Sparse linear models and denoising

1 Introduction

1.1 Definition and motivation

Finding representations of signals that allow to process them more effectively is a central problem in signal processing and data analysis. We will consider linear models, where the signal x is a represented as a sum of weighted atoms $\phi_1, \phi_2, \ldots, \phi_m$

$$x = \sum_{i=1}^{m} c_i \, \phi_i. \tag{1}$$

The set of atoms $\{\phi_1, \phi_2, \ldots, \phi_m\}$ is often called a dictionary. The m-dimensional vector of coefficients c is the representation of the signal x in the dictionary.

A sparse linear model consists of the sum of a small number of atoms selected from a certain dictionary

$$x = \sum_{i \in \mathcal{I}} c_i \, \phi_i \qquad |\mathcal{I}| \ll m, \tag{2}$$

here $\mathcal{I}$ is a set of indices of atoms in the dictionary. Its cardinality is significantly smaller than the number of atoms in the dictionary.

Signals in specific applications tend to have similar characteristics. In this lecture we will study how to use sparse linear models to exploit this common structure and enhance data analysis in applications such as compression and denoising. These models may also be applied to tackle inverse problems, as we will see in the next couple of lectures.

Finding dictionaries that are able to represent classes of signals, such as images or speech, parsimoniously has been an extremely active area of research in the last 30 years. It is very related to the problem of computing useful features in machine learning. There are two main approaches to building sparsifying dictionaries. The first is to use domain knowledge and intuition. This is the approach on which we will focus in this lecture. The second approach is to learn the transformation directly from a database of signals in the class of interest. We will study such methods later on in the course.

1.2 Bases and overcomplete dictionaries

If the atoms of the dictionary $\{\phi_1, \ldots, \phi_n\}$ form an orthonormal basis of $\mathbb{R}^n$ (or C^n) fitting the coefficients of a linear model is extremely simple. If (1) and the atoms are orthonormal then

$$c_i = \langle \phi_i, x \rangle \,. \tag{3}$$

The coefficients are obtained by computing inner products with the atoms. The representation is simply

$$x = \sum_{i=1}^{m} \langle \phi_i, x \rangle \, \phi_i. \tag{4}$$

If we construct a matrix using the atoms as columns

$$U := \begin{bmatrix} \phi_1 & \phi_2 & \cdots & \phi_n \end{bmatrix} \tag{5}$$

then $c = U^T x$ (or $c = U^*$ in the complex case).

If the atoms $\{\phi_1, \ldots, \phi_n\}$ form a basis, then the matrix

$$B := \begin{bmatrix} \phi_1 & \phi_2 & \cdots & \phi_n \end{bmatrix}, \tag{6}$$

has an inverse matrix B^{-1}. The rows of B^{-1}, which we denote by $\theta_1, \theta_2, \ldots, \theta_n$, can be interpreted as dual atoms. We have

$$x = BB^{-1}x = \sum_{i=1}^{m} \langle \theta_i, x \rangle \, \phi_i \tag{7}$$

so

$$c_i = \langle \theta_i, x \rangle \,. \tag{8}$$

If the atoms $\{\phi_1, \ldots, \phi_m\}$ are linearly independent and $m > n$, then the dictionary

$$D := \begin{bmatrix} \phi_1 & \phi_2 & \cdots & \phi_m \end{bmatrix} \tag{9}$$

is overcomplete and no longer has an inverse. There are two alternative ways in which we can use a given overcomplete dictionary to define a sparse linear model:

1. **Synthesis:** The synthesis sparse model assumes that there is a sparse m-dimensional vector of coefficients c such that

   $$x = Dc. \tag{10}$$

   As we will see, finding such a c from a given x is not necessarily an easy task.

2. **Analysis:** In the analysis sparse model assumes that

$$D^T x \tag{11}$$

is sparse, i.e. that the signal has nonzero correlation with a small number of atoms in the dictionary.

If the dictionary is an orthonormal basis, both models are equivalent.

2 Linear transforms

In this section we describe some of the most important transforms in signal processing.

2.1 Frequency representation

The frequency decomposition or spectrum of a function is obtained by representing the function as a superposition of sinusoids. To make this more formal, let us consider an infinite dictionary of sinusoids

$$\phi_k := \left\{ e^{2\pi k t} = \cos(2\pi k t) + i \sin(2\pi k t),\ k \in \mathbb{Z} \right\}. \tag{12}$$

Recall that $\mathbb{L}_2([0,1])$, the space of square-integrable functions defined on the unit interval, is a Hilbert space when endowed with the inner product

$$\langle f, g \rangle = \int_0^1 \overline{f(t)} g(t)\, \mathrm{d}\, t. \tag{13}$$

It is not difficult to check that the dictionary of sinusoidal is orthonormal under this inner product: the atoms are mutually orthogonal and have unit norm.

