-2}	7.74×10^{-2}	6.09×10^{-2}	6.09×10^{-2}	7.04×10^{-2}	1.07×10^{-1}	6.09×10^{-2}
0.75	1.41×10^{-1}	1.41×10^{-1}	9.69×10^{-2}	9.20×10^{-2}	1.07×10^{-1}	2.52×10^{-1}	9.20×10^{-2}
1.00	3.73×10^{-1}	2.81×10^{-1}	1.28×10^{-1}	1.26×10^{-1}	1.58×10^{-1}	4.11×10^{-1}	1.22×10^{-1}
	sym12						
0.10	4.87×10^{-3}	5.16×10^{-3}	4.87×10^{-3}	5.33×10^{-3}	5.91×10^{-3}	5.20×10^{-3}	3.61×10^{-3}
0.25	1.80×10^{-2}	1.79×10^{-2}	2.05×10^{-2}	2.22×10^{-2}	3.19×10^{-2}	2.66×10^{-2}	1.67×10^{-2}
0.50	7.01×10^{-2}	6.07×10^{-2}	6.52×10^{-2}	6.52×10^{-2}	5.33×10^{-2}	9.98×10^{-2}	4.96×10^{-2}
0.75	5.74×10^{-1}	1.38×10^{-1}	8.33×10^{-2}	8.33×10^{-2}	1.03×10^{-1}	2.07×10^{-1}	8.33×10^{-2}
1.00	2.67×10^{-1}	2.24×10^{-1}	1.13×10^{-1}	1.13×10^{-1}	1.38×10^{-1}	3.44×10^{-1}	1.09×10^{-1}

Table 3.10: Reconstruction error for function Bumps decomposed using DWT. D (R) and D (E) stand for Donoho universal threshold with, respectively, real noise variance and estimated noise variance, Opt. stands for optimum threshold and σ is the noise variance.

	MSE						
σ	KLID	JID	D (R)	D (E)	SURE	Minimax	Opt.
	db2						
0.10	2.70×10^{-3}	2.95×10^{-3}	2.29×10^{-3}	2.37×10^{-3}	4.86×10^{-3}	3.87×10^{-3}	2.18×10^{-3}
0.25	1.28×10^{-2}	1.21×10^{-2}	1.16×10^{-2}	1.15×10^{-2}	2.62×10^{-2}	2.20×10^{-2}	1.09×10^{-2}
0.50	3.59×10^{-2}	4.02×10^{-2}	4.04×10^{-2}	4.01×10^{-2}	8.43×10^{-2}	7.09×10^{-2}	3.33×10^{-2}
0.75	6.68×10^{-2}	6.61×10^{-2}	6.79×10^{-2}	6.90×10^{-2}	1.77×10^{-1}	1.48×10^{-1}	6.35×10^{-2}
1.00	9.99×10^{-2}	9.95×10^{-2}	9.01×10^{-2}	9.15×10^{-2}	3.23×10^{-1}	2.51×10^{-1}	8.96×10^{-2}
	db4						
0.10	3.90×10^{-3}	3.41×10^{-3}	2.35×10^{-3}	2.66×10^{-3}	2.26×10^{-3}	2.95×10^{-3}	2.20×10^{-3}
0.25	1.12×10^{-2}	1.11×10^{-2}	1.12×10^{-2}	1.21×10^{-2}	1.11×10^{-2}	1.59×10^{-2}	1.07×10^{-2}
0.50	3.38×10^{-2}	3.46×10^{-2}	3.82×10^{-2}	3.94×10^{-2}	3.37×10^{-2}	5.63×10^{-2}	3.31×10^{-2}
0.75	6.61×10^{-2}	6.50×10^{-2}	6.59×10^{-2}	6.60×10^{-2}	6.39×10^{-2}	1.17×10^{-1}	6.29×10^{-2}
1.00	8.65×10^{-2}	8.86×10^{-2}	9.27×10^{-2}	9.58×10^{-2}	8.97×10^{-2}	1.94×10^{-1}	8.63×10^{-2}
	sym4						
0.10	3.28×10^{-3}	3.16×10^{-3}	2.38×10^{-3}	2.73×10^{-3}	2.19×10^{-3}	2.79×10^{-3}	2.19×10^{-3}
0.25	1.20×10^{-2}	1.12×10^{-2}	1.05×10^{-2}	1.05×10^{-2}	1.65×10^{-2}	1.53×10^{-2}	9.60×10^{-3}
0.50	3.62×10^{-2}	3.65×10^{-2}	4.15×10^{-2}	4.56×10^{-2}	5.49×10^{-2}	5.10×10^{-2}	3.30×10^{-2}
0.75	7.09×10^{-2}	6.91×10^{-2}	6.95×10^{-2}	7.15×10^{-2}	1.13×10^{-1}	1.08×10^{-1}	6.56×10^{-2}
1.00	9.84×10^{-2}	9.92×10^{-2}	9.65×10^{-2}	9.59×10^{-2}	1.75×10^{-1}	1.86×10^{-1}	9.24×10^{-2}
	db12						
0.10	4.63×10^{-3}	4.84×10^{-3}	3.57×10^{-3}	3.53×10^{-3}	3.44×10^{-3}	4.05×10^{-3}	3.03×10^{-3}
0.25	2.00×10^{-2}	1.64×10^{-2}	1.49×10^{-2}	1.42×10^{-2}	1.43×10^{-2}	2.09×10^{-2}	1.34×10^{-2}
0.50	4.47×10^{-2}	4.33×10^{-2}	4.37×10^{-2}	4.27×10^{-2}	4.00×10^{-2}	6.63×10^{-2}	3.90×10^{-2}
0.75	6.67×10^{-2}	6.81×10^{-2}	6.88×10^{-2}	6.78×10^{-2}	7.37×10^{-2}	1.58×10^{-1}	6.46×10^{-2}
1.00	1.79×10^{-1}	1.18×10^{-1}	1.01×10^{-1}	9.38×10^{-2}	1.14×10^{-1}	2.64×10^{-1}	8.94×10^{-2}
	sym12						
0.10	3.07×10^{-3}	3.15×10^{-3}	2.82×10^{-3}	2.98×10^{-3}	4.12×10^{-3}	3.26×10^{-3}	2.54×10^{-3}
0.25	1.13×10^{-2}	1.13×10^{-2}	1.12×10^{-2}	1.14×10^{-2}	2.24×10^{-2}	1.62×10^{-2}	1.10×10^{-2}
0.50	3.69×10^{-2}	3.64×10^{-2}	4.17×10^{-2}	4.41×10^{-2}	3.48×10^{-2}	5.88×10^{-2}	3.41×10^{-2}
0.75	7.30×10^{-2}	7.06×10^{-2}	7.21×10^{-2}	7.26×10^{-2}	7.09×10^{-2}	1.25×10^{-1}	6.98×10^{-2}
1.00	1.52×10^{-1}	1.31×10^{-1}	9.91×10^{-2}	1.01×10^{-1}	1.07×10^{-1}	2.13×10^{-1}	9.61×10^{-2}

Table 3.11: Reconstruction error for function Bumps decomposed using SWT. D (R) and D (E) stand for Donoho universal threshold with, respectively, real noise variance and estimated noise variance, Opt. stands for optimum threshold and σ is the noise variance.

	MSE						
σ	KLID	JID	D (R)	D (E)	SURE	Minimax	Opt.
	db2						
0.10	1.34×10^{-3}	1.31×10^{-3}	2.20×10^{-3}	2.20×10^{-3}	4.54×10^{-3}	4.00×10^{-3}	1.28×10^{-3}
0.25	1.26×10^{-2}	1.26×10^{-2}	7.42×10^{-3}	6.74×10^{-3}	9.42×10^{-3}	2.33×10^{-2}	6.50×10^{-3}
0.50	4.73×10^{-2}	7.20×10^{-2}	1.76×10^{-2}	1.76×10^{-2}	9.43×10^{-2}	8.33×10^{-2}	1.76×10^{-2}
0.75	2.25×10^{-1}	1.57×10^{-1}	3.46×10^{-2}	3.46×10^{-2}	2.07×10^{-1}	1.83×10^{-1}	3.46×10^{-2}
1.00	5.36×10^{-1}	2.76×10^{-1}	5.84×10^{-2}	5.84×10^{-2}	3.64×10^{-1}	3.20×10^{-1}	5.84×10^{-2}
	db4						
0.10	2.40×10^{-3}	2.23×10^{-3}	1.96×10^{-3}	1.96×10^{-3}	2.62×10^{-3}	3.30×10^{-3}	1.96×10^{-3}
0.25	1.29×10^{-2}	1.29×10^{-2}	6.44×10^{-3}	6.44×10^{-3}	6.58×10^{-3}	2.06×10^{-2}	5.43×10^{-3}
0.50	1.16×10^{-1}	6.93×10^{-2}	1.59×10^{-2}	1.59×10^{-2}	4.00×10^{-2}	7.57×10^{-2}	1.59×10^{-2}
0.75	2.49×10^{-1}	1.58×10^{-1}	3.17×10^{-2}	3.17×10^{-2}	8.60×10^{-2}	1.68×10^{-1}	3.17×10^{-2}
1.00	4.57×10^{-1}	2.79×10^{-1}	5.38×10^{-2}	5.38×10^{-2}	1.50×10^{-1}	2.96×10^{-1}	5.38×10^{-2}
	sym4						
0.10	2.01×10^{-3}	2.01×10^{-3}	1.47×10^{-3}	1.89×10^{-3}	3.21×10^{-3}	3.03×10^{-3}	1.38×10^{-3}
0.25	1.30×10^{-2}	1.64×10^{-2}	6.13×10^{-3}	6.13×10^{-3}	1.76×10^{-2}	1.56×10^{-2}	5.42×10^{-3}
0.50	6.41×10^{-2}	7.72×10^{-2}	1.64×10^{-2}	1.64×10^{-2}	6.41×10^{-2}	5.97×10^{-2}	1.64×10^{-2}
0.75	1.77×10^{-1}	1.64×10^{-1}	3.34×10^{-2}	3.34×10^{-2}	1.40×10^{-1}	1.30×10^{-1}	3.34×10^{-2}
1.00	3.88×10^{-1}	2.91×10^{-1}	5.73×10^{-2}	5.73×10^{-2}	2.46×10^{-1}	2.28×10^{-1}	5.73×10^{-2}
	db12						
0.10	2.79×10^{-3}	2.80×10^{-3}	2.71×10^{-3}	2.71×10^{-3}	6.77×10^{-3}	4.16×10^{-3}	2.44×10^{-3}
0.25	4.16×10^{-2}	1.20×10^{-2}	6.43×10^{-3}	6.43×10^{-3}	8.25×10^{-3}	2.21×10^{-2}	6.43×10^{-3}
0.50	1.02×10^{-1}	6.24×10^{-2}	1.66×10^{-2}	1.66×10^{-2}	2.66×10^{-2}	8.54×10^{-2}	1.65×10^{-2}
0.75	1.57×10^{-1}	1.14×10^{-1}	3.34×10^{-2}	3.34×10^{-2}	5.64×10^{-2}	1.85×10^{-1}	3.32×10^{-2}
1.00	3.34×10^{-1}	1.92×10^{-1}	5.68×10^{-2}	5.68×10^{-2}	9.00×10^{-2}	3.64×10^{-1}	5.66×10^{-2}
	sym12						
0.10	2.82×10^{-3}	3.15×10^{-3}	1.81×10^{-3}	1.81×10^{-3}	1.97×10^{-3}	3.32×10^{-3}	1.67×10^{-3}
0.25	1.33×10^{-2}	1.05×10^{-2}	6.00×10^{-3}	6.56×10^{-3}	6.47×10^{-3}	1.90×10^{-2}	5.39×10^{-3}
0.50	8.77×10^{-2}	7.32×10^{-2}	1.64×10^{-2}	1.64×10^{-2}	2.14×10^{-2}	7.10×10^{-2}	1.64×10^{-2}
0.75	1.18×10^{-1}	1.12×10^{-1}	3.28×10^{-2}	3.28×10^{-2}	4.40×10^{-2}	1.62×10^{-1}	3.28×10^{-2}
1.00	1.87×10^{-1}	1.87×10^{-1}	5.58×10^{-2}	5.58×10^{-2}	7.56×10^{-2}	2.86×10^{-1}	5.58×10^{-2}

Table 3.12: Reconstruction error for function Heavy Sine decomposed using DWT. D (R) and D (E) stand for Donoho universal threshold with, respectively, real noise variance and estimated noise variance, Opt. stands for optimum threshold and σ is the noise variance.

	MSE						
σ	KLID	JID	D (R)	D (E)	SURE	Minimax	Opt.
	db2						
0.10	9.83×10^{-4}	9.83×10^{-4}	9.09×10^{-4}	9.09×10^{-4}	3.07×10^{-3}	2.42×10^{-3}	8.01×10^{-4}
0.25	5.67×10^{-3}	5.05×10^{-3}	4.68×10^{-3}	4.71×10^{-3}	5.19×10^{-3}	1.42×10^{-2}	4.58×10^{-3}
0.50	1.71×10^{-2}	1.93×10^{-2}	1.40×10^{-2}	1.40×10^{-2}	7.03×10^{-2}	5.42×10^{-2}	1.40×10^{-2}
0.75	5.09×10^{-2}	4.30×10^{-2}	2.79×10^{-2}	2.79×10^{-2}	1.55×10^{-1}	1.21×10^{-1}	2.79×10^{-2}
1.00	8.41×10^{-2}	7.53×10^{-2}	4.73×10^{-2}	4.73×10^{-2}	2.72×10^{-1}	2.14×10^{-1}	4.73×10^{-2}
	db4						
0.10	1.15×10^{-3}	1.14×10^{-3}	1.15×10^{-3}	1.17×10^{-3}	1.56×10^{-3}	1.98×10^{-3}	1.12×10^{-3}
0.25	6.26×10^{-3}	6.72×10^{-3}	5.00×10^{-3}	5.00×10^{-3}	4.31×10^{-3}	1.20×10^{-2}	4.30×10^{-3}
0.50	2.56×10^{-2}	2.42×10^{-2}	1.48×10^{-2}	1.48×10^{-2}	2.40×10^{-2}	4.44×10^{-2}	1.48×10^{-2}
0.75	7.34×10^{-2}	6.06×10^{-2}	2.98×10^{-2}	2.98×10^{-2}	5.20×10^{-2}	9.86×10^{-2}	2.98×10^{-2}
1.00	1.50×10^{-1}	1.03×10^{-1}	5.07×10^{-2}	5.07×10^{-2}	9.03×10^{-2}	1.74×10^{-1}	5.07×10^{-2}
	sym4						
0.10	1.04×10^{-3}	1.42×10^{-3}	9.70×10^{-4}	9.98×10^{-4}	2.04×10^{-3}	1.91×10^{-3}	9.70×10^{-4}
0.25	4.95×10^{-3}	7.16×10^{-3}	4.75×10^{-3}	5.23×10^{-3}	1.16×10^{-2}	1.06×10^{-2}	4.56×10^{-3}
0.50	3.10×10^{-2}	1.77×10^{-2}	1.48×10^{-2}	1.48×10^{-2}	4.36×10^{-2}	3.91×10^{-2}	1.48×10^{-2}
0.75	5.25×10^{-2}	4.33×10^{-2}	2.98×10^{-2}	2.98×10^{-2}	9.60×10^{-2}	8.58×10^{-2}	2.98×10^{-2}
1.00	7.47×10^{-2}	1.05×10^{-1}	5.07×10^{-2}	5.07×10^{-2}	1.69×10^{-1}	1.51×10^{-1}	5.07×10^{-2}
	db12						
0.10	1.79×10^{-3}	1.77×10^{-3}	1.83×10^{-3}	1.84×10^{-3}	5.92×10^{-3}	2.89×10^{-3}	1.72×10^{-3}
0.25	8.78×10^{-3}	7.08×10^{-3}	5.66×10^{-3}	5.66×10^{-3}	6.65×10^{-3}	1.39×10^{-2}	5.53×10^{-3}
0.50	2.04×10^{-2}	2.11×10^{-2}	1.56×10^{-2}	1.57×10^{-2}	2.16×10^{-2}	5.29×10^{-2}	1.56×10^{-2}
0.75	4.52×10^{-2}	4.41×10^{-2}	3.14×10^{-2}	3.16×10^{-2}	4.58×10^{-2}	1.17×10^{-1}	3.14×10^{-2}
1.00	8.27×10^{-2}	7.89×10^{-2}	5.34×10^{-2}	5.38×10^{-2}	7.89×10^{-2}	2.27×10^{-1}	5.34×10^{-2}
	sym12						
0.10	1.26×10^{-3}	1.34×10^{-3}	1.22×10^{-3}	1.23×10^{-3}	1.26×10^{-3}	2.10×10^{-3}	1.19×10^{-3}
0.25	7.66×10^{-3}	6.72×10^{-3}	5.38×10^{-3}	5.67×10^{-3}	5.51×10^{-3}	1.23×10^{-2}	4.97×10^{-3}
0.50	2.03×10^{-2}	1.86×10^{-2}	1.57×10^{-2}	1.57×10^{-2}	1.93×10^{-2}	4.55×10^{-2}	1.56×10^{-2}
0.75	3.92×10^{-2}	3.81×10^{-2}	3.16×10^{-2}	3.16×10^{-2}	3.99×10^{-2}	1.01×10^{-1}	3.14×10^{-2}
1.00	9.78×10^{-2}	6.63×10^{-2}	5.38×10^{-2}	5.34×10^{-2}	6.85×10^{-2}	1.76×10^{-1}	5.34×10^{-2}

Table 3.13: Reconstruction error for function Heavy Sine decomposed using SWT. D (R) and D (E) stand for Donoho universal threshold with, respectively, real noise variance and estimated noise variance, Opt. stands for optimum threshold and σ is the noise variance.

	MSE						
σ	KLID	JID	D (R)	D (E)	SURE	Minimax	Opt.
	db2						
0.10	3.26×10^{-3}	2.04×10^{-3}	8.86×10^{-4}	8.86×10^{-4}	3.79×10^{-3}	3.45×10^{-3}	8.86×10^{-4}
0.25	2.05×10^{-2}	1.19×10^{-2}	3.76×10^{-3}	3.76×10^{-3}	2.27×10^{-2}	1.96×10^{-2}	3.76×10^{-3}
0.50	9.63×10^{-2}	5.18×10^{-2}	1.40×10^{-2}	1.40×10^{-2}	8.84×10^{-2}	7.74×10^{-2}	1.40×10^{-2}
0.75	2.09×10^{-1}	1.27×10^{-1}	3.11×10^{-2}	3.11×10^{-2}	1.99×10^{-1}	1.74×10^{-1}	3.11×10^{-2}
1.00	3.90×10^{-1}	2.54×10^{-1}	5.51×10^{-2}	5.51×10^{-2}	3.53×10^{-1}	3.17×10^{-1}	5.51×10^{-2}
	db4						
0.10	2.37×10^{-3}	2.26×10^{-3}	1.10×10^{-3}	1.10×10^{-3}	2.03×10^{-3}	3.20×10^{-3}	1.07×10^{-3}
0.25	2.78×10^{-2}	1.63×10^{-2}	4.40×10^{-3}	4.40×10^{-3}	9.89×10^{-3}	1.90×10^{-2}	3.86×10^{-3}
0.50	1.13×10^{-1}	6.79×10^{-2}	1.39×10^{-2}	1.39×10^{-2}	3.80×10^{-2}	7.42×10^{-2}	1.39×10^{-2}
0.75	2.22×10^{-1}	1.56×10^{-1}	2.98×10^{-2}	2.98×10^{-2}	8.41×10^{-2}	1.66×10^{-1}	2.98×10^{-2}
1.00	4.34×10^{-1}	2.77×10^{-1}	5.20×10^{-2}	5.20×10^{-2}	1.48×10^{-1}	2.94×10^{-1}	5.20×10^{-2}
	sym4						
0.10	2.51×10^{-3}	1.84×10^{-3}	1.52×10^{-3}	1.46×10^{-3}	2.76×10^{-3}	2.59×10^{-3}	1.33×10^{-3}
0.25	1.90×10^{-2}	1.80×10^{-2}	4.46×10^{-3}	4.46×10^{-3}	1.65×10^{-2}	1.54×10^{-2}	4.46×10^{-3}
0.50	9.67×10^{-2}	7.91×10^{-2}	1.47×10^{-2}	1.47×10^{-2}	6.39×10^{-2}	5.95×10^{-2}	1.47×10^{-2}
0.75	2.10×10^{-1}	1.84×10^{-1}	3.17×10^{-2}	3.17×10^{-2}	1.43×10^{-1}	1.33×10^{-1}	3.17×10^{-2}
1.00	3.27×10^{-1}	3.13×10^{-1}	5.55×10^{-2}	5.55×10^{-2}	2.53×10^{-1}	2.35×10^{-1}	5.55×10^{-2}
	db12						
0.10	2.59×10^{-3}	2.05×10^{-3}	1.28×10^{-3}	1.31×10^{-3}	6.76×10^{-3}	4.03×10^{-3}	1.28×10^{-3}
0.25	2.01×10^{-2}	1.34×10^{-2}	4.37×10^{-3}	4.37×10^{-3}	4.87×10^{-3}	2.30×10^{-2}	4.36×10^{-3}
0.50	8.08×10^{-2}	4.57×10^{-2}	1.44×10^{-2}	1.44×10^{-2}	2.51×10^{-2}	9.03×10^{-2}	1.43×10^{-2}
0.75	1.38×10^{-1}	8.29×10^{-2}	3.10×10^{-2}	3.10×10^{-2}	5.00×10^{-2}	2.08×10^{-1}	3.10×10^{-2}
1.00	2.83×10^{-1}	1.58×10^{-1}	5.44×10^{-2}	5.44×10^{-2}	8.80×10^{-2}	3.70×10^{-1}	5.42×10^{-2}
	sym12						
0.10	2.37×10^{-3}	2.58×10^{-3}	1.11×10^{-3}	1.34×10^{-3}	1.34×10^{-3}	3.00×10^{-3}	9.57×10^{-4}
0.25	1.42×10^{-2}	1.42×10^{-2}	4.10×10^{-3}	4.10×10^{-3}	6.62×10^{-3}	1.85×10^{-2}	4.10×10^{-3}
0.50	4.69×10^{-2}	5.21×10^{-2}	1.40×10^{-2}	1.40×10^{-2}	2.13×10^{-2}	7.15×10^{-2}	1.40×10^{-2}
0.75	1.10×10^{-1}	1.16×10^{-1}	3.04×10^{-2}	3.04×10^{-2}	4.69×10^{-2}	1.60×10^{-1}	3.04×10^{-2}
1.00	1.85×10^{-1}	2.06×10^{-1}	5.35×10^{-2}	5.35×10^{-2}	8.28×10^{-2}	2.83×10^{-1}	5.35×10^{-2}

Table 3.14: Reconstruction error for function Poly decomposed using DWT. D (R) and D (E) stand for Donoho universal threshold with, respectively, real noise variance and estimated noise variance, Opt. stands for optimum threshold and σ is the noise variance.

	MSE						
σ	KLID	JID	D (R)	D (E)	SURE	Minimax	Opt.
	db2						
0.10	7.86×10^{-4}	7.72×10^{-4}	6.82×10^{-4}	6.47×10^{-4}	2.88×10^{-3}	2.36×10^{-3}	6.39×10^{-4}
0.25	3.77×10^{-3}	4.23×10^{-3}	3.05×10^{-3}	3.06×10^{-3}	1.74×10^{-2}	1.37×10^{-2}	3.01×10^{-3}
0.50	1.72×10^{-2}	2.11×10^{-2}	1.17×10^{-2}	1.17×10^{-2}	6.79×10^{-2}	5.29×10^{-2}	1.16×10^{-2}
0.75	3.03×10^{-2}	4.06×10^{-2}	2.53×10^{-2}	2.53×10^{-2}	1.52×10^{-1}	1.19×10^{-1}	2.53×10^{-2}
1.00	7.15×10^{-2}	7.18×10^{-2}	4.45×10^{-2}	4.45×10^{-2}	2.71×10^{-1}	2.13×10^{-1}	4.45×10^{-2}
	db4						
0.10	7.71×10^{-4}	8.77×10^{-4}	7.27×10^{-4}	7.27×10^{-4}	1.19×10^{-3}	1.89×10^{-3}	7.08×10^{-4}
0.25	8.95×10^{-3}	8.92×10^{-3}	3.64×10^{-3}	3.64×10^{-3}	5.91×10^{-3}	1.15×10^{-2}	3.33×10^{-3}
0.50	3.31×10^{-2}	2.01×10^{-2}	1.25×10^{-2}	1.25×10^{-2}	2.28×10^{-2}	4.40×10^{-2}	1.25×10^{-2}
0.75	5.43×10^{-2}	4.55×10^{-2}	2.74×10^{-2}	2.74×10^{-2}	5.03×10^{-2}	9.84×10^{-2}	2.74×10^{-2}
1.00	1.47×10^{-1}	8.63×10^{-2}	4.81×10^{-2}	4.81×10^{-2}	8.86×10^{-2}	1.74×10^{-1}	4.81×10^{-2}
	sym4						
0.10	7.29×10^{-4}	7.36×10^{-4}	7.08×10^{-4}	7.05×10^{-4}	1.86×10^{-3}	1.74×10^{-3}	6.85×10^{-4}
0.25	5.86×10^{-3}	5.09×10^{-3}	3.37×10^{-3}	3.44×10^{-3}	1.08×10^{-2}	9.60×10^{-3}	3.34×10^{-3}
0.50	1.87×10^{-2}	2.23×10^{-2}	1.25×10^{-2}	1.25×10^{-2}	4.17×10^{-2}	3.68×10^{-2}	1.25×10^{-2}
0.75	4.13×10^{-2}	4.44×10^{-2}	2.74×10^{-2}	2.74×10^{-2}	9.34×10^{-2}	8.33×10^{-2}	2.74×10^{-2}
1.00	7.30×10^{-2}	6.00×10^{-2}	4.81×10^{-2}	4.81×10^{-2}	1.65×10^{-1}	1.48×10^{-1}	4.81×10^{-2}
	db12						
0.10	9.98×10^{-4}	1.01×10^{-3}	8.12×10^{-4}	8.27×10^{-4}	5.60×10^{-3}	2.56×10^{-3}	8.12×10^{-4}
0.25	4.68×10^{-3}	5.44×10^{-3}	3.85×10^{-3}	3.93×10^{-3}	4.08×10^{-3}	1.42×10^{-2}	3.85×10^{-3}
0.50	2.27×10^{-2}	2.00×10^{-2}	1.33×10^{-2}	1.33×10^{-2}	1.92×10^{-2}	5.54×10^{-2}	1.33×10^{-2}
0.75	5.17×10^{-2}	4.53×10^{-2}	2.89×10^{-2}	2.91×10^{-2}	4.32×10^{-2}	1.27×10^{-1}	2.89×10^{-2}
1.00	8.94×10^{-2}	7.99×10^{-2}	5.07×10^{-2}	5.11×10^{-2}	7.62×10^{-2}	2.26×10^{-1}	5.07×10^{-2}
	sym12						
0.10	7.65×10^{-4}	7.68×10^{-4}	7.67×10^{-4}	7.81×10^{-4}	8.74×10^{-4}	1.88×10^{-3}	7.49×10^{-4}
0.25	4.71×10^{-3}	4.71×10^{-3}	3.88×10^{-3}	3.85×10^{-3}	4.38×10^{-3}	1.12×10^{-2}	3.76×10^{-3}
0.50	1.74×10^{-2}	1.62×10^{-2}	1.34×10^{-2}	1.33×10^{-2}	1.65×10^{-2}	4.38×10^{-2}	1.33×10^{-2}
0.75	3.55×10^{-2}	3.62×10^{-2}	2.91×10^{-2}	2.89×10^{-2}	3.62×10^{-2}	9.79×10^{-2}	2.89×10^{-2}
1.00	6.26×10^{-2}	6.75×10^{-2}	5.12×10^{-2}	5.07×10^{-2}	6.38×10^{-2}	1.73×10^{-1}	5.07×10^{-2}

Table 3.15: Reconstruction error for function Poly decomposed using SWT. D (R) and D (E) stand for Donoho universal threshold with, respectively, real noise variance and estimated noise variance, Opt. stands for optimum threshold and σ is the noise variance.

3.4 An application to Fourier shape descriptors

In the following section, ENID, KLID and JID have been employed to determine the optimal number of shape descriptors. For the sake of completeness, a brief introduction to shape descriptors will be given before providing numerical results.

3.4.1 Shape descriptors

Shape, along with texture and color, is a key feature to describe any visual object, and the increasing amount of visual data available nowadays demands

fast and efficient storage and analysis methods. Contour-based methods allow for a compact and effective shape description that is crucial in many tasks.

A shape descriptor should define an accurate and compact shape representation, allowing for straightforward comparisons and, in addition, should not be computationally demanding.

Shape descriptors can be classified in two categories: volume-based and pixel-based. Volume-based shape descriptors, such as geometric moments [61] and Zernike moments [128] involve all the pixel of the object. Conversely, pixel-based descriptors only take into account shape boundary. Curvature scale-space [129, 130], active shape models [131], hierarchical skeletons [132] and Fourier descriptors [133, 134] belong to this class.

A further classification can be made between local and global descriptors. Local descriptors focus on the shape's local feature, and they are used in feature detection, registration, segmentation and labelling. On the contrary, global descriptors describe the entire contour, and are employed in object retrieval, classification, recognition and clustering. Recently, descriptors combining both global and local properties have been proposed [135].

In this application we focus on Fourier shape descriptors. Given any shape (i.e. a closed curve in the euclidean plane), a Fourier shape descriptor is the Fourier transform of a given representation of its boundary, which is usally parametrized by means of complex coordinates, curvature centroid distance or cumulative angular function [136, 61, 137, 138]. Among the many advantageous properties of Fourier descriptors, they are rotation invariant and it is easy to make them translation invariant and scale invariant.

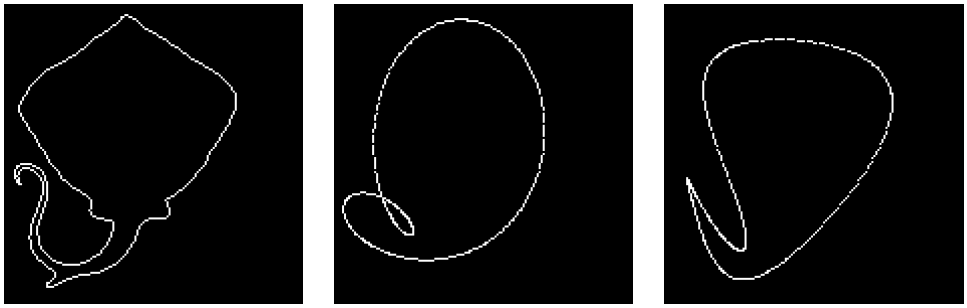
In order to achieve compactness, usually the first M coefficient of the Fourier transform of a shape are chosen as descriptors. Following the approach already used in wavelet bases, we consider the first M rearranged coefficients instead, i.e. the first M coefficients of greater absolute value, hence defining a hierarchical Fourier descriptor. As a result, the underlying shape is represented by its most representative frequency content, rather than its low-frequency content. As it can be observed in figs. 3.33, 3.34 and 3.35, if M is small, hierarchical Fourier descriptors are capable of better encoding relevant details of a shape. In particular, it can be observed in fig. 3.33 that six hierarchical Fourier descriptors are enough to avoid self-intersection of the shape, while six linear Fourier descriptors are not.

Even in the case of hierarchical Fourier descriptors, the number M of Fourier coefficients is a sensitive parameter of the model. In order to choose M automatically, ENID, KLID and JID are employed to select the optimal set of representative coefficients.

3.4.2 Algorithm and numerical results

The main numerical results of this section regard the task of shape compression, and early results about shape recognition tasks are included as well.

Complex Fourier descriptor have been chosen. Consider a shape parametrized as a curve $P(t) = (x(t), y(t)) : [0, T] \to \mathbb{R}^2$, then a complex curve $z(t) : [0, T] \to \mathbb{C}$ is constructed by defining x and y to be, respectively, its real and imaginary part:

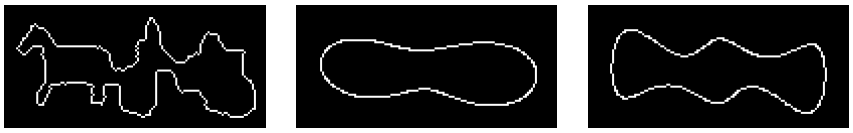


(a) (b) (c)

Figure 3.33: Ray shape: original (a), approximated with 6 linear Fourier descriptors (b), approximated with 6 rearranged Fourier descriptors (c).

(a) (b) (c)

Figure 3.34: Bone shape: original (a), approximated with 6 linear Fourier descriptors (b), approximated with 6 rearranged Fourier descriptors (c).



(a) (b) (c)

Figure 3.35: Carriage shape: original (a), approximated with 6 linear Fourier descriptors (b), approximated with 6 rearranged Fourier descriptors (c).

$$z(t) = x(t) + iy(t).$$

The Fourier descriptors are then obtained as the Fourier transform $\hat{z}(\xi)$ of $z(t)$. Remark that, since $z(t) \in \mathbb{C}$, no symmetry is present in $\hat{z}(\xi)$. As stated before, the optimal number of coefficients $\tilde{M}$ is determined according to the chosen measure (ENID, KLID, JID) and the $\tilde{M}$ largest magnitude coefficients are retained, the remaining ones are set to 0. Even if the length of the parametrized curve does not change, the induced sparsity of the transform allows for a more efficient lossless compression, for example using entropic coding.

The shape-compression algorithm is structured as follows.

Algorithm 4 Hierarchical Fourier descriptors

Input: Discrete parametrized complex shape $z_k = x_k + iy_k$, $k = 1, \ldots, n$
Output: Compressed discrete parametrized complex shape $\tilde{z}_k = \tilde{x}_k + i\tilde{y}_k$, $k = 1, \ldots, n$

1. Compute the discrete Fourier transform $\hat{z}(\xi)$ of z_k
2. Define $a(t) = |z|^*(t)$ as the decreasing monotonic rearrangement of the absolute value of $z(t)$
3. Normalize $a(t)$ so that $a(t) : [0, 1] \to [0, 1]$
4. Determine $\tilde{M}$ according to the chosen measure (ENID, KLID or JID) applied to $a(t)$
5. Preserve the first $\tilde{M}$ rearranged coefficients and set to zero the remaining ones
6. Invert the discrete Fourier transform

Algorithm 4 has been tested on a database of 1069 bitmap images, each containing a different white shape on black background, whose contour has been extracted using the built-in Matlab function **bwtraceboundary**.

For each shape, mean square error (MSE) has been measured between original and reconstructed image; compression ratio has been measured as $\tilde{M}/n$, i.e. the ratio between the number of selected coefficients and length of the discretized curve.

The results are shown in table 3.16. To better evaluate the global performance of ENID, KLID and JID, both MSEs and compression ratios have been aggregated by taking the average value and the median.

It can be noted that ENID tends to select, slightly more than half of the coefficients on average, but the average MSE is small and the median is zero. Indeed, MSE is not zero fo only eight shapes in the entire database. When MSE is not zero, the compressed shape differs form the original one by only two pixels.

Selecting a very small number of coefficients, KLID tends to realize a higher MSE. In this case the difference between the average and the median of the compression ratio indicates that KLID produces less stable results than ENID.

	Average		Median	
	MSE	ratio	MSE	ratio
ENID	1.3487×10^{-6}	56.61%	0	56.60%
KLID	0.0289	0.99%	0.0273	3.37%
JID	0.0263	4.62%	0.0235	4.63%

Table 3.16: Performances of proposed measures on the database.

On the other hand, JID produces lower MSE than KLID, while still achieving very low compression ratio, (around 4% for both average and median aggregation) and it is more consistent across the database than KLID.

Examples of shapes compressed using the different measures are shown in figs. 3.36 and 3.37 – since the compressed version provided by ENID is identical to the original shape, the latter is omitted. Conversely, shapes compressed according to KLID and JID are less detailed. However, they are still distinguishable from one another and a human observer is still capable of recognizing the subject represented by the shape.

As a last test, a preliminary investigation of the discriminative properties of hierarchical Fourier descriptors has been conducted. Euclidean distance between hierarchical Fourier descriptor vectors, selected according to ENID, has been considered as a metric to evaluate similarity between shapes. In fig. 3.38 the first fifteen matches of the shape of a hand are shown, ordered from left to right and from top to bottom – the first is the query shape. It is worth observing that hierarchical Fourier descriptors are capable of recognizing similarities between shapes, even if they are not identical: in many of the images in fig. 3.38 the fingers of the hand are in different position than in the original one.

A more challenging case is if one considers the shape of a ray, due to the complexity of the shape and the high amount of fine details. However, as it can be seen in fig. 3.39, in the first fourteen matched shapes, ten are in the same class as the original one, while only four of them belong to other classes (the hand, the bottles and the plane).

In addition, some preliminary tests in clustering tasks using k-means algorithm have been conducted and early results are encouraging, motivating further studies concerning this topic.