The Fourier series coefficients of a signal $f \in \mathbb{L}_2$ are obtained by taking its inner product with atoms from the dictionary,

$$c_k := \langle \phi_k, f \rangle \tag{14}$$

$$= \int_0^1 f(t)\, e^{-i2\pi k t}\, \mathrm{d}\, t. \tag{15}$$

The Fourier series of order n of f is defined as

$$S_n(t) := \sum_{k=-n}^{n} c_k e^{i2\pi k t} = \sum_{k=-n}^{n} \langle \phi_k, f \rangle \phi_k. \tag{16}$$

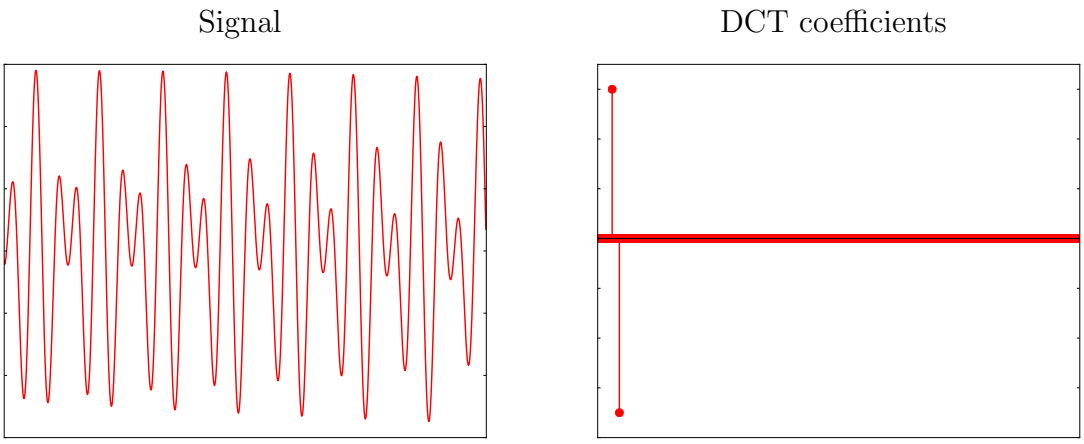

Figure 1: A signal that is sparse in the DCT dictionary.

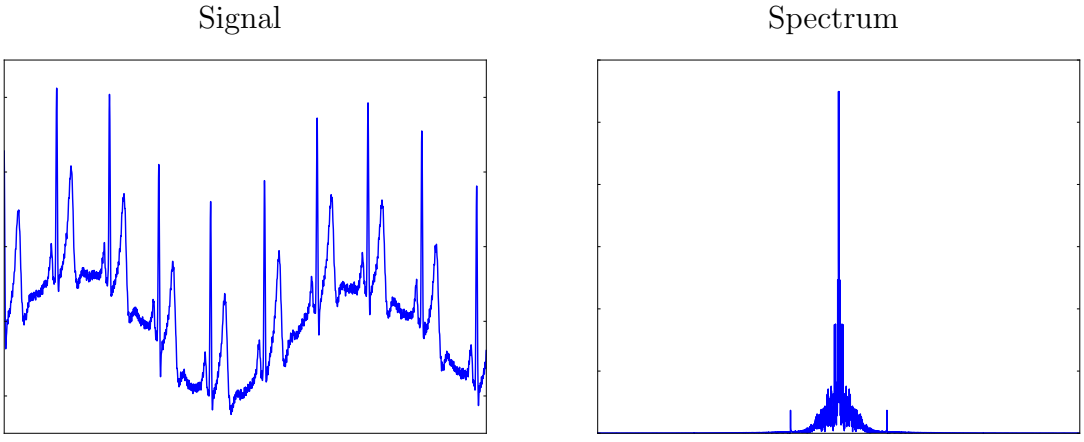

Figure 2: Electrocardiogram signal (left) and the magnitude of its spectrum (right).

It turns out that the dictionary is actually an orthonormal basis for $\mathbb{L}_2$, since

$$\lim_{n\to\infty} ||f(t) - S_n(t)|| = 0 \qquad \text{for all } f \in \mathbb{L}_2 \ . \tag{17}$$

This is a classical result, we refer to any text on Fourier analysis for the proof.