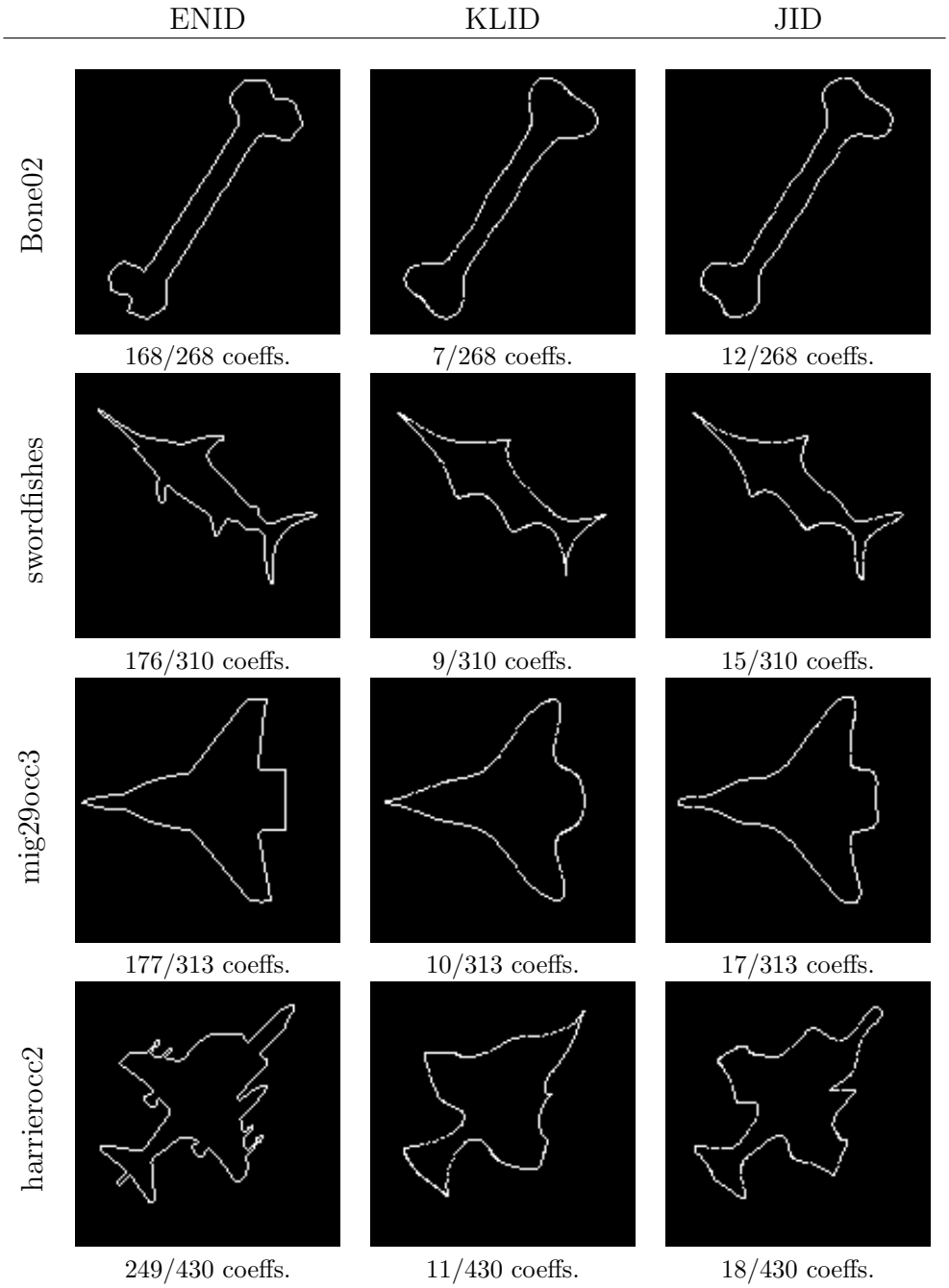


Figure 3.36: Compressed versions of shapes Bone02, swordfishes, mig29occ3 and harrierocc2. Below each shape the number of selected coefficients is shown versus the total number of coefficients.

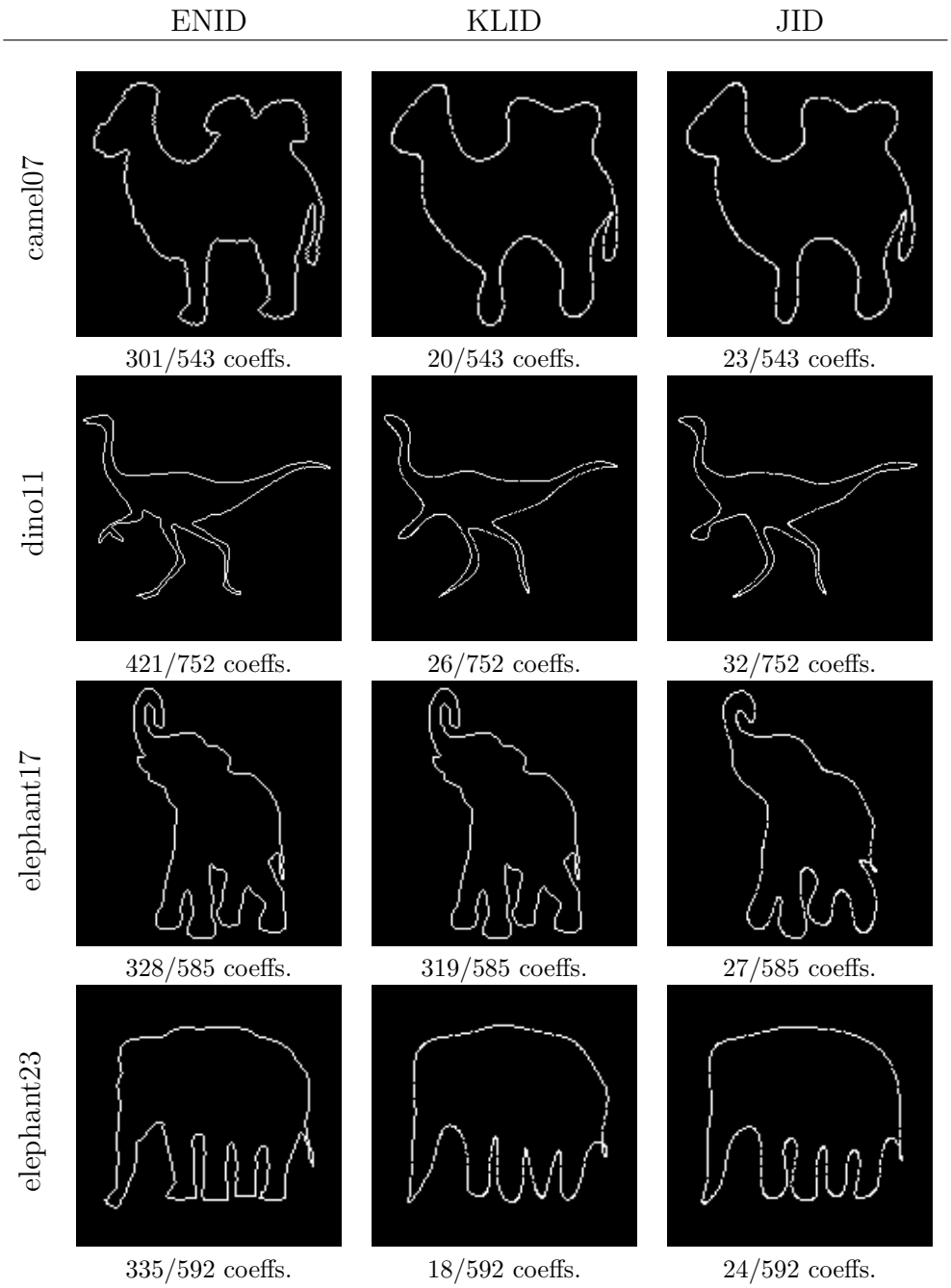


Figure 3.37: Compressed versions of shapes camel07, dino11, elephant17 and elephant23. Below each shape the number of selected coefficients is shown versus the total number of coefficients.

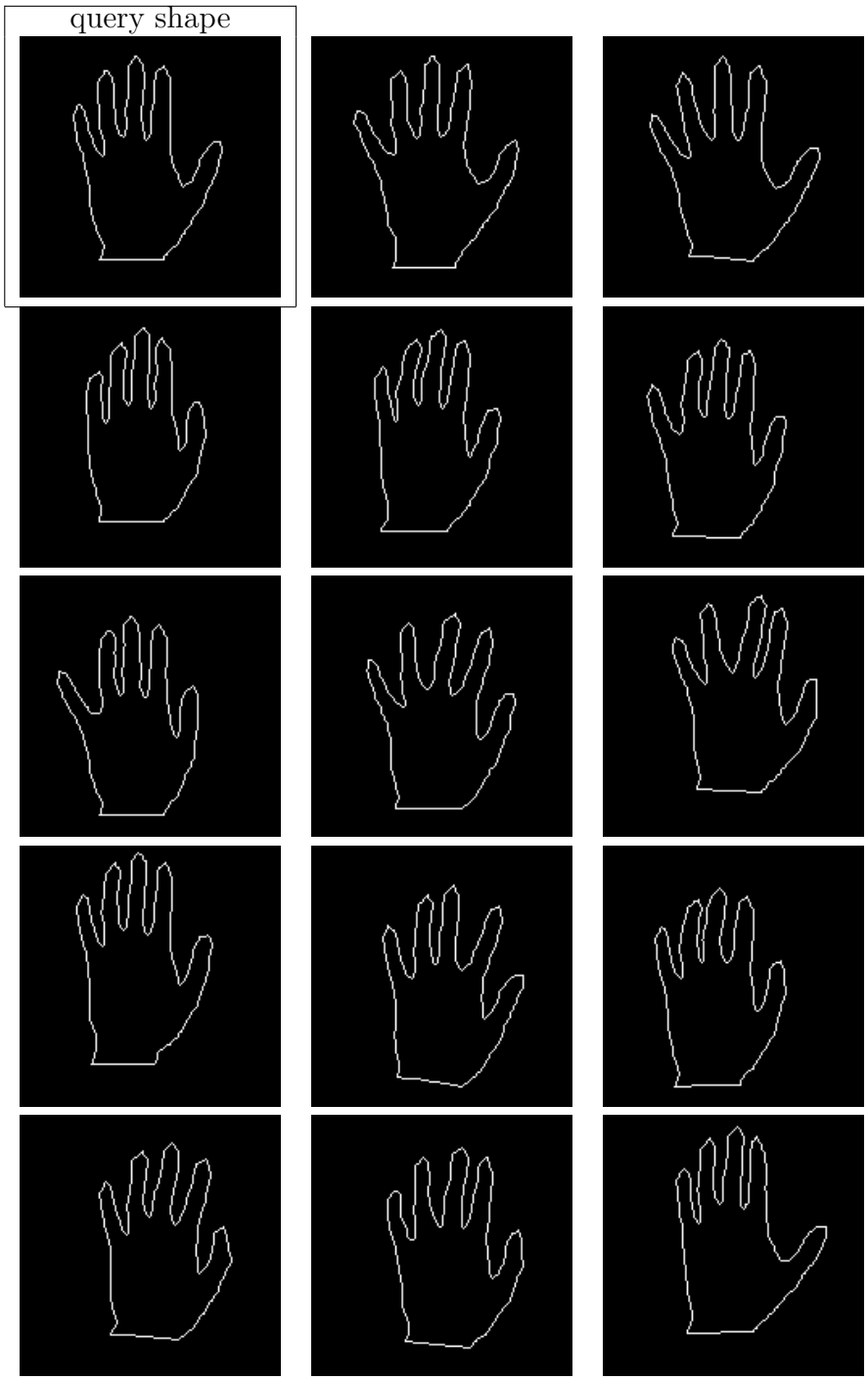


Figure 3.38: First fourteen shapes in the database matching the query shape **cthand06**, ordered from left to right and from top to bottom.

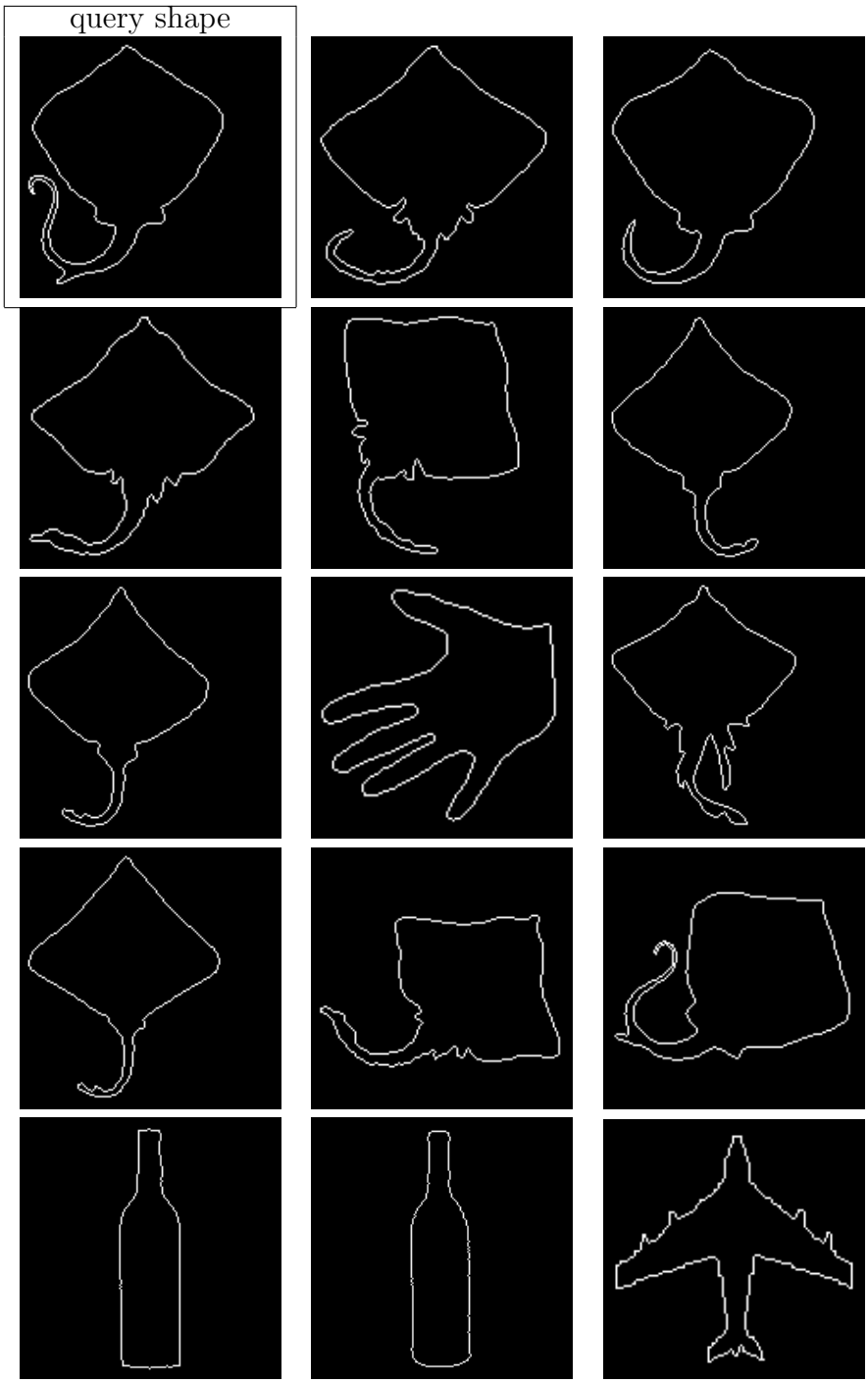


Figure 3.39: First fourteen shapes in the database matching the query shape **ray20**, ordered from left to right and from top to bottom.

Chapter 4

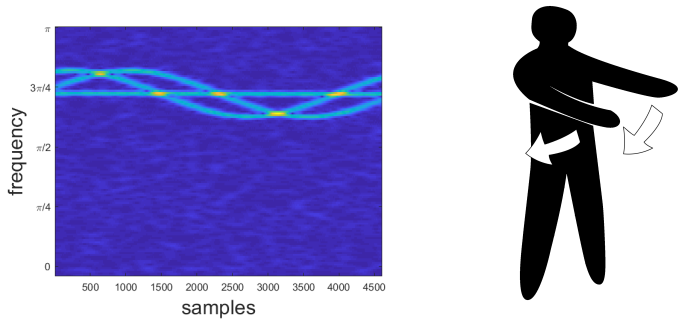
Optimal filtering via theoretic information criteria

Contents

In this chapter the issue of locating interference regions between individual modes in multicomponent non-stationary signals is addressed. A brief introduction about the problem will be given, then the proposed method will be described and finally experimental results will be presented. A crucial step in the proposed method is the filtering of an energy signal, where the exact amount of filtering is not given in advance but depends on the signal. In order to select the optimum level, the differential-entropy based information measures developed in Chapter 3 for coefficients selection have been used.

Multicomponent non-stationary signals are present in a variety of real world applications, ranging from radio communications to gravitational waves, including surveillance, human gait recognition, radar and sonar analysis and medical analysis. A multicomponent signal can be viewed as composed by individual modes, whose characterization depends on the particular model. Separation of the individual modes is a central issue in many of the aforementioned applications. As an example, consider the task of micro-doppler human gait recognition and classification. In regular doppler analysis (such as radar systems) an impulse is broadcast and the response signal, produced when it encounters any object, is analysed to infer information concerning relative position and velocity. Micro-doppler analysis aims at analysing the fine time-frequency variations of the response signal in order to gain knowledge about the moving parts of the object producing the reflections. In the case of human body, the different moving parts generate different individual modes in the response signal (see fig. 4.1) so that separating and extracting these modes allows for gait classification. Micro doppler sensors provide several advantages over optical sensor: they are

cheaper, provide automatic information about distance and velocity and are capable of operating in conditions of low visibility. Among the many applications, micro-doppler analysis is used in pedestrian recognition for self-driving cars [65].



(a) (b)

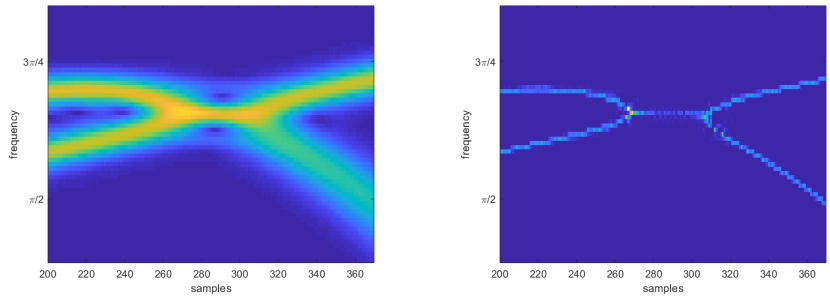
Figure 4.1: Simulated micro-doppler data: spectrogram (a) of the swinging arms movement (b). The non stationary modes in (a) are generated by the movement of the arms.

The two main approaches in the literature to analyse multicomponent signals rely either on time-frequency (TF) representations [62, 63] or on empirical mode decomposition (EMD) [139]. In TF representation approach, individual modes appear as ridges in the time-frequency domain but, due to energy spread and to the ambiguity caused by indetermination principle, post processing techniques are often required to improve the readability before separation and extraction operations. One of the main enhancement techniques is reassignment [140, 141], consisting in a reallocation of TF points toward the centroids of the TF distribution. Reassignment techniques have the advantage of simplicity and give good results if the individual modes are well separated in the TF domain but, if two modes intersect in the TF domain – i.e. they interfere with each other, reassignment performs poorly, as shown in fig. 4.2.

Empirical mode decomposition, conversely to TF representations, directly extracts the individual modes through a time-domain process named sifting: signal trend is iteratively computed by estimating envelopes of the input signal and then subtracted, until the residue satisfies some stopping criterion. Although the process is simple and operates completely in time domain, EMD presents some disadvantages: the extracted modes are themselves non-stationary – requiring Hilbert spectral analysis for instantaneous frequency estimation – and a proper theoretical framework is lacking.

Providing a more rigorous alternative to EMD, synchrosqueezing transform [142, 141, 64] combines time-frequency reassignment with complex wavelet transform. It allows for modes extraction through intrinsic modes decomposition (IMD), a procedure where well-separated –i.e. non interfering– modes are approximately reconstructed. The approximation has proven to be suitable for the implementation of IMD in real world applications.

Recently, direct reconstruction of individual modes in TF interference regions has been proposed [143], but the precision of the proposed method strongly depends on interference regions localization.



(a) (b)

Figure 4.2: Spectrogram (a) and reassigned spectrogram (b) of two interfering modes in a multicomponent signal. As it can be observed, wherever two instantaneous frequency laws are not separable, reassignment method is not able to distinguish them.

Although TF representations and reassignment methods are commonly used when addressing human gait recognition and classification issues [65, 66], the precision of such tools strongly depends on the selection of a proper analysis window and on the robustness of reassignment to modes interference. with regard to the analysis window choice, experiment-based heuristics are commonly used [65]; With regard to the robustness to interference, in the case of several gestures in the micro-doppler signal, it is crucial to avoid the reconstruction of segments of the different gestures as a single mode. The knowledge of the locations where single modes overlap (interference regions) allows to detect portions of the TF representation that might be misreassigned.

Interference detection can be carried out using counting methods. Following the well-known counting property of Rényi entropy [144], a method based on the estimation of short-time Rényi entropy is proposed in order to estimate the number of modes in a multicomponent signal [145, 146, 147, 148]. If the modes are locally similar and have locally similar amplitudes, short-time Rényi entropy is capable of correctly estimating the number of components in a multicomponent signal: interferences are then drops in the counting signal [149, 150, 151]. Although Rényi entropy based methods achieve overall good performances, a proper choice of the analysis window remains crucial. Furthermore, counting methods do not distinguish between constructive interference, i.e. when the overlapping modes are in phase with each other and the total energy increases, and destructive interference, i.e. when out-of-phase modes tend to cancel each other and energy decreases.

The method proposed in this chapter aims at the detection of interference temporal intervals (i.e. the temporal regions where a TF representation is not able to distinguish individual modes) by exploiting properties of local signal energy. It will be shown that interference points generate local extremal points in a smoothed energy signal. Each of them characterizes the main lobe of a fast decaying shape, whose first sidelobes correspond to the boundary of the corresponding interference time interval. The latter corresponds to the interference region measured on a TF representation whose analysis window has the same

support of the smoothing kernel. A study concerning the selection of the best kernel support has been conducted, and different optimality criteria are investigated. Interference are then located by selecting significant extrema of the smoothed energy signal. Constructive and destructive interference correspond, respectively, to local maxima and local minima of the smoothed energy signal (see fig. 4.3).

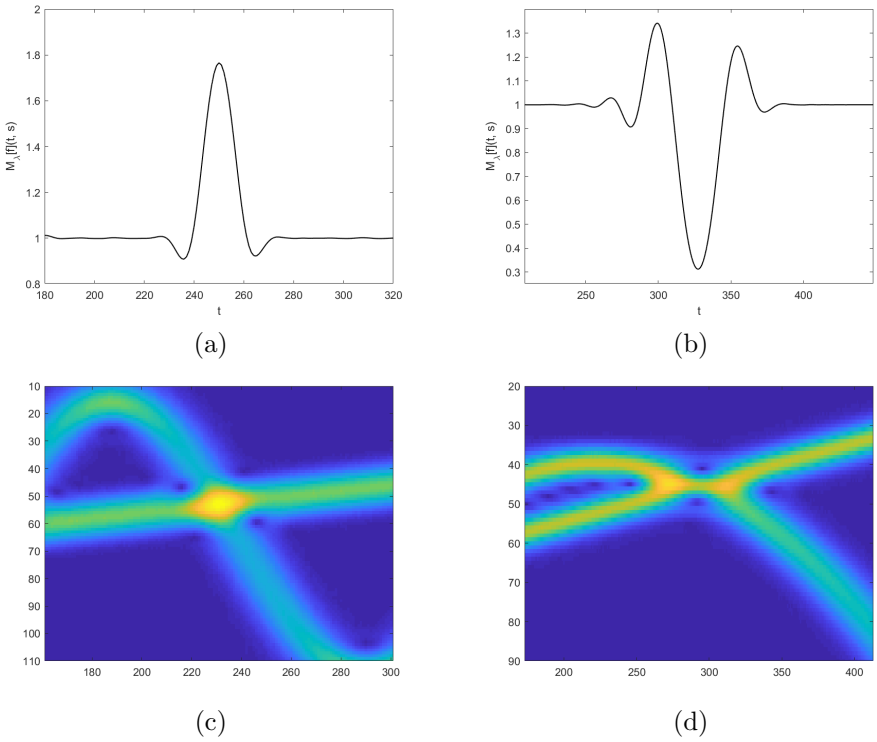


Figure 4.3: Example of proposed method performance and relative spectrograms on constructive (a & c) and on destructive (b & d) interference.

4.1 Interference location

The method presented in this section aims at defining the resolution scale that results optimal for interference analysis. A smoothed energy signal is constructed at different scales and a relative cost function is evaluated in order to identify the optimal scale, i.e. the scale at which signal natural oscillations are sufficiently dampened, while emphasizing modes interferences.

4.1.1 Mathematical formulation

Let $f(t) : \Omega \subset \mathbb{R} \to \mathbb{R}$ be a multicomponent signal, i.e. such that

$$f(t) = \sum_{i=1}^{n} A_k \sin(\phi_k(t)), \tag{4.1}$$

where the A_k's are positive constant amplitudes, the $\phi_k(t)$'s are phase functions (monotonically increasing functions) and Ω is a limited interval of the real line.

Definition 4.1.1. *Let $f(t)$ be defined as in eq. (4.1), $\lambda(t)$ be a positive compactly supported function such that $\int_{\mathbb{R}} \lambda(t)dt = 1$ and define $\lambda_s(t) = s^{-1}\lambda(t/s)$. The **smoothed energy signal** associated to f is defined as*

$$M_\lambda[f](u,s) = \int_\Omega \lambda_s(u-t)|f(t)|^2 dt. \tag{4.2}$$

The kernel $\lambda(t)$ is the impulse response of a low-pass filter, hence $M_f(u,s)$ is expected to be constant if no interferences occur, and to show severe variations otherwise. In [143] it has been shown that the energy of the spectrogram of a multicomponent signal computed with respect to the frequency axis is itself a multicomponent signal, whose frequencies depend on the sum and the differences of the original instantaneous frequencies. Due to Plancharel theorem, the smoothed energy signal itself is a multicomponent signal with specific instantaneous frequency laws and time dependent amplitudes.

As an example, consider the simple case of the signal $f(t) = f_1(t) + f_2(t)$, where $f_1 = \sin(\phi_1(t))$ and $f_2(t) = \sin(\phi_2(t))$, with ϕ_1 and ϕ_2 phase functions. The smoothed energy signal $M_f(u,s)$ can be decomposed as

$$M_\lambda[f](u,s) = (\lambda_s * |f_1|^2)(u) + (\lambda_s * |f_2|^2)(u) + 2(\lambda_s * f_1 f_2)(u). \tag{4.3}$$

Since f_1 and f_2 have constant amplitudes, we expect the first two members of eq. (4.3) to be constant (this claim will be proved later), while the third term can be rewritten as

$$2(\lambda_s * f_1 f_2)(u) = (\lambda_s * \tilde{f}_1) + (\lambda_s * \tilde{f}_1),$$

where

$$\tilde{f}_1(t) = -\frac{1}{2}\cos(\phi_1(t) + \phi_2(t))$$

and

$$\tilde{f}_2(t) = \frac{1}{2}\cos(\phi_1(t) - \phi_2(t)).$$

If no interferences occur, both $\tilde{f}_1$ and $\tilde{f}_2$ oscillate fast enough and the kernel λ filters them out. Conversely, if for some t_0 it happens that $\phi_1'(t_0) = \phi_2'(t_0)$, then $\tilde{f}_1$ is still filtered out around t_0 while $\tilde{f}_2$ is not, since its instantaneous frequency is almost zero.

To formalize the previous example, some mathematical tools are needed.

Definition 4.1.2. *Let P be a periodic function of period T. Define the mean of P as*

$$M[P] = \frac{1}{T}\int_0^T P(\theta)d\theta.$$

Definition 4.1.3. *Let $f : \mathbb{R} \to \mathbb{R}$ be a function and let $\lambda : \Gamma \subset \mathbb{R} \to \mathbb{R}$ be a function such that $\int_\Gamma \lambda(t)dt = 1$.*

*For any $t \in \mathbb{R}$, define the **smoothed mean** of g according to λ as*

$$M_\lambda[g](t) = (\lambda * g)(t).$$

Remark 4.1.4. *For any fixed s, the smoothed energy signal of a function f is a smoothed mean with $g = |f|^2$ and $\lambda = \lambda_s$*

The following lemma formalizes the aforementioned notion, concerning the fact that if a chirp is not in interference (as it is a single mode), then it has constant energy or, more generally, constant mean.

Lemma 4.1.5. *Let $P : \mathbb{R} \to \mathbb{R}^+$ be a positive periodic function with period T, $\phi : \mathbb{R} \to \mathbb{R}^+$ be a strictly monotonically increasing function, $\lambda : [-1, 1] \to \mathbb{R}$ be a positive function such that $\int_{-1}^{1} \lambda(t)dt = 1$ and let $\lambda_s(t) = s^{-1}\lambda(t/s)$. Then, if $\phi'(t) \geq \Theta$,*

$$M_{\lambda_s}[P \circ \phi](u) \approx M[P],$$

in the sense that, for any $t \in \mathbb{R}$,

$$\left|\frac{M_{\lambda_s}[P \circ \phi](u) - M[P]}{M[P]}\right| \leq \frac{2T}{\Theta} + \left(|(\lambda_s)'| * \frac{1}{|\phi'|}\right)(u) + \left(|\lambda_s| * \frac{|\phi''|}{|\phi'|^2}\right)(u).$$

Qualitatively speaking, Lemma 4.1.5 shows that the mean value of a positive chirp that is "fast enough" (i.e. its instantaneous frequency is sufficiently high), is almost constant. Consider a sinusoidal chirp $f(t) = |\sin(\phi(t))|$: if the instantaneous frequency of the chirp is constant, then it is trivial that its mean is equal to the mean value of the absolute value of the sine function. Even if the instantaneous frequency is not constant (but it is sufficiently high), if we consider a single period of the sine function, we can consider it approximately constant. From another point of view, if the instantaneous frequency is high enough, the shape of the chirp on a single period does not differ much from the shape of a sine function, because the phase function has not "enough time" to change the shape of the sine, in that small time that is the considered period. The complete proof is in the appendix.

Remark 4.1.6. *If we assume that $\|\lambda'\|_1 = L$ and that, for any t, $|\phi''(t)| \leq K$, then*

$$\begin{aligned}\left|\frac{M_{\lambda_s}[P \circ \phi](u) - M[P]}{M[P]}\right| &\leq \frac{2T}{\Theta} + \frac{1}{\Theta}\|(\lambda_s)'\|_1 + \frac{K}{\Theta^2} \\ &\leq \frac{1}{\Theta}\left(2T + \frac{L}{s} + \frac{K}{\Theta}\right).\end{aligned}$$

Since $1/\Theta$ is small, the error is large only if s is close to zero. In practice, numerical experiments show that for s sufficiently far from zero the error is small.

As a consequence, it is possible to prove that chirps with high enough frequency have constant smoothed energy.

Corollary 4.1.7. *Let $f(t) = \sin(\phi(t))$, $\phi(t)$ a phase function, $\phi \gg 0$. Then*

$$M[f](u,s) \approx \frac{1}{2},$$

in the sense of Lemma 4.1.5.

Proof. Observe that

$$\begin{aligned} M[\sin^2] &= \frac{1}{\pi}\int_0^\pi \sin^2(\theta)d\theta \\ &= \frac{1}{2}. \end{aligned}$$

Then

$$\begin{aligned} M[f](u,s) &= M[\sin^2 \circ \phi](u,s) \\ &\approx M[\sin^2] \\ &= \frac{1}{2} \end{aligned}$$

□

More generally, consider a superposition $f(t)$ of n modes $f_i(t) = A_i \sin(\phi_i(t))$, i.e.

$$f(t) = \sum_{i=1}^{n} A_i \sin(\phi_i(t)).$$

Then, a simple calculation yields

$$M_\lambda[f](u,s) = \sum_{i=1}^{n} M_\lambda[f_i](u,s) + \sum_{i=1}^{n}\left[(\lambda_s * \tilde{f}_{ij}^{+})(u) + (\lambda_s * \tilde{f}_{ij}^{-})(u)\right],$$

where

$$\tilde{f}_{ij}^{\pm}(u) = -A_i A_j \cos(\phi_i(u) \pm \phi_j(u)).$$

Thanks to Lemma A.1.1, the first term is constant and it results equal to $2^{-1}\sum_{i=1}^{n} A_i$.

With regard to the second term, the linearity of the convolution allows us to consider each addendum in the sum separately. Since for each $i = 1, \dots, n$ $\phi_i'(t) \geq 0$, then the instantaneous frequency $\phi_i' + \phi_j'$ of $\tilde{f}_{ij}^{+}$ is always much greater than zero and, then, the term $\tilde{f}_{ij}^{+}$ is suppressed by the low-pass filtering action of λ_s. As it can be observed in fig. 4.4, for each term $\tilde{f}_{ij}^{-}$ two cases arises: either no interference occurs and then, for any $t \in [u-s, u+s]$,

$$\phi_i'(t) \neq \phi_j'(t)$$

and, for some $\Delta\xi > 0$,

$$|\phi_i'(t) - \phi_j'(t)| > \Delta\xi$$

or, if there exists $t_0 \in [u-s, u+s]$ such that

$$\phi_i'(t_0) = \phi_j'(t_0),$$

then for some $\Delta\xi > 0$ and for any $t \in [u-s, u+s]$,

$$|\phi_i'(t) - \phi_j'(t)| < \Delta\xi.$$

In the first case, since its instantaneous frequency is separated from zero, a proper choice of s correctly suppresses the term $\tilde{f}_{ij}^{-}$. In the second case, since its instantaneous frequency is close to zero, the term is approximately constant and $(\lambda_s * \tilde{f}_{ij}^{-})(u)$ will not be filtered out, as λ_s has unity gain at zero frequency,. The presence of this last term (which can be positive as well as negative), in case of interference, determines a variation in the smoothed energy signal which identifies interferences.

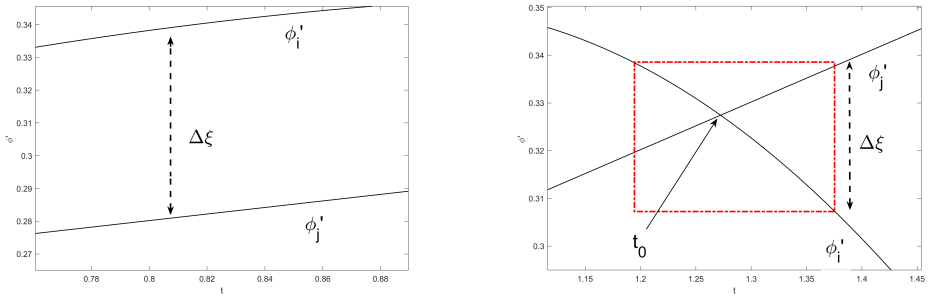
Since the instantaneous frequencies of the $f_{ij}^{\pm}$'s are not constant, questions might arise whether a linear filtering process is effective. The following lemma proves that, even if the instantaneous frequency of a sinusoidal chirp is not constant, a linear filtering still suppresses high frequencies.

Lemma 4.1.8. *Let $\phi(t) : \mathbb{R} \to \mathbb{R}$ be a phase function, i.e. a continuous monotonically increasing function and set $f(t) = \sin(\phi(t))$. Consider $\lambda(t) : [-1,1] \to \mathbb{R}^+$ such that $\int_{\mathbb{R}} \lambda(t)dt = 1$ and, for any $s > 0$, define $\lambda_s(t) = s^{-1}\lambda(t/s)$.*

Then, for any $u \in \mathbb{R}$,

$$|(\lambda_s * f)(u)| \leq \sup_{t\in[u-s,u+s]} |\hat{\lambda}(s\phi'(t))|.$$

The proof is in the appendix.



(a) (b)

Figure 4.4: Example of well separated (a) and interfering (b) frequency laws ϕ_i' and ϕ_j'.

The amount of smoothing applied to the signal depends on the support of λ_s and then, on s: the choice of s plays a crucial role in the proposed method. As shown in fig. 4.5(a), if the support of λ_s is too small, then an excessive amount of high frequency content is still present in the smoothed energy signal; hence interference regions are not enhanced enough to allow identification. On the other hand, as shown in fig. 4.5(b), if the support of λ_s is too large, then $M_\lambda[f](u,s)$ is excessively smoothed, and the distinction between meaningful extrema and ripples due to the filtering becomes troublesome.

The optimal scale $\tilde{s}$ is then identified as the one at which all the fast oscillations are sufficiently suppressed, but the low frequency information is not oversmoothed.

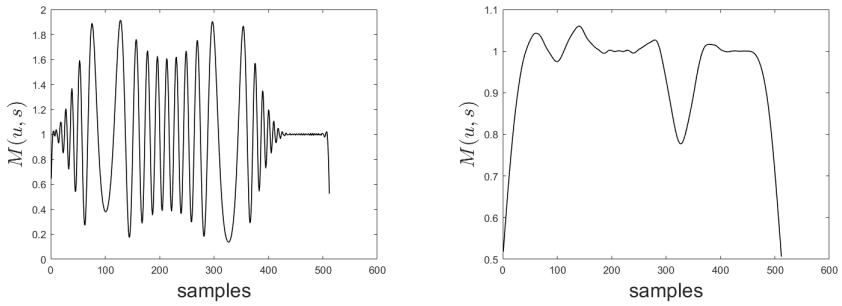


Figure 4.5: Smoothed energy signal (4.2) in case of too small kernel support (a) and too large kernel support (b)

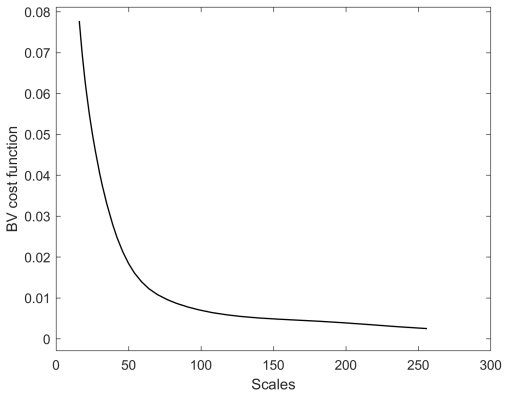


Figure 4.6: Behaviour of $C_{BV}(M(u,s))$ with respect to the scale.