The discrete Fourier transform (DFT) is a discrete counterpart to the Fourier series. It maps vectors in $\mathbb{C}^n$ to their decomposition in terms of discrete sinusoids of the form

$$\phi_k = \frac{1}{\sqrt{n}} \begin{bmatrix} 1 \\ e^{\frac{i2\pi k}{n}} \\ e^{\frac{i2\pi k2}{n}} \\ \cdots \\ e^{\frac{i2\pi k(n-1)}{n}} \end{bmatrix}, \quad 0 \leq k \leq n-1. \tag{18}$$

This dictionary is an orthonormal basis of C^n. The corresponding matrix is known as the DFT matrix,

$$F := \begin{bmatrix} \phi_0 & \phi_1 & \cdots & \phi_{n-1} \end{bmatrix}, \tag{19}$$

$$\text{DFT}\{x\} := Fx. \tag{20}$$

The fast Fourier transform (FFT) algorithm allows to compute the DFT in $\mathcal{O}(n \log n)$. This efficiency is crucial in practical applications. The discrete cosine transform (DCT) is a related transformation designed for real vectors; the corresponding atoms are shifted cosines instead of complex exponentials. Figure 1 shows a signal that is sparse in the DCT dictionary. Figure 2 shows the frequency representation of an electrocardiogram signal. The energy of its spectrum is highly concentrated in the low frequencies, as the signal fluctuates slowly. The representation allows to detect periodicities in the signal, which correspond to DFT coefficients with large magnitudes.

The DFT can be extended to two dimensions by considering two-dimensional sinusoidal atoms obtained by taking the outer product of the one-dimensional atoms defined by (18).

$$\phi^{\text{2D}}_{k_1,k_2} := \frac{1}{n} \begin{bmatrix} 1 & e^{\frac{i2\pi k_2}{n}} & \cdots & e^{\frac{i2\pi k_2(n-1)}{n}} \\ e^{\frac{i2\pi k_1}{n}} & e^{\frac{i2\pi (k_1+k_2)}{n}} & \cdots & e^{\frac{i2\pi (k_1+k_2(n-1))}{n}} \\ & & \cdots & \\ e^{\frac{i2\pi k_1(n-1)}{n}} & e^{\frac{i2\pi (k_1(n-1)+k_2)}{n}} & \cdots & e^{\frac{i2\pi (k_1(n-1)+k_2(n-1))}{n}} \end{bmatrix} \tag{21}$$

$$= \phi^{\text{1D}}_{k_1} \left(\phi^{\text{1D}}_{k_2}\right)^T. \tag{22}$$

To compute the 2D DFT of an image we take inner products with these 2D sinusoidal atoms, which form an orthonormal basis of $\mathbb{C}^{n\times n}$. The transform can be computed efficiently by applying 1D DFTs to the rows and columns of the array,

$$\text{DFT}^{\text{2D}}\{X\} := FXF \tag{23}$$

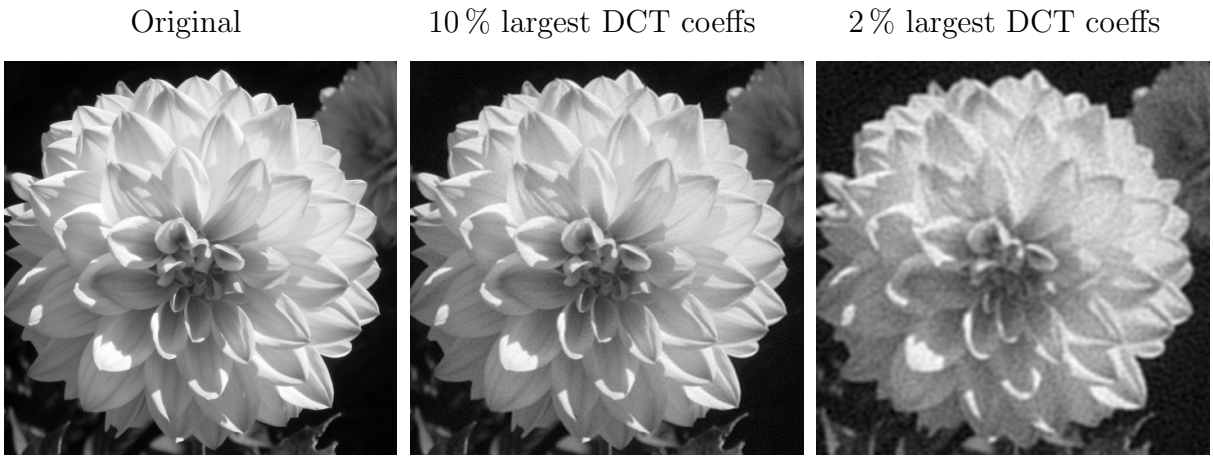

Figure 3: Discarding the smallest DCT coefficients of an image allows to compress it quite effectively.