4.1.2 Optimal scale estimation

In order to select the optimal scale $\tilde{s}$ automatically, a rate/distortion type curve is constructed using a BV cost function, which is defined as

$$C_{BV}[g] = \int_{\Omega} |g'(t)|dt, \tag{4.4}$$

where $g(t) : \Omega \to \mathbb{R}$. The BV cost function is a natural choice to measure how much a signal g oscillates.

In fig. 4.6 the typical profile of C_{BV} is shown with respect to the scales, and it can be noted that it strongly resembles a rate/distortion curve, where C_{BV} is the distortion while s is the rate.

Since as the scale increases the signal oscillates less, the optimal point $\tilde{s}$ is the point in the rate/distortion curve that gives the best compromise between oscillations and scale. As mentioned in the preceding chapters, several methods have been proposed to estimate the optimal point in a rate/distortion curve. Classical methods include L-curve methods [57], where usually the optimal point is identified as the one maximizing the second derivative of the curve, or methods based on the Occam's Razor principle, as Rissanen's MDL [152] or the tangent line approximation technique in [58].

Although the curve given by C_{BV} seems to be smooth, it presents little oscillations that make direct estimation of the maximum of the second derivative, required in the L-curve method, not feasible in practice. Since there is no underlying model for $C_{BV}(M_\lambda[f](\cdot, s))$, any kind of regularization might alter the underlying information. For the same reason, MDL principle is not straightforward applicable. Furthermore, MDL functionals require a parameter to balance the model-data likelihood and the cost of the model. The choice of the parameter heavily influences the performances and, in addition, for a truly automatic scale selection method, no parameters should be present.

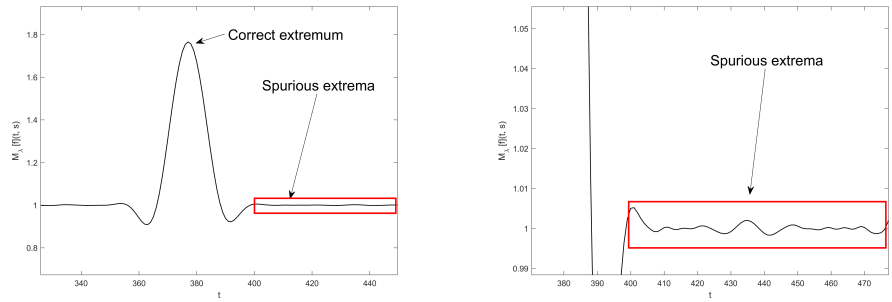
On the other hand, the tangent line approximation technique, proposed as part of Natarajan's Occam filter for denoising through compression, estimates two straight lines, one at the beginning and the other at the end of the curve, and individuates the optimal point as the intersection of these two lines, as shown in fig. 4.6. Since in practice the curve is not regular, tangent estimation gives very poor results. Instead, they are estimated as the best fitting straight lines (in the sense of least squares) in a small interval around the beginning or the end of the curve.

In addition, the information-based selection methods proposed in Chapter 3 for coefficients selection have been implemented to find the optimal point. The application of both KLID selection method and JID selection method is straightforward for the present problem: we stress the fact that the purpose of the proposed measures is to find the optimal point in a given decreasing curve *from an information standpoint*. The only assumption is that there is a separation in the curve between a "relevant part" and a "less relevant part", but this assumption can be usually made in a wide number of cases.

Although both methods have been detailed in Algorithm 2 and Algorithm 3 in Chapter 3, we briefly summarize them here for the sake of completeness. Assuming $s \in [s_1, s_2]$, for each scale $\bar{s}$, the BV cost curve $C_{BV}[M_\lambda[f](\cdot, \bar{s})]$ is divided into two normalized functions: $\tilde{C}_1^{\bar{s}}(s)$, for $s \in [s_1, \bar{s}]$ and $\tilde{C}_2^{\bar{s}}(s)$, for $s \in [\bar{s}, s_2]$. Then KLID (or JID) between $\tilde{C}_1^{\bar{s}}(s)$ and $\tilde{C}_1^{\bar{s}}(s)$ is computed and the $\tilde{s}$ realizing the minimum is picked as the optimal level.

4.1.3 Selection of energy extrema

Once the optimal scale is selected, the extraction of the meaningful extrema is performed, as they carry information concerning modes interference. Let $P = \{t_1, \dots, t_m\}$ the time positions of the local extrema of $M_\lambda[f](u, \tilde{s})$. High frequencies at the optimal level are suppressed but, since the filtering process is linear, spurious extrema remain, and need to be excluded from the candidates for interference locations (see fig. 4.7).



(a) (b)

Figure 4.7: Correct and spurious extrema (a) and detail of spurious extrema (b).

Since $M_\lambda[f](u, \tilde{s})$ has not zero mean, a direct thresholding based on its values is not feasible. Instead, in order to discard the local extremal points which are due to suppressed high frequencies, the following variation measure on the set P is defined:

$$DP(i) = \frac{1}{2}\Big[\Big|M_\lambda[f](t_{i+1}, \tilde{s}) - M_\lambda[f](t_i, \tilde{s})\Big| + \Big|M_\lambda[f](t_i, \tilde{s}) - M_\lambda[f](t_{i-1}, \tilde{s})\Big|\Big]. \tag{4.5}$$

The function DP measures the variation of the extremal point located in t_i with respect to its neighbours. Given a threshold $\tau > 0$, the set $\widetilde{P}$ of the time positions of the interferences is the set of those extrema whose variation amplitude exceeds τ:

$$\widetilde{P} = \{t_i \in P \mid DP(i) > \tau\}.$$

For each $t_i \in \widetilde{P}$, the corresponding interference region R_i is defined as the interval comprised between the closer leftmost extrema and the closer rightmost extrema, i.e.

$$R_i = [t_{i-1}, t_{i+1}].$$

Interference regions with common boundaries are merged together.

4.1.4 The algorithm

Regardless of the adopted method used to find the optimal level, the algorithm consists of the following steps.

Algorithm 5 Interference detection

Input: Signal $f(t)$, smoothing kernel λ, scales S, threshold τ.
Output: Interference time positions $\widetilde{P}$ and interference regions R_i's.

1. For each scale $s \in S$, compute $M_\lambda[f](u,s)$ as in eq (4.2);
2. for each scale $s \in S$, compute $C_{BV}(M_\lambda[f](\cdot,s))$ as in eq. (4.4);
3. using the chosen selection method applied to $C_{BV}(M_\lambda[f](\cdot,s))$, find the optimal scale $\tilde{s}$;
4. extract the extrema from $M_\lambda[f](u,\tilde{s})$ and collect their time positions in the set $P = \{t_1, \dots, t_m\}$;
5. for each $t_i \in P$, if $DP(i) > \tau$, assign t_i to the set $\widetilde{P}$, where DP is defined as in eq. (4.5);
6. for each $t_i \in \widetilde{P}$, assign the interference region as $R_i = [t_{i-1}, t_{i+1}]$;
7. for each R_i and R_j, $i \neq j$, if $t_{i+1} = t_{j-1}$ or $t_{i-1} = t_{j+1}$, merge R_i and R_j.

4.2 Performance analysis

Several signals have been considered to test the proposed method. In particular, numerical simulations have been performed on two synthetic signals consisting of two chirps with different time-frequency laws. These different signals represent the cases of interference between a linear and a quadratic chirp and of interference between a linear and a sine-modulated chirp. The test signals have been produced at a sampling frequency of $\xi_s = 256$hz for 512 samples, resulting in a length of 2 seconds. The scale set has been chosen to be dyadic, i.e. $S = \{\lfloor 2^{4+i/8} \rfloor\}_{i=0,\dots,32}$, the smoothing kernel λ is a 4-th order uniform B-spline and the threshold τ has been empirically set to $\tau = 1/5$.

The test signals are constructed as linear combination of the following chirps:

$$
\begin{aligned}
f_1(t) &= \sin\left(\pi\xi_s\left[\frac{1}{10}t^2 + \frac{2}{5}t + \frac{1}{5}\right]\right);\\
f_2(t) &= \sin\left(\pi\xi_s\left[\frac{1}{2}t + \frac{1}{20}\sin\left(\frac{5\pi}{2}t\right)\right]\right);\\
f_3(t) &= \sin\left(\pi\xi_s\left[-\frac{1}{5}t^3 + \frac{6}{10}t^2 + \frac{1}{10}t + \frac{1}{2}\right]\right).
\end{aligned}
$$

These signals have been designed to represent well real life applications, such as micro-doppler response signals [153, 154]. In addition, the performances on a simulation of micro-doppler data will be discussed further on.

With reference to the relative phase functions, we will refer to the signal $f(t) = f_1(t) + f_2(t)$ as the quadratic/sinusoidal signal, and to the signal $f(t) = f_1(t) + f_3(t)$ as the quadratic/cubic signal.

In table 4.1, the scales selected by the proposed methods are reported. With regard to the tangent line technique, the first ten and the last ten scales were considered. It is noteworthy that, for the synthetic data, KLID and JID always select the same point. In the case of the quadratic/sinusoidal signal, the selected optimal scales are very close. The corresponding smoothed energy signals and spectrogram are depicted in fig. 4.9. As it can be observed in figs. 4.9 (a) and (c), significant extrema are correctly identified by the proposed method, and hence interference points and interference regions are correctly estimated. To prove the correctness of the locations, in figs. 4.9 (b) and (c) detected interferences and interference regions are plotted on signal spectrogram: each interference point is correctly located where instantaneous frequency laws intersect and the corresponding interference region correctly encompasses a region around it where individual laws are not distinguishable. The spectrograms have been computed using a window with length equal to the selected scale.

In this case, the difference between the optimal scale selected using different methods is not relevant, but it can be noted that the smoothed energy signal $M_\lambda[f](u, s)$ presents less oscillations at the scale selected by KLID and JID than the ones selected by the tangent lines method. Although the larger the scale the more smoothed the signal, it is notable that the energy signal is still not oversmoothed at the scales selected by KLID and JID.

The difference between the optimal scale selected by different methods is more evident in the case of the quadratic/cubic signal, shown in fig. 4.11. The relative smoothed energy signals are depicted in figs. 4.11 (a) and (c), and it can be noted that the scale provided by the tangent lines method still presents many spurious oscillations, which are not present in the level selected by KLID and JID. Although the method in both cases correctly succeeds in locating interferences and interference regions, the level chosen using KLID and JID results more robust to changes in the threshold value during the extrema extraction procedure. In figs. 4.11 (b) nd (d), the relative spectrograms are shown: as in the previous case, for both selected levels the method correctly identifies interference points and interference regions. Spectrograms have been computed using a window length equal to the selected scale.

By comparing smoothed energy signals in figs. 4.9 (c) and 4.11 (c) with the relative spectrograms, it can be observed that constructive interferences correspond to maxima in $M_\lambda[f](u, s)$, while destructive interferences correspond to minima.

The proposed approach is further validated through a direct comparison with the reassigned spectrograms: in figs. 4.12 (a) and (b) interference regions are marked on the reassigned spectrogram and it can be observed that they exactly correspond to regions where reassignment fails. Even in this case, spectrograms have been computed using a window of size equal to the selected optimal scale.

	Tangent lines	KLID	JID
Quadr./sine	35	39	39
Quadr./cubic	46	70	70
Micro-doppler	166	305	559

Table 4.1: For each test signals, selected optimal scales according to the proposed methods: tangent lines technique, KLID and JID.

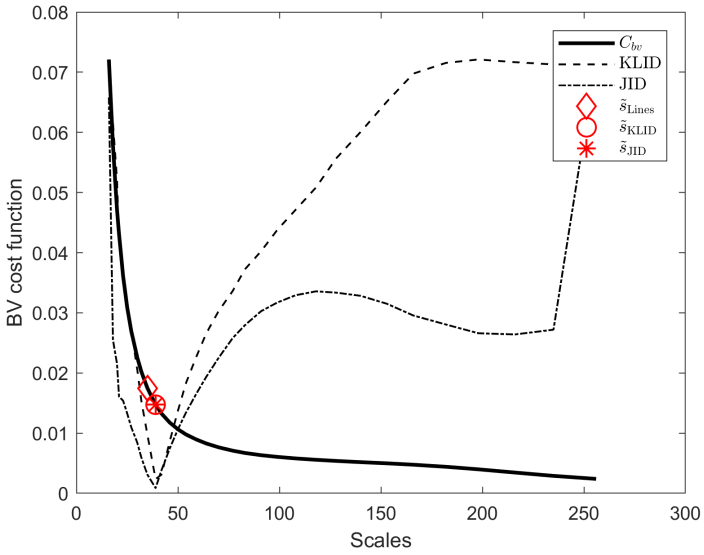


Figure 4.8: Signal $f(t) = f_1(t) + f_2(t)$ – quadratic and sine-modulated chirps. BV cost function versus scales. Optimal points estimated with different methods are shown, along with KLID and JID. KLID and JID have been normalized for visualization purposes.

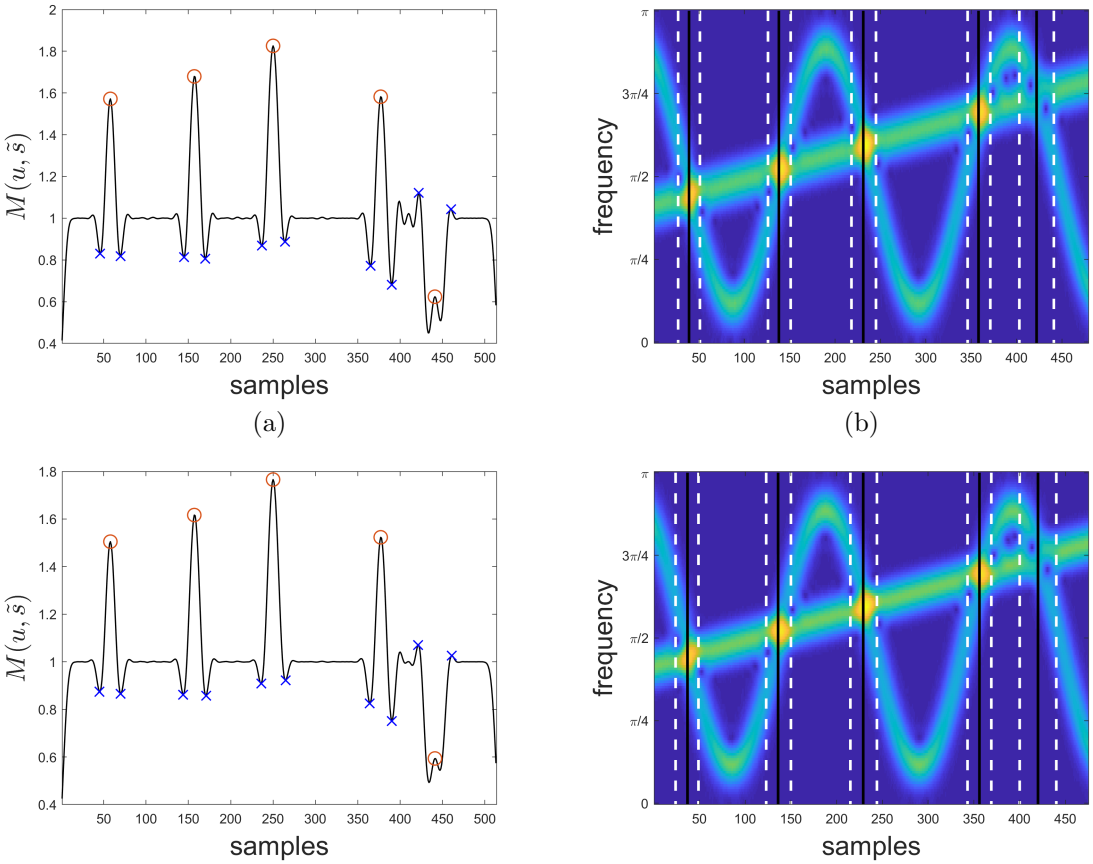


Figure 4.9: Signal $f(t) = f_1(t) + f_2(t)$ – quadratic and sine-modulated chirps. Smoothed energy signal and spectrogram at optimal level as selected by tangent lines technique (a, b) and both KLID and JID selection methods (c, d). Circles and black lines represent interferences while crosses and dashed lines represent interference regions.

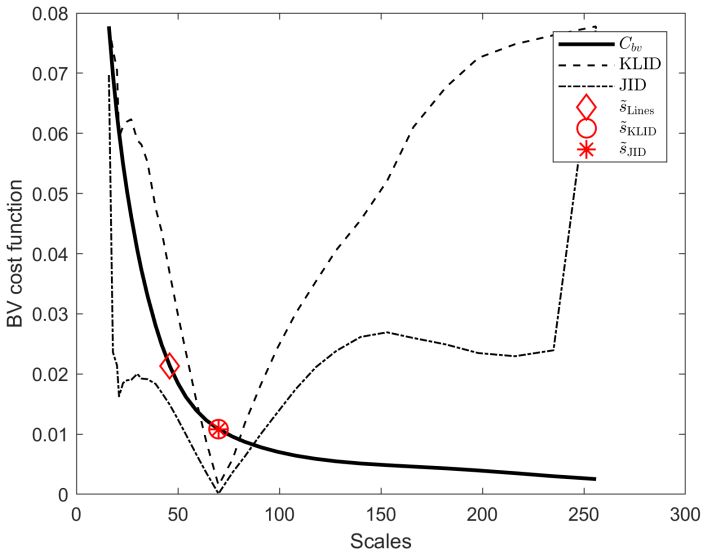


Figure 4.10: Signal $f(t) = f_1(t) + f_3(t)$ – quadratic and cubic chirps. BV cost function versus scales. Optimal points estimated with different methods are shown, along with KLID and JID. KLID and JID have been normalized for visualization purposes.