The 2D frequency representation of images tends to be sparse. In particular, most of the energy of the image tends to be concentrated in the lower frequencies. This insight can be used to compress the image by just retaining the coefficients with larger magnitudes. Figure 3 shows that this simple scheme can be quite effective. The JPEG compression standard is based on a similar idea: high-frequency coefficients are discarded according to a perceptual model.

2.2 Short-time Fourier transform (STFT)

The Fourier series or the DFT provide global information about the periodicities of a signal, but they do not capture localized periodicities. However, the spectrum of speech, music and other sound signals changes with time. To analyze the changes in the spectrum we can compute the DFT of time segments of the signal. However, doing this in a naïve way may introduce spurious high-frequencies, as shown in Figure 4. Multiplying the time segment with a window that tapers off at the ends smoothens the transitions and avoids introducing high-frequency artifacts.

The short-time Fourier transform (STFT) is defined as the Fourier series coefficients of the pointwise product between a function and a shifted window $w : [0, 1] \to \mathbb{C}$ approximately localized in time and frequency

$$\text{STFT}\left\{f\right\}(k, \tau) := \int_0^1 f(t)\, \overline{w(t-\tau)} e^{-i2\pi kt}\, \mathrm{d}\, t. \tag{24}$$

Equivalently, the STFT coefficients are equal to the inner product between the signal and atoms of the form $\phi_{k,\tau}(t) := w(t-\tau)\, e^{i2\pi kt}$, which corresponds to copies of w shifted by τ