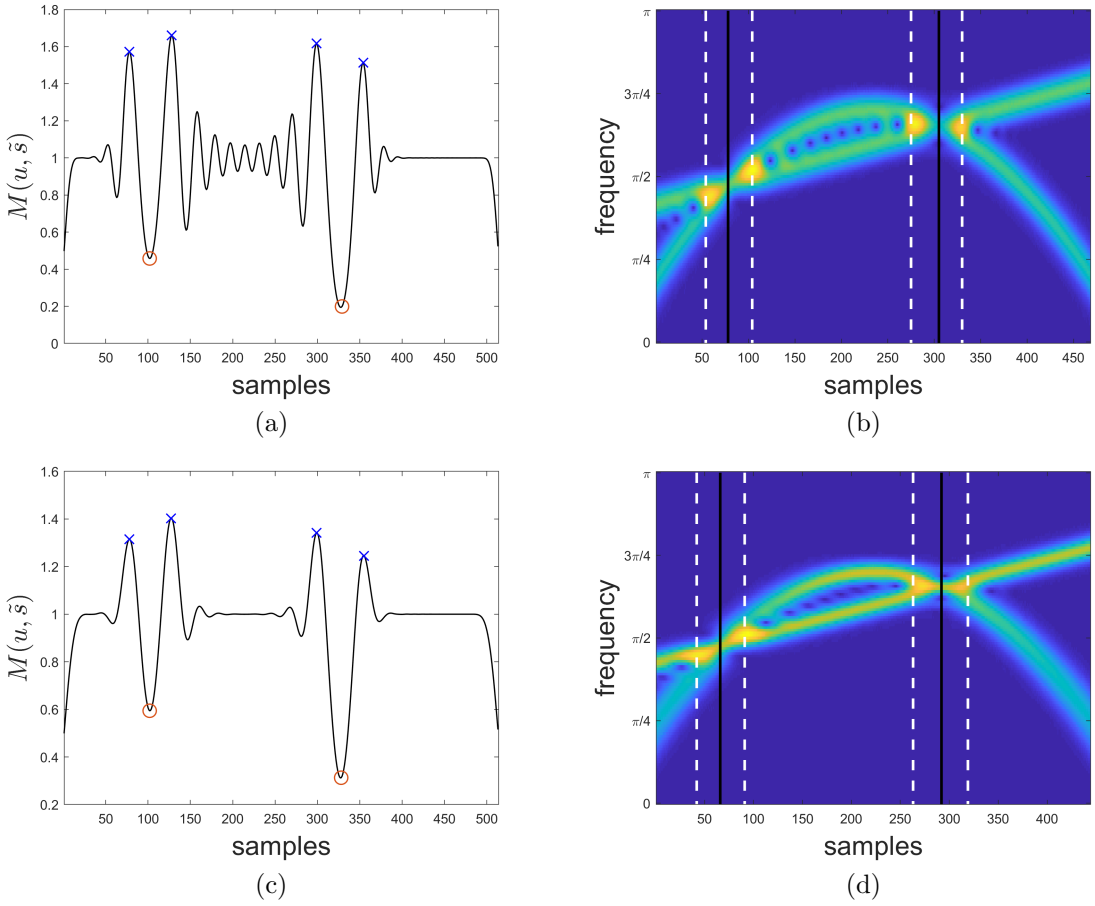


Figure 4.11: Signal $f(t) = f_1(t) + f_3(t)$ – quadratic and cubic chirps. Smoothed energy signal and spectrogram at optimal level as selected by tangent lines technique (a, b) and both KLID and JID selection methods (c, d). Circles and black lines represent interferences while crosses and dashed lines represent interference regions.

In fig. 4.13 the same test signals have been processed using the short-time Rényi entropy based method proposed in [145], where the choice of the window length is crucial. Rényi entropy of the third order has been considered. Results obtained using a 16 samples window are shown in figs. 4.13 (a) and (b). It can be observed that the counting signal tends to oscillate heavily. Conversely, using a window length of 32, small oscillations are dampened but the counting signal tends to get smeared: in fig. 4.13 (d) the two rightmost interferences are barely distinguishable. For the sake of comparison, short-time Rényi entropies have been computed using windows with lengths corresponding to the optimal scales (figs. 4.13 (c) and (d)). In this case, although the counting signals present very few oscillations, they are not precise as the proposed method in locating interference regions.

Finally, the proposed method has been tested on synthetic data simulating micro-doppler response signal of a man walking and swinging both arms. The

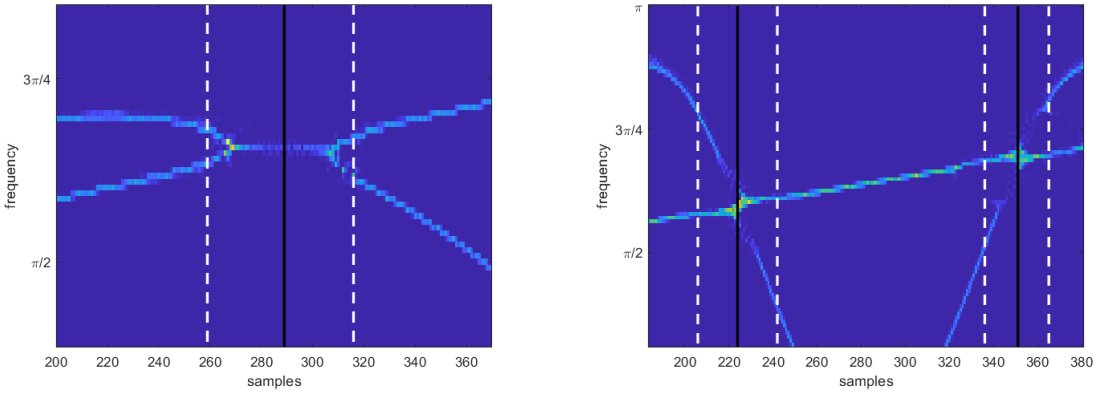


Figure 4.12: Reassigned spectrograms: interference between quadratic and cubic chirps (a) and between quadratic and sine-modulated chirps (b). White dashed lines limit interference regions and correspond to regions where reassignment fails.

simulation has been realized using the Matlab Phased Array System Toolbox.

As shown in fig. 4.15, in this case the scale selected by KLID selection method and the one selected by JID selection method are not the same.

In fig. 4.14 smoothed energy signals and relative spectrograms at the scales selected by KLID (figs. 4.14 (a) and (b)) and JID (figs. 4.14 (c) and (d)) are shown. For KLID, the smoothed energy signal is still heavily oscillating: even though it locates most of the interferences, it fails in the correct detection of the leftmost interference. This phenomenon does not happen if the scale is selected by JID, where the signal is correctly smoothed and all the interferences and interference regions are correctly estimated.

The present numerical experiment also shows that, even if the signal amplitude is not constant over time and more than two modes are present, the method correctly locates interference points where reassignment would fail.

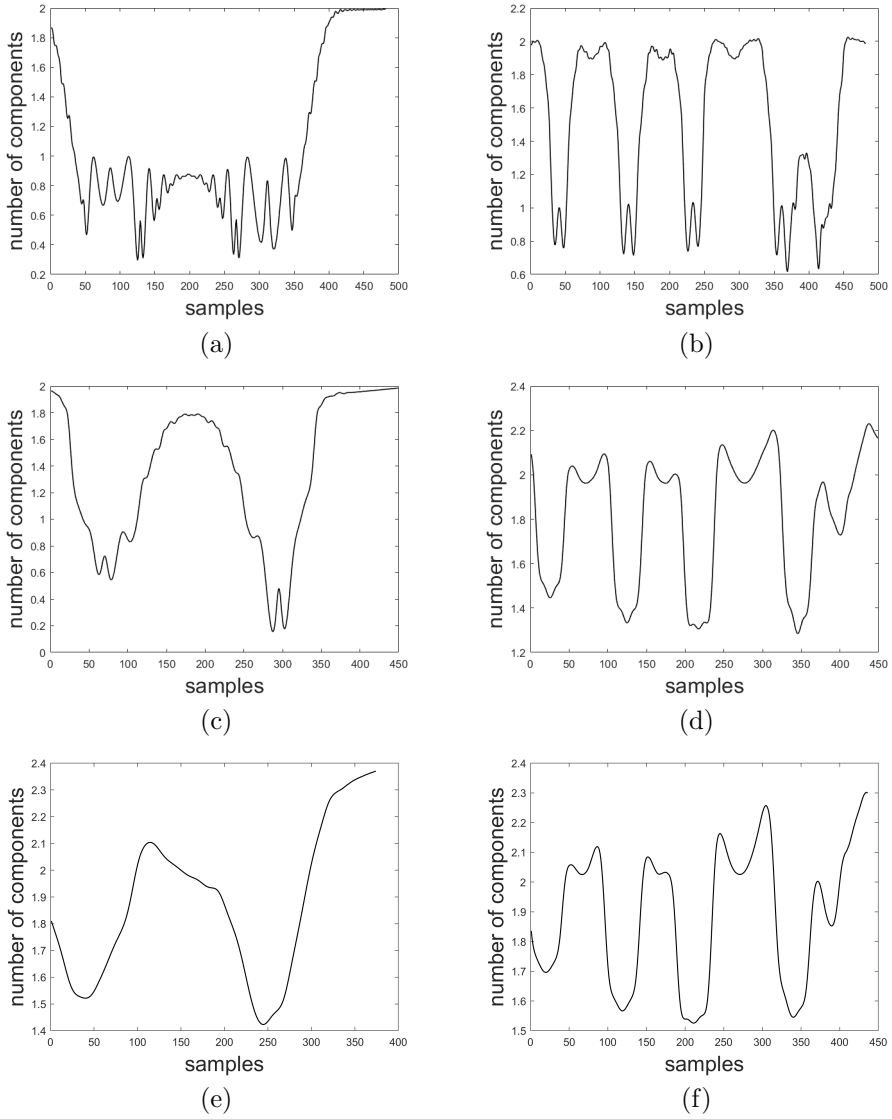


Figure 4.13: Number of components estimated using short time Rényi entropy [145]. Interference between quadratic and cubic chirps (a, c & e) and quadratic and sine-modulated chirps (b, d & f). The first row refers to a 16 taps, the second row refers to 32 taps window while the last row refers to a window whose length is equal to the optimal scale selected according to KLID and JID.

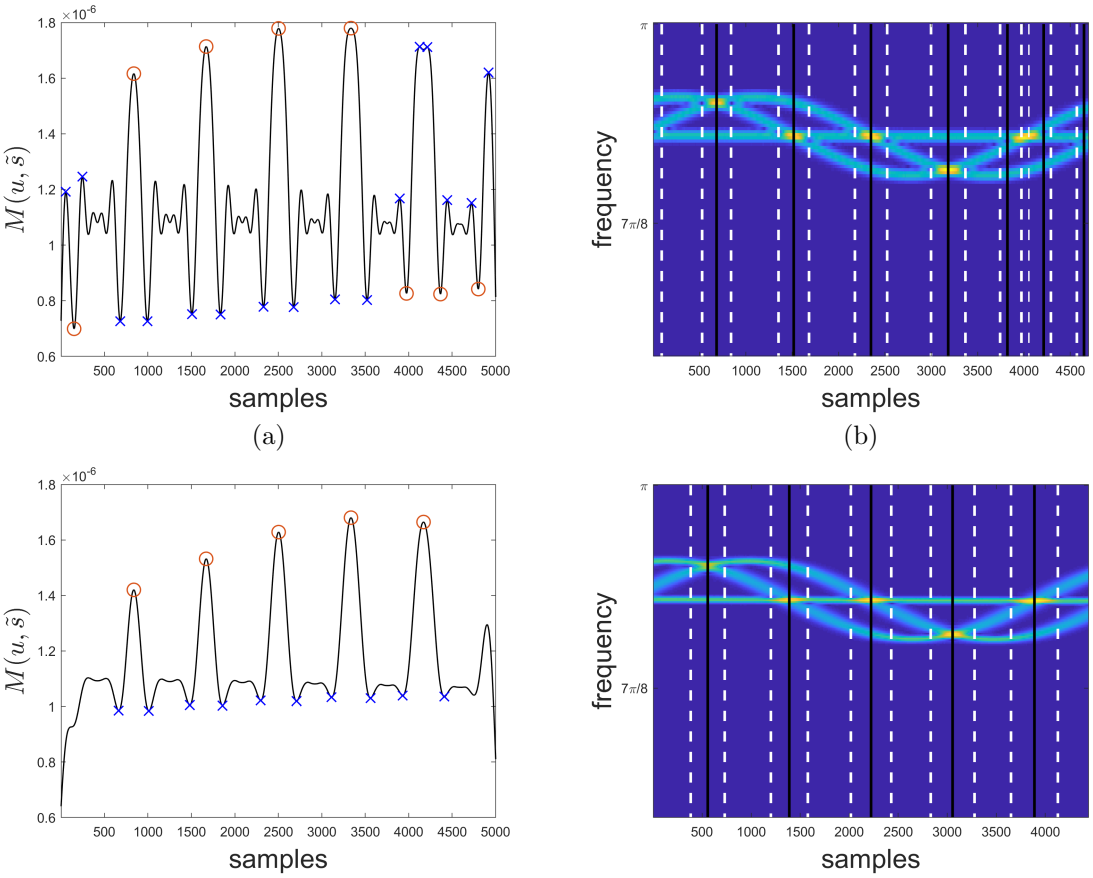


Figure 4.14: Simulation of micro doppler signal. Smoothed energy signal and spectrogram (detail) at the optimal level as selected by KLID selection method (a & b) and as selected by JID selection method (c & d). Circles and black lines represent interferences while crosses and dashed lines represent interference regions.

It turns out the proposed method could be twofold advantageous in human gait micro-doppler signal analysis. On the one hand, the proposed optimal scale selection process based on KLID and JID information measures might be implemented in existing separation and analysis strategies to automatize analysis window size selection; on the other hand the temporal interference occurrence may represent a further feature to be used in human gait recognition and classification procedures. This will be the topic for further studies.

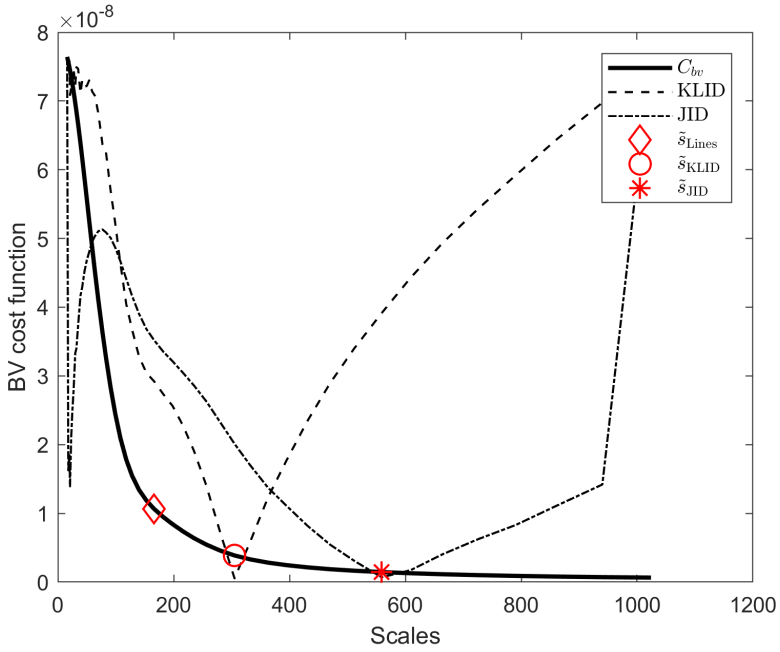


Figure 4.15: Simulation of micro doppler signal. BV cost function versus scales. Optimal points estimated with different methods are shown, along with KLID and JID. KLID and JID have been normalized for visualization purposes.

Conclusions

In this thesis an information theoretic framework for the selection of the optimum set of expansion coefficients of a function on a Hilbert basis has been defined and three different novel parameter-free methods have been presented.

The first proposed information measure for coefficients selection has been inspired by Vitanyi's well-known Normalized Information Distance (NID). It has been defined through the use of differential entropy, which has been taken as a complexity measure for deterministic functions. For this reason it has been named Entropic Normalized Information Distance (ENID). A motivation for the choice of differential entropy has been given in terms of divergence from a uniform distribution and it has been ensured that ENID is well defined and always attains a minimum, whose location defines the separation point between relevant and less relevant coefficients. In addition, the choice of differential entropy avoids the quantization step that is necessary in Shannon entropy computations, and which could destroy important infromation. Numerical error is estimated, providing conditions under which the numerical scheme converges with order 2 and a thorough study has been presented, showing that the performance of the proposed model in different wavelet bases are comparable to well-established methods in signal processing applications, and in some cases outperforms them. Numerical experiments also give evidence that ENID is strongly related to Normalized Compression Distance, a compression-based approximation of NID.

The other two information measures have been named respectively Kullback-Leibler Information Distance (KLID) and Jeffrey Information Distance JID, and are based on Kullback-Leibler divergence between deterministic functions. A suitable approximation has been proposed for them that depends on the differential entropies of the functions we are comparing plus a term depending on the integral mean of their difference. Conditions on the decay of the derivatives of the functions have been given to ensure the validity of the approximation. Numerical experiments proved that coefficients selection methods based on both KLID and JID performs well and are capable of achieving a good balance between MSE and compression ratio. Moreover, experimental evidence shows that a correspondence exists between KLID, JID and NCD.

In addition, they are robust to noise and they provide an automatic and parameter-free tool for denoising.

While ENID tends to select enough coefficients for nearly perfect reconstruction in any condition, KLID and JID proved themselves capable of recognizing the difference between coefficients that are representative of the underlying function and those that are too contaminated by noise, providing results that are comparable to the state-of-the-art global thresholding-based denoising methods and close to the oracle one.

As a result, they provide direct techniques to estimate the set of expansion coefficients that are optimal for signal reconstruction, in the sense of information contribution.

Extension to bidimensional signals (i.e. images) is one of the main direction of future work on this topic, as well as investigation of proper preprocessing techniques to suppress numerical instabilities.

Furthermore, the proposed measures performed well in the task of automatically selecting the optimal number of hierarchical Fourier descriptors, allowing for efficient, nearly lossless compression of shapes and defining a promising feature for shape recognition and classification tasks.

As a last application, KLID and JID have been used to select the optimal smoothing kernel to locate interference regions in time domain in a multicomponent chirp-like signal. Thanks to the proposed measures, the interference detection method is automatic and has proven to be robust to signals having not comparable amplitudes. In addition, it results more stable and precise than existing competing methods. This motivates future research concerning the actual use of the proposed method in this specific application. Robustness to noise and to modes with different time duration (i.e. birth and deaths of modes) is a further topic of future work.

To conclude, the success of the proposed measures in the task of determining the optimal filtering level in interference location shows that their application exceeds the sole individuation of the optimal coefficients set. Implementation of the proposed measures in other problems concerning optimal points in rate/distortion curves is a topic worth investigating.

Appendix A

Proofs

A.1 Proof of Lemma 4.1.5

The following simple technical result will be necessary.