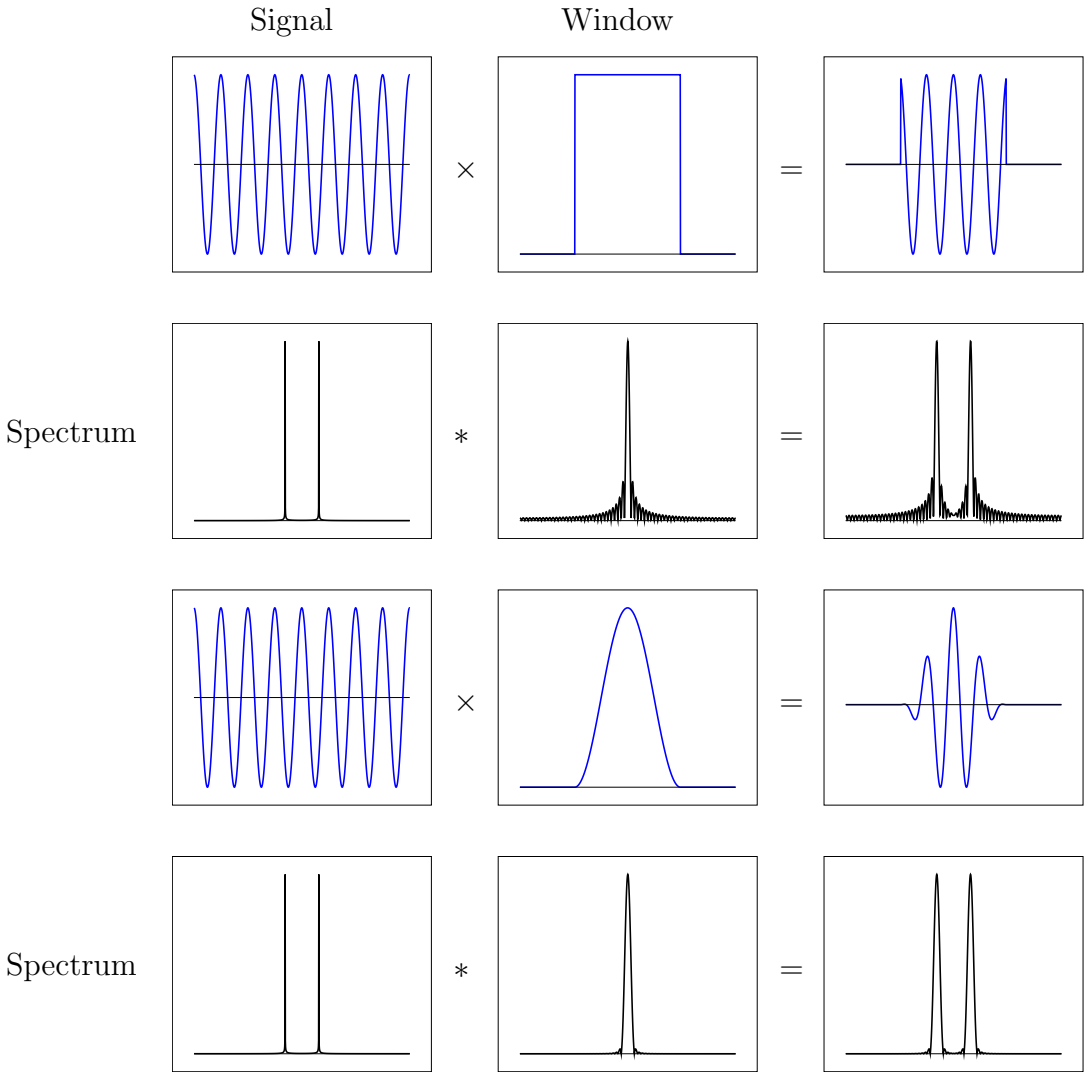

Figure 4: The spectrum of a time segment may contain spurious high-frequency content produced by the sudden transition at the ends of the segment. In the frequency domain, the spectrum is being convolved by a sinc function, which has a very heavy tail. Multiplying the signal by a localized window that has a faster decay in the frequency domain alleviates the problem.

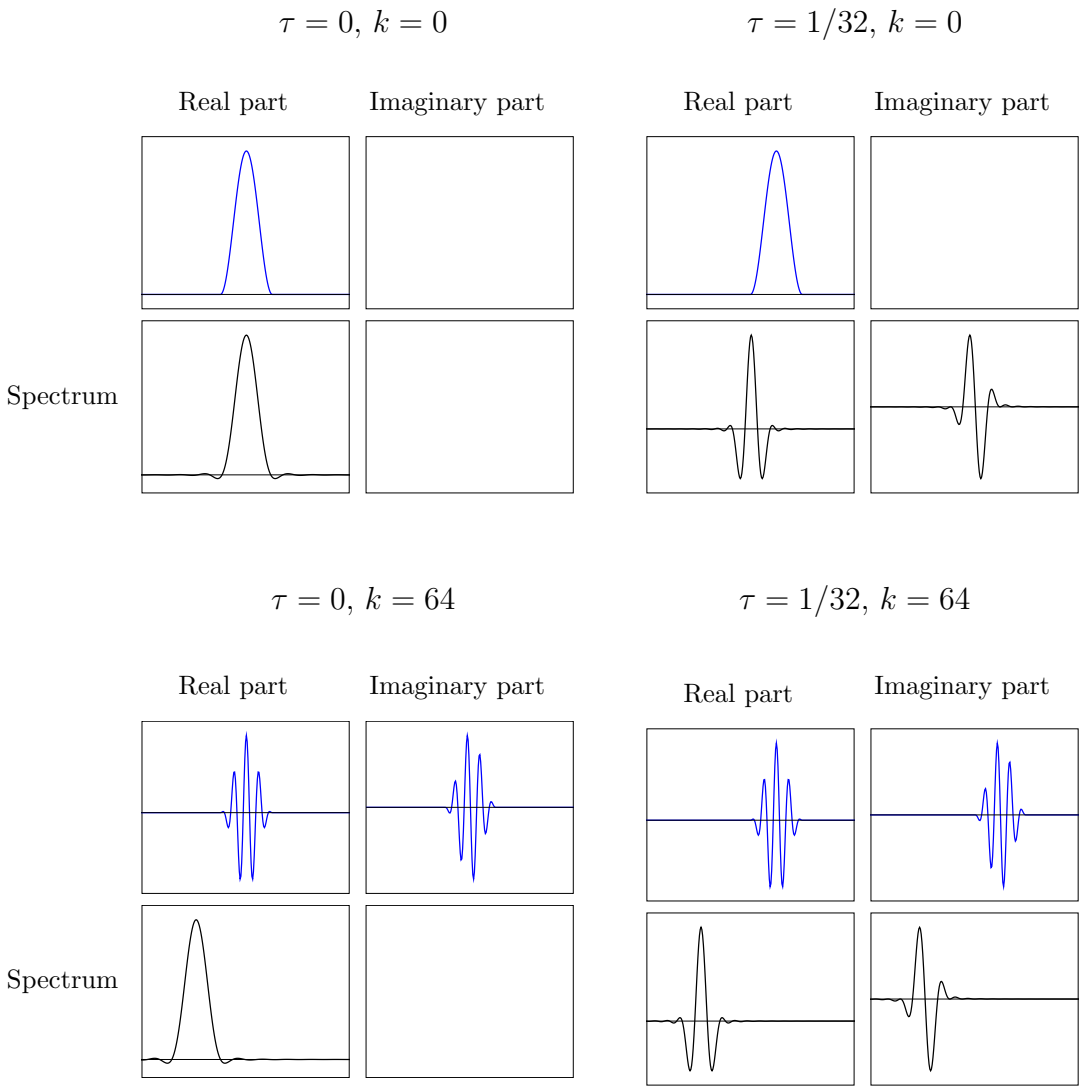

Figure 5: Atoms in the STFT dictionary.

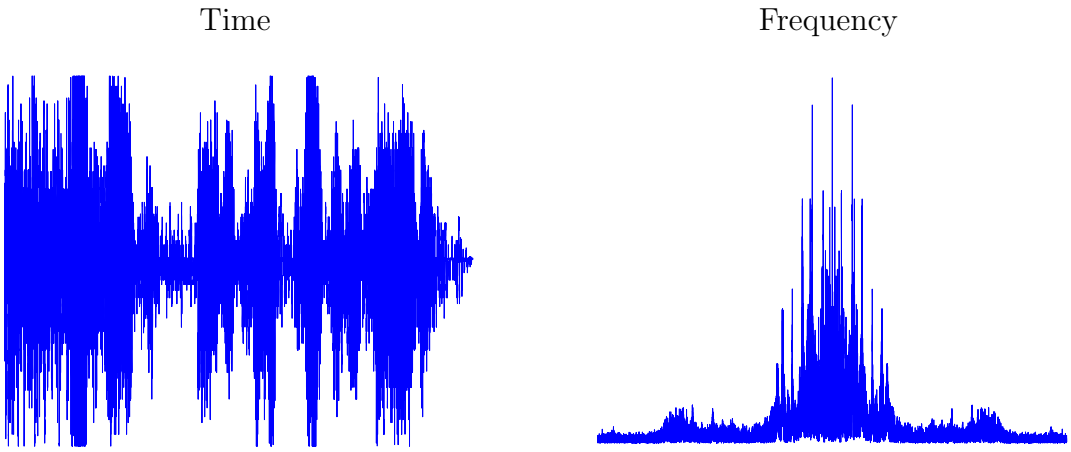

Figure 6: Time and frequency representation of a speech signal.

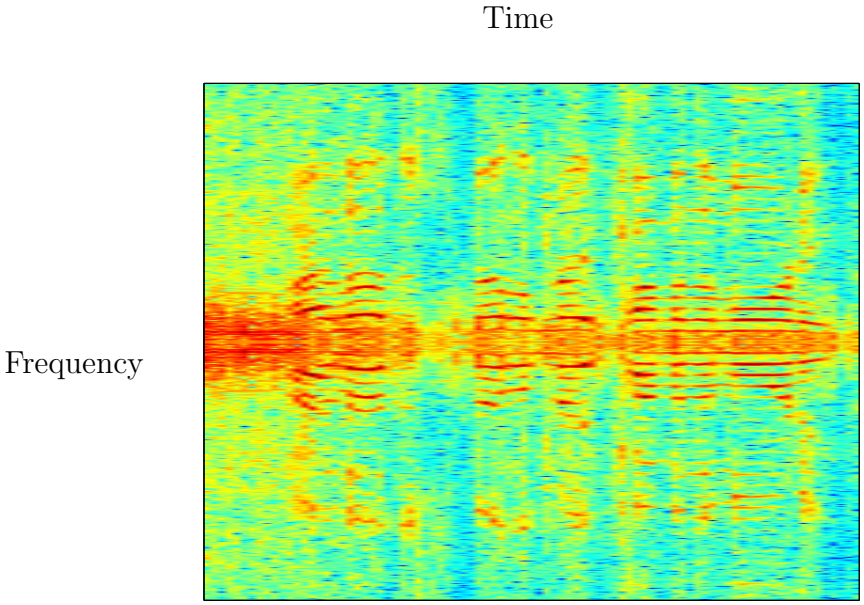

Figure 7: Spectrogram (log magnitude of STFT coefficients) of the speech signal in Figure 6.