Lemma A.1.1. *Let $g(t) : [a, b] \to \mathbb{R}$ be a twice differentiable function and let $c \in (a, b)$.*
Then

$$g'(c) = \frac{g(b) - g(a)}{b - a} + R(c),$$

where

$$|R(c)| \leq \int_a^b |g''(t)|dt.$$

Proof. Consider the first order Taylor expansion of g around c

$$g(t) = g(c) + g'(c)(t - c) + \int_c^t g''(\tau)(c - t)d\tau$$

and compute it for $t = a$ and $t = b$:

$$g(a) = g(c) - g'(c)(c - a) - \int_a^c g''(t)(c - t)dt$$
$$g(b) = g(c) + g'(c)(b - c) + \int_c^b g''(t)(c - t)dt.$$

Then,

$$g(a) - g(b) = g'(c)(b - a) + \int_a^b g''(t)(c - t)dt$$

Since $\left|\int_a^b \frac{g''(t)}{2}(c - t)dt\right| \leq |a - b| \int_a^b |g''(t)|dt$, the thesis follows. □

Proof.[Lemma 4.1.5]
Set $\lambda_{u,s}(t) = \lambda_s(u-t)$ and let us fix u. From the definition of $M_{\lambda_s}[P \circ \phi]$ and using the change of variables $\theta = \phi(t)$, it follows that

$$\begin{aligned} M_{\lambda_s}[P \circ \phi](u) &= \int_{u-s}^{u+s} \lambda_s(u-t)P(\phi(t))dt \\ &= \int_{\phi(u-s)}^{\phi(u+s)} \lambda_s(u-\phi^{-1}(\theta))P(\theta)(\phi^{-1})'(\theta)d\theta \\ &= \int_{\phi(u-s)}^{\phi(u+s)} \lambda_{u,s}(\phi^{-1}(\theta))(\phi^{-1})'(\theta)P(\theta)d\theta \\ &= \int_{\phi(u-s)}^{\phi(u+s)} \frac{d}{d\theta}\left[\Lambda_{u,s}(\phi^{-1}(\theta))\right]P(\theta)d\theta, \end{aligned}$$

where $\Lambda_{u,s}$ is a primitive of $\lambda_{u,s}$.
Assume that $\phi(u-s) = n_1 T$ and $\phi(u+s) = n_2 T$. Then, for the periodicity of P,

$$\begin{aligned} M_{\lambda_s}[P \circ \phi](t) &= \sum_{k=n_1}^{n_2-1} \int_{kT}^{(k+1)T} \frac{d}{d\theta}\left[\Lambda_{u,s}(\phi^{-1}(\theta))\right]P(\theta)d\theta \\ &= \sum_{k=n_1}^{n_2-1} \int_{0}^{T} \frac{d}{d\theta}\left[\Lambda_{u,s}(\phi^{-1}(kT+\theta))\right]P(\theta)d\theta. \end{aligned}$$

Let us consider a single element of the sum. Since both $(d/d\theta)\Lambda_{u,s}$ and P are positive, for the integral mean value theorem there exists $\theta_k \in [kT, (k+1)T]$ such that

$$\int_0^T \frac{d}{d\theta}\left[\Lambda_{u,s}(\phi^{-1}(kT+\theta))\right]P(\theta)d\theta = \frac{d}{d\theta}\left[\Lambda_{u,s}(\phi^{-1}(kT+\theta))\right]_{\theta=\theta_k}\int_0^T P(\theta)d\theta.$$

Using Lemma A.1.1 with $g = (d/d\theta)[\Lambda_{u,s} \circ \phi^{-1}]$, $a = kT$, $b = (k+1)T$ and $c = \theta_k$,

$$\begin{aligned} &\int_0^T \frac{d}{d\theta}\left[\Lambda_{u,s}(\phi^{-1}(kT+\theta))\right]P(\theta)d\theta = \\ &\qquad \left(\frac{1}{T}\left[\Lambda_{u,s}(\phi^{-1}((k+1)T)) - \Lambda_{u,s}(\phi^{-1}(kT))\right] + R(\theta_k)\right)\int_0^T P(\theta)d\theta. \end{aligned}$$

Summing up for $k = n_1, \ldots, n_2 - 1$ it holds

$$\begin{aligned} &\sum_{k=n_1}^{n_2-1} \int_0^T \frac{d}{d\theta}\left[\Lambda_{u,s}(\phi^{-1}(kT+\theta))\right]P(\theta)d\theta = \\ &\qquad \left(\Lambda_{u,s}(\phi^{-1}(\theta))\Big|_{\theta=n_1T}^{n_2T} + \sum_{k=n_1}^{n_2-1} TR(\theta_k)\right)M[P]. \end{aligned}$$

But, since $\Lambda_{u,s}$ is a primitive of $\lambda_{u,s}$ and $\int_{\mathbb{R}} \lambda_{u,s}(t)dt = 1$, then

$$\begin{aligned}\Lambda_{u,s}(\phi^{-1}(\theta))\Big|_{\theta=n_1T}^{n_2T} &= \int_{\phi^{-1}(n_1T)}^{\phi^{-1}(n_2T)} \lambda_{u,s}(t)dt \\ &= \int_{u-s}^{u+s} \lambda_{u,s}(t)dt \\ &= 1.\end{aligned}$$

As a consequence,

$$M_{\lambda_s}[P \circ \phi](u) = M[P] + \left(\sum_{k=n_1}^{n_2-1} R(\theta_k)\right) M[P].$$

For the explicit bound on the error, we use Lemma A.1.1:

$$\begin{aligned}|R(\theta_k)| &\leq \int_{kT}^{(k+1)T} \left|\frac{d^2}{d\theta^2}\Lambda_{u,s}(\phi^{-1}(\theta))\right| d\theta \\ &= \int_{kT}^{(k+1)T} \left|\frac{d}{d\theta}\left[\lambda_{u,s}(\phi^{-1}(\theta))(\phi^{-1})'(\theta)\right]\right| d\theta \\ &= \int_{kT}^{(k+1)T} \left|\frac{d}{d\theta}\left[\frac{1}{s}\lambda\left(\frac{\phi^{-1}(\theta)-u}{s}\right)(\phi^{-1})'(\theta)\right]\right| d\theta \\ &= \int_{kT}^{(k+1)T} \left|\frac{1}{s^2}\lambda'\left(\frac{\phi^{-1}(\theta)-u}{s}\right)|(\phi^{-1})'(\theta)|^2 \right. \\ &\qquad \left. +\frac{1}{s}\lambda\left(\frac{\phi^{-1}(\theta)-u}{s}\right)(\phi^{-1})''(\theta)\right| d\theta \\ &= \frac{1}{s}\int_{kT}^{(k+1)T} \left|\frac{1}{s}\lambda'\left(\frac{\phi^{-1}(\theta)-u}{s}\right)\frac{1}{|\phi'(\phi^{-1}(\theta))|^2} \right. \\ &\qquad \left. +\lambda\left(\frac{\phi^{-1}(\theta)-u}{s}\right)\frac{\phi''(\phi^{-1}(\theta))}{|\phi'(\phi^{-1}(\theta))|^3}\right| d\theta \\ &= \int_{\phi^{-1}(kT)}^{\phi^{-1}((k+1)T)} \left|\frac{1}{s^2}\lambda'\left(\frac{t-u}{s}\right)\frac{1}{|\phi'(t)|^2} \right. \\ &\qquad \left. +\frac{1}{s}\lambda\left(\frac{t-u}{s}\right)\frac{\phi''(t)}{|\phi'(t)|^3}\right| \phi'(t)dt \\ &= \int_{\phi^{-1}(kT)}^{\phi^{-1}((k+1)T)} \left|\frac{1}{s^2}\lambda'\left(\frac{t-u}{s}\right)\frac{1}{|\phi'(t)|} \right. \\ &\qquad \left. +\frac{1}{s}\lambda\left(\frac{t-u}{s}\right)\frac{\phi''(t)}{|\phi'(t)|^2}\right| dt \\ &\leq \left(\int_{\phi^{-1}(kT)}^{\phi^{-1}((k+1)T)} \left|\frac{1}{s^2}\lambda'\left(\frac{t-u}{s}\right)\frac{1}{|\phi'(t)|}\right| dt \right. \\ &\qquad \left. +\int_{\phi^{-1}(kT)}^{\phi^{-1}((k+1)T)} \frac{1}{s}\lambda\left(\frac{t-u}{s}\right)\frac{|\phi''(t)|}{|\phi'(t)|^2}dt\right).\end{aligned}$$

Summing up for $k = n_1, \dots, n_2$, by observing that $(\lambda_s)'(u) = s^{-2}\lambda'(u/s)$ we get

$$\begin{aligned}
\left|\sum_{k=n_1}^{n_2-1} R(\theta_k)\right| &\leq \left(\int_{\phi^{-1}(n_1T)}^{\phi^{-1}(n_2T)} \left|\frac{1}{s^2}\lambda'\left(\frac{t-u}{s}\right)\right| \frac{1}{|\phi'(t)|}dt \right. \\
&\qquad \left. + \int_{\phi^{-1}(n_1T)}^{\phi^{-1}(n_2T)} \frac{1}{s}\lambda\left(\frac{t-u}{s}\right)\frac{|\phi''(t)|}{|\phi'(t)|^2}dt\right) \\
&= \left(\int_{u-s}^{u+s} \left|\frac{1}{s^2}\lambda'\left(\frac{t-u}{s}\right)\right| \frac{1}{|\phi'(t)|}dt \right. \\
&\qquad \left. + \int_{u-s}^{u+s} \frac{1}{s}\lambda\left(\frac{t-u}{s}\right)\frac{|\phi''(t)|}{|\phi'(t)|^2}dt\right) \\
&= \left(|(\lambda_s)'| * \frac{1}{|\phi'|}\right)(u) + \left(|\lambda_s| * \frac{|\phi''|}{|\phi'|^2}\right)(u).
\end{aligned}$$

In general, $\phi^{-1}(n_1T) \neq u - s$ and $\phi^{-1}(n_2T) \neq u + s$; as a result, another error term has to be considered. Since its estimate is the same for the left $(u-s)$ and right $(u+s)$ boundary, we limit the proof to the first case, and then

Following the same technique as before,

$$\begin{aligned}
\int_{u-s}^{\phi^{-1}(n_1T)} \lambda_s(u-t)P(\phi(t))dt &\leq \int_{\phi^{-1}((n_1-1)T)}^{\phi^{-1}(n_1T)} \lambda_s(t-\tau)P(\phi(\tau))d\tau \\
&= \int_{(n_1-1)T}^{n_1T} \frac{d}{d\theta}\left[\Lambda_{u,s}(\phi^{-1}(\theta))\right]P(\theta)d\theta \\
&= \frac{d}{d\theta}\left[\Lambda_{u,s}(\phi^{-1}(\theta_{n_1}))\right]\int_{(n_1-1)T}^{n_1T} P(\theta)d\theta \\
&= (\phi^{-1})'(\theta_{n_1})\int_{u-s}^{\phi^{-1}(\theta_{n_1})} \lambda_{u,s}(t)dt \int_0^T P(\theta)d\theta \\
&\leq (\phi^{-1})'(\theta_{n_1})TM[P] \\
&= \frac{T}{\phi'(\phi^{-1}(\theta_{n_1}))}M[P] \\
&\leq \frac{T}{\Theta}M[P].
\end{aligned}$$

□

A.2 Proof of Lemma 4.1.8

Proof. Let us fix u. For any $t_0 \in [u-s, u+s]$, there exists $\tilde{t} \in [u-s, u+s]$ such that

$$\begin{aligned}
f(t) &= \sin(\phi(t)) \\
&= \sin(\phi(t_0) + \phi'(\tilde{t})(t - t_0)).
\end{aligned}$$

Then, using Plancherel theorem,

$$\begin{aligned}
|(\lambda_s * f)(u)| &= \left| \int_{\mathbb{R}} \lambda_s(u-t) f(t) dt \right| \\
&= \left| \int_{\mathbb{R}} \lambda_s(u-t) \sin(\phi(t_0) + \phi'(\tilde{t})(t-t_0)) dt \right| \\
&= \left| \int_{\mathbb{R}} e^{-i(u+\phi(t_0)-\phi'(\tilde{t})t_0)\xi} \hat{\lambda}_s(-\xi) \frac{\delta(\xi-\phi'(\tilde{t})) - \delta(\xi+\phi'(\tilde{t}))}{2i} d\xi \right| \\
&= \left| \int_{\mathbb{R}} e^{-i(u+\phi(t_0)-\phi'(\tilde{t})t_0)\xi} \hat{\lambda}(-s\xi) \frac{\delta(\xi-\phi'(\tilde{t})) - \delta(\xi+\phi'(\tilde{t}))}{2i} d\xi \right| \\
&\leq |\hat{\lambda}(s\phi'(\tilde{t}))|.
\end{aligned}$$

Taking the supremum for $\tilde{t} \in [u-s, u+s]$ in the last term completes the proof. □

□

Bibliography

[1] C. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, 1948.

[2] D. Huffman, "A method for the construction of minimum-redundancy codes," *Resonance*, vol. 11, pp. 91–99, 02 2006.

[3] J. Rissanen, "Generalised kraft inequality and arithmetic coding," *IBM J. Res. Dev.*, no. 20, 1976.

[4] J. Ziv and A. Lempel, "Compression of individual sequences via variable-rate coding," vol. 24, no. 5, p. 530–536, 2006.

[5] G. Hudson, A. Léger, B. Niss, I. Sebestyén, and J. Vaaben, "JPEG-1 standard 25 years: past, present, and future reasons for a success," *Journal of Electronic Imaging*, vol. 27, no. 4, pp. 1 – 19, 2018.

[6] H. G. Musmann, "Genesis of the mp3 audio coding standard," *IEEE Transactions on Consumer Electronics*, vol. 52, no. 3, pp. 1043–1049, 2006.

[7] I. Daubechies, *Ten Lectures on Wavelets*. CBMS-NSF Regional Conference Series in Applied Mathematics, 1992.

[8] S. Mallat, *A wavelet tour of signal processing*. Academic Press, 1998.

[9] D. L. Donoho and I. M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, vol. 81, no. 3, 1994.

[10] R. R. Coifman and D. Donoho, "Translation invariant de-noising," *Technical Report 475, Dept. of Statistic, Stanford University*, May 1995.

[11] R. A. DeVore, "Nonlinear approximation," *Acta Numerica*, vol. 7, 1998.

[12] R. DeVore, G. Kyriazis, and P. Wang, "Multiscale characterizations of besov spaces on bounded domains," *Journal of Approximation Theory*, vol. 93, no. 2, pp. 273 – 292, 1998.

[13] R. R. Coifman and M. V. Wickerhauser, "Entropy-based algorithms for best basis selection," *IEEE Transactions on Information Theory*, vol. 38, march 1992.

[14] N. K. Karmarkar, "A new polynomial-time algorithm for linear programming.," *Combinatorica*, vol. 4, 1984.

[15] I. J. Lustig, R. E. Marsten, and D. F. Shanno, "Interior point methods for linear programming: computational state of the art," *ORSA J. on Comput.*, vol. 6, 1994.

[16] S. Chen and D. Donoho, "Atomic decomposition by basis pursuit," *SPIE international Conference on Wavelets, San Diego, July 1995*, 1995.

[17] G. B. Dantzig, "On the non-existence of tests of "student's" hypothesis having power functions independent of σ," *Annals of Mathematical Statistics*, vol. 11, 1940.

[18] M. Gr otschel, L. Lovász, and A. Schrijver, *Geometric Algorithms and Combinatorial Optimization.* Springer, 1988.

[19] S. Mallat and Z. Zhang, "Matching pursuit with time-frequency dictionaries," *IEEE Transaction on Signal Processing*, vol. 41, no. 12, 1993.

[20] F. B. S. Mallat, "Matching pursuit: Adaptive representations of images," *Computational and Applied Mathematics*, vol. 51, no. 2, 1996.

[21] G. M. Davis, S. Mallat, and Z. Zhang, "Adaptive time-frequency decompositions," *SPIE J. of Opt. Engin.*

[22] Y. C. Pati, R. Rzaifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with application to wavelet decomposition,"

[23] Y. C. Eldar, P. Kuppinger, and H. Bolcskei, "Block-sparse signals: Uncertainty relations and efficient recovery," *IEEE Transactions on Signal Processing*, vol. 58, no. 6, 2010.

[24] S. Huang and T. D. Tran, "Sparse signal recovery via generalized entropy functions minimization," *IEEE Transactions on Signal Processing*, vol. 67, no. 5, 2019.

[25] D. Needell and J. A. Tropp, "Cosamp: Iterative signal recovery from incomplete and inaccurate samples," *Applied and computational harmonic analysis*, vol. 26, no. 3, 2009.

[26] J. Wang, S. Kwon, and B. Shim, "Generalized orthogonal matching pursuit," *IEEE Transactions on Signal Processing*, vol. 60, no. 12, 2012.

[27] A. M. Bruckstein, M. Elad, and M. Zibulevsky, "On the uniqueness of nonnegative sparse solutions to underdetermined systems of equations," *IEEE Transactions on Information Theory*, vol. 54, no. 11, 2008.

[28] J. Shen and S. Mousavi, "Exact support and vector recovery of constrained sparse vectors via constrained matching pursuit," *arXiv:1903.07236v2*, 2019.

[29] A. Quarteroni, R. Sacco, and F. Saleri, *Numerical Mathematics.* Springer, 2001.

[30] J. Kaipio and E. Somersalo, *Statistical and Computational Inverse Problems.* Springer, 2005.

[31] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society, Series B (Methodological)*, vol. 58, no. 1, 1996.

[32] R. J. Tibshirani, "The lasso problem and uniqueness," *Electronic Journal of Statistics*, vol. 7, 2013.

[33] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society: Series B*, vol. 67, no. 2, 2005.

[34] R. Chartrand, "Exact reconstruction of sparse signals via nonconvex minimization," *IEEE Signal Process. Lett.*, vol. 14, 2007.

[35] S. Foucart and M. Lai, "Sparsest solutions of underdetermined linear systems via l_q-minimization for $0 < q \leq 1$," *Appl. Comput. Harmon. Anal*, vol. 26, 2009.

[36] J. Fan and R. Li, "Variable selection via nonconcave penalized likelihood and its oracle properties," *Journal of the American Statistical Association*, vol. 96, no. 456, p. 1348–1360, 2001.

[37] E. Candes, M. Wakin, and S. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, p. 877–905, 2008.

[38] C. Zhang, "Nearly unbiased variable selection under min-imax concave penalty," *The Annals of Statistics*, vol. 38, no. 2, pp. 894–942.

[39] D. Geman and C. Yang, "Nonlinear image recovery with half-quadratic regularization," *IEEE Transactions on Image Processing*, vol. 4, no. 7, pp. 932–964, 1995.

[40] J. Trzasko and A. Manduca, "Relaxed conditions for sparse signal recovery with general concave priors," *IEEE Transactions on Signal Processing*, vol. 57, no. 11, pp. 4347–4354, 2009.

[41] Y. J. Gong, P. and C. Zhang, "Multi-stage multi-task feature learning," in *NIPS*.

[42] T. Zhang, "Analysis of multi-stage convex relaxation forsparse regularization," *Journal of Machine Learning Research*, vol. 11, pp. 1081–1107, 2010.

[43] R. Chartrand and W. Yin, "Iteratively reweighted algorithms for compressive sensing," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2008.