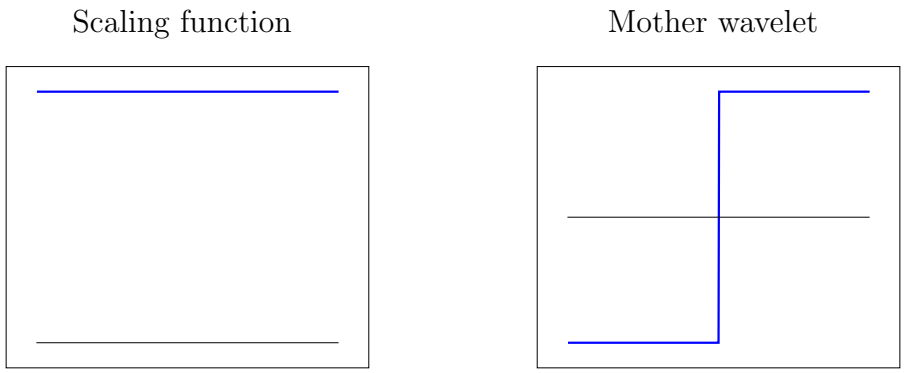

Figure 8: Scaling function and mother wavelet of the Haar wavelet transform.

in time and by k in frequency. Some examples are shown in Figure 5. Including dilations of w (in addition to time and frequency translations) yields a dictionary of Gabor atoms.

The discrete-time STFT consists of pointwise multiplication by a shifted window followed by a DFT. This is equivalent to computing $D^T x$, where $D \in \mathbb{C}^{n \times m}$, $m > n$ is an overcomplete dictionary. The corresponding analysis sparse model is very useful for speech analysis. The logarithm of the magnitude of the STFT coefficients, called an spectrogram, is widely used for sound processing. Figure 7 shows the spectrogram of a real speech signal. The time and frequency representation of the same signal are shown in Figure 6.

2.3 Wavelets

Wavelets are atoms that allow to capture signal structure at different scales. A wavelet ψ is a unit-norm, zero-mean function in $\mathbb{L}_2$. The wavelet transform of a function $f \in \mathbb{L}_2$ is defined as

$$\mathrm{W}\left\{f\right\}(s,\tau) := \frac{1}{\sqrt{s}} \int f(t)\, \overline{\psi\left(\frac{t-\tau}{s}\right)}\, \mathrm{d}\, t = \langle \phi_{s,\tau}, f \rangle \,. \tag{25}$$

The wavelet coefficients are obtained by taking the inner product with atoms that are shifted and dilated copies of the *mother* wavelet ψ

$$\phi_{s,\tau}(t) = \frac{1}{\sqrt{s}} \psi\left(\frac{t-\tau}{s}\right). \tag{26}$$

The wavelet transform coefficients provide information about the signal at different scales. This rather vague statement can be made precise using the multiresolution framework of Mallat and Meyer.

Definition 2.1 (Multiresolution approximation)**.** *A multi resolution approximation is a sequence* $\{\mathcal{V}_j, j \in \mathbb{Z}\}$ *of closed subspaces of* $\mathbb{L}_2(\mathbb{R})$ *satisfying the following conditions.*

- *Dilating functions in* $\mathcal{V}_j$ *by 2 yields functions in* $\mathcal{V}_{j+1}$

$$f(t) \in \mathcal{V}_j \iff f\left(\frac{t}{2}\right) \in \mathcal{V}_{j+1}. \tag{27}$$

- *Approximations at a scale* 2^j *are always better than at* 2^{j+1}

$$\mathcal{V}_{j+1} \subset \mathcal{V}_j. \tag{28}$$

- $\mathcal{V}_j$ *is invariant to translations at the scale* 2^j

$$f(t) \in \mathcal{V}_j \iff f\left(t - 2^j k\right) \in \mathcal{V}_j \qquad \text{for all } k \in \mathbb{Z}. \tag{29}$$

- *As* $j \to \infty$ *the approximation loses all information*

$$\lim_{j \to \infty} \mathcal{V}_j = \{0\}. \tag{30}$$

- *As* $j \to -\infty$ *the approximation is perfect*

$$\lim_{j \to -\infty} \mathcal{V}_j = \mathbb{L}_2. \tag{31}$$

- *There exists a scaling function* $\zeta \in \mathcal{V}_0$ *such that*

$$\{\zeta_{0,k}(t) := \zeta(t-k),\ k \in \mathbb{Z}\} \tag{32}$$

  is an orthonormal basis for $\mathcal{V}_0$*.*

Under these conditions, we can interpret the projection $\mathcal{P}_{\mathcal{V}_j}(f)$ of a function f onto $\mathcal{V}_j$ as an approximation of f at scale 2^j. In a remarkable result, Mallat and Meyer prove that for any multiresolution approximation there exists a wavelet ψ such that

$$\mathcal{P}_{\mathcal{V}_j}(f) = \mathcal{P}_{\mathcal{V}_{j+1}}(f) + \sum_{k \in \mathbb{Z}} \left\langle \psi_{2^j,k}, f \right\rangle \psi_{2^j,k}. \tag{33}$$