[44] P. Combettes and J. Pesquet, "Proximal splitting methods in signal processing," in *Fixed-point Algorithms for Inverse Problems in Science and Engineering*, Springer.

[45] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183 – 202.

[46] M. Hong, Z. Luo, and M. Razaviyayn, "Convergence analysis of alternating direction method of multipliers for a family of nonconvex problems," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3836–3840, April 2015.

[47] T. Blumensath and M. Davies, "Iterative hard thresholding for compressed sensing," *Applied and Computational Harmonic Analysis*, vol. 27, 2009.

[48] T. Blumensath and M. Davies, "Normalized iterative hard thresholding: Guaranteed stability and performance," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, 2010.

[49] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, pp. 1289–1306, April 2006.

[50] E. Candès and T. Tao *Near-optimal signal recovery from random projections: Universal encoding strategies.*

[51] E. Candès and T. Tao, "Decoding by linear programming," *IEEE Trans. Inform. Theory*, vol. 51, December 2005.

[52] E. Candès, "The restricted isometry property and its implications for compressed sensing," *C. R. Acad. Sci. Sér. I 346*, 2008.

[53] Q. Mo and S. Li, "New bounds on the restricted isometry constant δ2k," *Applied and Computational Harmonic Analysis*, vol. 31, no. 3, pp. 460 – 468, 2011.

[54] Z. Wen, W. Yin, D. Goldfarb, and Y. Zhang, "A fast algorithm for sparse reconstruction based on shrinkage, subspace optimization, and continuation," *SIAM Journal on Scientific Computing*, vol. 32, no. 4, 2010.

[55] W.Yin, S. Osher, D. Goldfarb, and J. Darbon, "Bregman iterative algorithms for ℓ_1-minimization with applications to compressed sensing," *SIAM Journal on Imaging sciences*, vol. 1, no. 1, pp. 143 –168, 2008.

[56] M. J. Wainwright, "Sharp thresholds for high-dimensional and noisy sparsity recovery using ℓ_1-constrained quadratic programming (lasso)," *IEEE transactions on information theory*, vol. 55, no. 5, 2009.

[57] D. P. O'Leary and P.C.Hansen, "The use of the l–curve in the regularization of discrete ill–posed problems," *SIAM J. Sci. Comp.*, vol. 14, 1993.

[58] B. K. Natarajan, "Filtering random noise from deterministic signals via data compression," *IEEE Transactions on Information Theory*, vol. 43, no. 11, 1995.

[59] A. N. Kolmogoroff, *Grundbegriffe der Wahrscheinlichkeitsrechnung.* Chelsea Publishing Company, 1946.

[60] G. Puccetti and R. Wang, "Extremal dependence concepts," *Statistical Sceince*, vol. 30, no. 4, 2015.

[61] R. Gonzalez and R. Woods, *Digital Image Processing, 3rd ed.* Prentice-Hall, New Jersey, 2008.

[62] B. Barkat and K. Abed-Meraim, "Algorithms for blind components separation and extraction from the time-frequency distribution of their mixture," *EURASIP Journal on App. Sig. Proc.*, vol. 13, 2004.

[63] Z. Li and N. Martin, "A time-frequency based method for the detection and tracking of multiple non-linearly modulated components with birth and deaths," *IEEE Trans. on Sig. Proc.*, vol. 64, 2016.

[64] F. F. Auger, Y. P. Lin, S. Mclaughlin, and S. e. a. Meigmen, "Time-frequency reassignment and synchrosqueezing: an overview," *IEEE Signal Processing Magazine*, vol. 30, 2013.

[65] Y. Ding and J. Tang, "Micro-doppler trajectory estimation of pedestrians using a continuous-wave radar," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 52, no. 9, 2014.

[66] S. S. Ram and H. Ling, "Analysis of microdopplers from human gait using reassigned joint time-frequency transform," *Electronics Letters*, vol. 43, no. 23, 2007.

[67] G. M. Davis, S. Mallat, and M. Avelanedo, "Greedy adaptive approximations," *Journal of Constructive Approximation*, vol. 13, no. 1, 1997.

[68] J. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Trans. Inform. Theory*, vol. 50, 2004.

[69] P. Schelkens, A. Skodras, and T. Ebrahimi, eds., *The JPEG 2000 Suite*. Wiley, August 2009.

[70] W. F. Schreiber, "Advanced television systems for terrestrial broadcasting: Some problems and some proposed solutions," *Proceedings of the IEEE*, vol. 83, no. 6, 1995.

[71] D. Taubman and M. Marcellin, "Jpeg2000: standard for interactive imaging," *Proceedings of the IEEE*, vol. 90, no. 8, 2002.

[72] V. Bruni, S. Marconi, and D. Vitulano, "Time-scale atoms chains for transients detection in audio signals," *IEEE Transactions on Audio, Speech and LAnguage Processing*, vol. 18, no. 3, pp. 420 – 433, March 2010.

[73] F. Angelini, V. Bruni, I. Selesnick, and D. Vitulano, "Adaptive scale selection for multiscale image denoising," *Proc. of ACIVS 2015, Catania, Italy, special issue in Lecture Notes in Computer Science*, vol. 9386, no. 12, pp. 81–92, 2015.

[74] V. Bruni, B. Piccoli, and D. Vitulano, "A fast computation method for time-scale signal denoising," *Signal Image and Video Processing*, vol. 3, no. 1, pp. 63–83, May 2009.

[75] V. Bruni and D. Vitulano, "Time scale similarities for robust image denoising," *Journal of Mathematical Imaging and Vision*, vol. 44, no. 1, pp. 52–64, September 2012.

[76] V. Bruni, D. De Canditiis, and D. Vitulano, "Time-scale energy based analysis of contours of real world shapes," *Mathematics and Computers in Simulation*, vol. 12, no. 12, pp. 2891 – 2907, August 2012.

[77] I. Daubechies, "Ten lectures on wavelets," *CBMS-NSF Regional Conference Series in Applied Mathematics*, 1992.

[78] J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex fourier series," *Mathematics of Computation*, vol. 19, 1965.

[79] C. V. Loan, *Computational Frameworks for the Fast Fourier Transform.* SIAM, 1992.

[80] S. Jaffard, "Pointwise smoothness, two-microlocalization and wavelet coefficients.," *Publicacions Matemàtiques*, vol. 35, 1991.

[81] S. Mallat and W. Hwang, "Singularity detection and processing with wavelets," *IEEE Transactions on Information Theory*, vol. 38, no. 2, 1992.

[82] M. Holschneider, R. Kronland-Martinet, J. Morlet, and P. Tchamitchian, "A real-time algorithm for signal analysis with the help of the wavelet transform," *Wavelets, Time-Frequency Methods and Phase Space*, 1989.

[83] R. R. Coifman and D. L. Donoho, "Translation invariant denoising,"

[84] M. Frazier and B. Jawerth, "Decomposition of besov spaces," *Indiana Univ. Math. J.*, vol. 34, 1985.

[85] Y. Meyer, "Principe d'incertitude, bases hilbertiennes et algèbres d'opératuers," *Séminaire Bourbaki*, vol. 662, 1986.

[86] G. Leoni, *A First Course in Sobolev Spaces: Second Edition.* American Mathematical Society, 2017.

[87] S. Mallat and F. Falzon, "Analysis of low bit rate image transform coding," *IEEE Trans. Signal. Proc.*, vol. 46, 1998.

[88] S. Quien and D. Chen, "Signal representation via adaptive normalized gaussian functions," *IEEE Trans. Signal. Proc.*, vol. 36, 1994.

[89] D. L. Donoho and I. M. Johnstone, "Adapting to unknown smoothness via wavelet shrinkage," *Journal of the American Statistical Association*, vol. 90, 12 1995.

[90] C. Stein, "Estimation of the mean of a multivariate normal distribution," *Annals of Statistics*, vol. 9, pp. 1135 – 1151, 1994.

[91] Breiman, "Better subset regression using th nonnegative garrote," *Technometrics*, vol. 37, no. 4, 1995.

[92] T. Hastie, R. J. Tibshirani, and M. Wainwright, *Statistical Learning with Sparsity: the Lasso and generalizations.* Chapman & All, 2015.

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

[94] N. Parikh, "Proximal algorithms," *Foundations and Trends in Optimization*, vol. 1, no. 3, 2014.

[95] P. Gong, C. Zhang, Z. Lu, J. Huang, and J. Ye, "A general iterative shrinkage and thresholding algorithm for non-convex regularized optimization problems," *Machine learning : proceedings of the International Conference. International Conference on Machine Learning*, vol. 28, 03 2013.

[96] C. O'Brien and M. D. Plumbley, "Inexact proximal operators for ℓ_p-quasinorm minimization," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4724–4728, April 2018.

[97] A. Mousavi, A. Maleki, and R. G. Baraniuk, "Consistent parameter estimation for lasso and approximate message passing," *Ann. Statist.*, vol. 45, no. 6, 2017.

[98] T. M. Cover and J. A. Thomas, *Elements of Information Theory, 2nd Edition*. Wiley, 2005.

[99] B. Natarajan, "Sparse approximate solutions to linear systems," *SIAM J. Comput.*, vol. 24, 1995.

[100] G. Wahba, "Practical approximate solutions to linear operator equations when the data are noisy," *SIAM J. Num. Anal.*, vol. 14, 1977.

[101] G. H. Golub, M. Heath, and G. Wahba, "Generalized cross-validation as a method for choosing a good ridge parameter," *Technometrics*, vol. 21, no. 2, 1979.

[102] S. Mousavi, M. M. R. Taghiabadi, and R. Ayanzadeh, "A survey on compressive sensing: Classical results and recent advancements,"

[103] S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing*. Springer, 2013.

[104] G. Shafer and V. Vovk, "The sources of kolmogorov's "grundbegriffe"," *Statistical Science*, vol. 21, no. 1, pp. 70–98, 2006.

[105] P. J. S. G. Ferreira, "Sorting continuous-time signals: Analog median and median-type filters," *IEEE Transactions on Signal Processing*, vol. 49, no. 11, 2001.

[106] G. F. D. Duff, "A general integral inequality for the derivative of an equimisurable rearrangement," *Can. J. Math*, vol. XXVIII, no. 4, 1976.

[107] G. F. D. Duff, "Integral inequalities for equimeasurable rearrangements," *Canadian Journal of Mathematics*, vol. 22, no. 2, p. 408–430, 1970.

[108] C. Bennet and R. Sharpley, *Interpolation of Operators*. Academic Press, 1988.

[109] A. A. Fedotov, P. Harremoes, and F. Topsoe, "Refinements of pinsker's inequality," *IEEE Transactions on Information Theory*, vol. 49, no. 6, pp. 1491–1498, 2003.

[110] M. S. Pinsker, "Information and information stability of random variables and processes," *U.S.S.R., Moscow:Izv. Akad. Nauk*, 1960.

[111] M. D. Reid and R. Williamson, "Generalised pinsker inequalities," *ArXiv*, vol. abs/0906.1244, 2009.

[112] D. Pollard, "Total variation distance between measures," in - *http://www.stat.yale.edu/ pollard/Courses/607.spring05/handouts/Totalvar.pdf*.

[113] I. Sason, "An improved reverse pinsker inequality for probability distributions on a finite set," *ArXiv*, vol. abs/1503.03417, 2015.

[114] M. Carro, "A useful estimate for the decreasing rearrangement of the sum of functions," *The Quarterly Journal of Mathematics*, vol. 55, pp. 41–45, 03 2004.

[115] S. Verdu and D. Guo, "A simple proof of the entropy-power inequality," *IEEE Transactions on Information Theory*, vol. 52, no. 5, pp. 2165–2166, 2006.

[116] S. G. Chang, B. Yu, and M. Vetterli, "Adaptive wavelet thresholding for image denoising and compression," *IEEE Transactions on Image Processing*, vol. 9, no. 9, September 2000.

[117] J. Rissanen, "A universal prior for integers and estimation by minimum description length," *Ann. Statist.*, vol. 11, no. 3, 1984.

[118] P. Grünwald, "A tutorial introduction to the minimum description length principle," in *Advances in Minimum Description Length: Theory and Applications*, MIT Press, 2005.

[119] V. Bruni, L. Della Cioppa, and D. Vitulano, "An entropy-based approach for shape description," *Proceedings of EUSIPCO 2018*, 2018.

[120] V. Bruni, L. Della Cioppa, and D. Vitulano, "A normalized information based method for efficient signal representation," *Proceedings of MASCOT 2018*, 2019.

[121] M. Li and P. Vitanyi, *An Introduction to Kolmogorov Complexity and Its Applications, 4th edition*. Springer, 2019.

[122] M. Li, X. Chen, X. Li, B. Ma, and P. M. B. Vitanyi, "The similarity metric," *IEEE Transactions on Information Theory*, vol. 50, no. 12, 2004.

[123] R. Cilibrasi and P. M. B. Vitanyi, "Clustering by compression," *IEEE Transactions on Information Theory*, vol. 51, no. 4, pp. 1523 – 1545, 2005.

[124] D. Cerra and M. Datcu, "Algorithmic relative complexity," *Entropy*, vol. 13, 2011.

[125] D. Benedetto, E. Caglioti, and V. Loreto, "Language trees and zipping," *Phys. Rev. Lett.*, vol. 88, p. 048702, Jan 2002.

[126] A. Kaltchenko, "Algorithms for estimating information distance with application to bioinformatics and linguistics," *Canadian Conference on Electrical and Computer Engineering 2004 (IEEE Cat. No.04CH37513)*, vol. 4, pp. 2255–2258 Vol.4, 2004.

[127] P. J. Rousseeuw and C. Croux, "Alternatives to the median absolute deviation," *Journal of the American Statistical Association*, vol. 88, no. 424, pp. 1273–1283, 1993.

[128] A. Khotanzad and Y. H. Hong, "Invariant image recognition by zernike moments," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 50, no. 12, 1990.

[129] A. Jalba, M. Wilkinson, and J. Roerdink, "Shape representation and recognition through morphological curvature scale-spaces," *IEEE Trans. Image Process.*, vol. 15, no. 2, 2006.

[130] F. Mokhtarian, S. Abbasi, and J. Kittler, "Robust and efficient shape indexing through curvature scale space," *British Machine Vision Conference*, 1996.

[131] T. Cootes, C. Taylor, D. Cooper, and J. Graham, "Active shape models-their training and application," *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38 – 59, 1995.

[132] C. Yang, O. Tiebe, K. Shirahama, and M. Grzegorzek, "Object matching with hierarchical skeletons," *Pattern Recogn.*, vol. 55, p. 183–197, July 2016.

[133] E. Persoon and K. Fu, "Shape discrimination using fourier descriptors," *IEEE Trans. Syst. Man Cybern.*, vol. 7, no. 3, 1997.

[134] D. Zhang and G. Lu, "Study and evaluation of different fourier methods for image retrieval," *Image and Vision Computing*, vol. 23, 2005.

[135] M. A. Savelonas, I. Pratikakis, and K. Sfikas, "Partial 3d object retrieval combining local shape descriptors with global fisher vectors," in *Proceedings of the 2015 Eurographics Workshop on 3D Object Retrieval*, 3DOR '15, (Goslar, DEU), p. 23–30, Eurographics Association, 2015.

[136] C. Dalitz, S. Brandt, C.and Goebbels, and D. Kolanus, "Fourier descriptors for broken shapes," *EURASIP Journal on Advances in Signal Processing*, vol. 2013, p. 161, 10 2013.

[137] E. Sokic and S. Konjicija, "Novel fourier descriptor based on complex coordinates shape signature," pp. 1–4, 06 2014.

[138] D. Zhang and G. Lu, "Shape-based image retrieval using generic fourier descritor," *Signal Processing: Image Communication*, vol. 17, pp. 825–848, 11 2002.

[139] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N.-C. Yen, C. C. Tung, and H. H. Liu, "The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis," *Proc. R. Soc. Lond. A*, 1998.

[140] F. Auger and P. Flandrin, "Improving the readability of time-frequency and time-scale representations by the reassignment method," *IEEE Trans. on Sig. Proc.*, vol. 43, 1995.

[141] S. Meignen, T. Oberlin, and M. S., "A new algorithm for multicomponent signal analysis based on synchrosqueezing: With an application to signal sampling and denoising," *IEEE Trans. Signal Process.*, vol. 60, no. 11, 2011.

[142] I. Daubechies, J. Lu, and H. T. Wu, "Synchrosqueezed wavelet transforms:an empirical mode decomposition-like tool," *Appl. Comput. Harmon. Anal.*, vol. 30, no. 2, 2011.

[143] V. Bruni, M. Tartaglione, and D. Vitulano, "On the time-frequency reassignment of interfering modes in multicomponent fm signals," *Proc. of EUSIPCO 2018*, 2018.

[144] A. Rényi, "On measures of entropy and information," *Proc 4th Berkeley Simp., in Mathematics of Statistic and Probability*, vol. 1, 1961.

[145] V.Sucic, N. Saulig, and B. Boashas, "Analysis of local time-frequency entropy features for nonstationary signal components time support detection," *Digital Signal Processing*, vol. 34, 2014.

[146] N. Saulig, V. Sucic, S. Stankovic, I. Orovic, and B. Boashash, "Signal content estimation based on the short-term time-frequency rényi entropy of the s-method time-frequency distribution," *Proc. of IWSSIP 2012*, 2012.

[147] N. Saulig, N. Pustelnik, P. Borgnat, P. Flandrin, and V. Sucic, "Instantaneous counting of components in nonstationary signals," *Proc. of EUSIPCO 2013*, 2013.

[148] V. Sucic, N. Saulig, and B. Boashash, "Estimating the number of components of a multicomponent nonstationary signal using the short-term time-frequency rényi entropy," *EURASIP Journal on Advances in Signal Processing 2011*, vol. 125, 2011.

[149] R. G. Baraniuk, P. Flandrin, A. J. E. M. Janssen, and O. J. J. Michel, "Measuring time-frequency informations content using renyi entropies," *IEEE Trans. Inf. Theory.*, vol. 47, no. 4, 2001.

[150] J. Lerga, N. Saulig, R. Lerga, and . Milanović, "Effects of tfd thresholding on eeg signal analysis based on the local rényi entropy," *2nd International Multidisciplinary Conference on Computer and Energy Science (SpliTech)*, 2017.

[151] J. Lerga, N. Saulig, V. Mozetic, and R. Lerga, "Number of eeg signal components estimated using the short-term renyi entropy," *2016 International Multidisciplinary Conference on Computer and Energy Science (SpliTech (2016)*, 2016.

[152] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, 1978.

[153] L. B., C. Ioana, and M. G. Amin, "Human gait classification using microdoppler time-frequency signal representations," *2010 IEEE Radar Conference, Washington, DC,*, 2010.

[154] T. Sakamoto, T. Sato, P. J. Aubry, and A. G. Yarovoy, "Texture-based automatic separation of echoes from distributed moving targets in uwb radar signals," *IEEE Trans. on Geos. and Rem. Sen.*, vol. 53, no. 1, 2015.