The particular wavelet ψ that yields this orthonormal basis depends on the scaling function. Figure 8 shows the scaling function and the corresponding mother wavelet for the Haar wavelet transform. The dictionary of shifted wavelets dilated by 2^j $\left\{\psi_{2^j,k},\, k \in \mathbb{Z}\right\}$ is an orthonormal basis for $\mathcal{V}_j \cap \mathcal{V}_{j+1}^{\perp}$, i.e. the subspace in $\mathcal{V}_j$ that contains the functions that

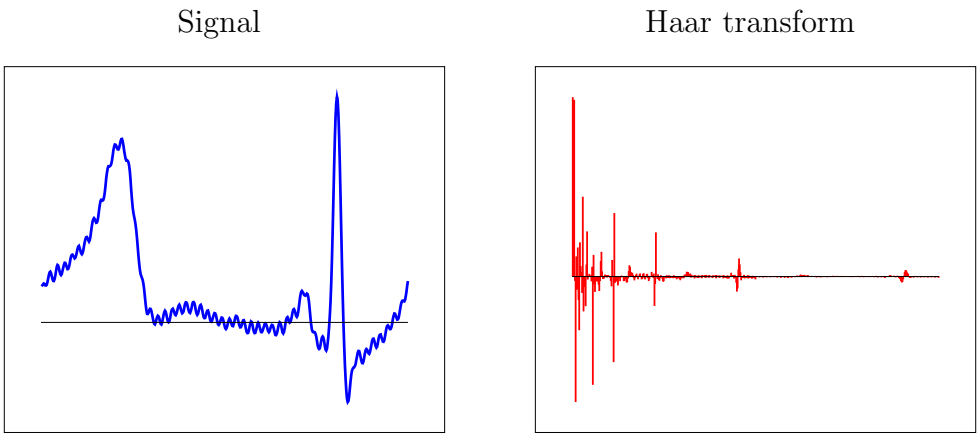

Figure 9: Haar wavelet coefficients (right) of an electrocardiogram signal (left).

are not available at coarser scales. As a result, $\left\{\zeta_{0,k}\left(t\right), \psi_{2^1,k}, \psi_{2^2,k}, \ldots, \psi_{2^j,k},\, k \in \mathbb{Z}\right\}$ is an orthonormal basis for $\mathcal{V}_j$.

Wavelet bases can be discretized to obtain orthonormal bases of $\mathbb{C}^n$ or $\mathbb{R}^n$. These discrete transforms can be computed in $\mathcal{O}\left(n\right)$. Figure 9 shows the Haar wavelet coefficients of an electrocardiogram signal. The corresponding multiresolution approximations at the different scales are shown in Figures 10, 11 and 12.

A signal-processing interpretation of the wavelet transform is that the scaling function acts as a low-pass filter, whereas the dilated and shifted wavelets act as band-pass filters in different bands. Many other wavelet bases apart from the Haar exist: Meyer, Daubechies, Battle-Lemarie, ... We refer the interested reader to [3] for more information. Chapter 7 provides a detailed and rigorous description of the construction of orthonormal wavelet bases from a multiresolution approximation.

Two-dimensional wavelets can be obtained by taking outer products of one-dimensional wavelets, as we did for the DFT.

$$\phi^{\text{2D}}_{s_1,s_2,k_1,k_2} := \phi^{\text{1D}}_{s_1,k_1} \left(\phi^{\text{1D}}_{s_2,k_2}\right)^T \tag{34}$$

The corresponding two-dimensional transform allows to obtain sparse representations of natural images. An example is shown in Figure 13. The wavelet coefficients at finer scales are mostly zero in most areas of the image. Figure 14 shows the sorted magnitudes of the coefficients on a logarithmic scale. A large majority of the coefficients are very small and can be discarded without significantly affecting the quality of the image. The JPEG 2000 compression standard is based this insight.

Finally we would like to mention that designing multidimensional transforms that are more effective at providing sparse representations for images has been a vibrant research subject

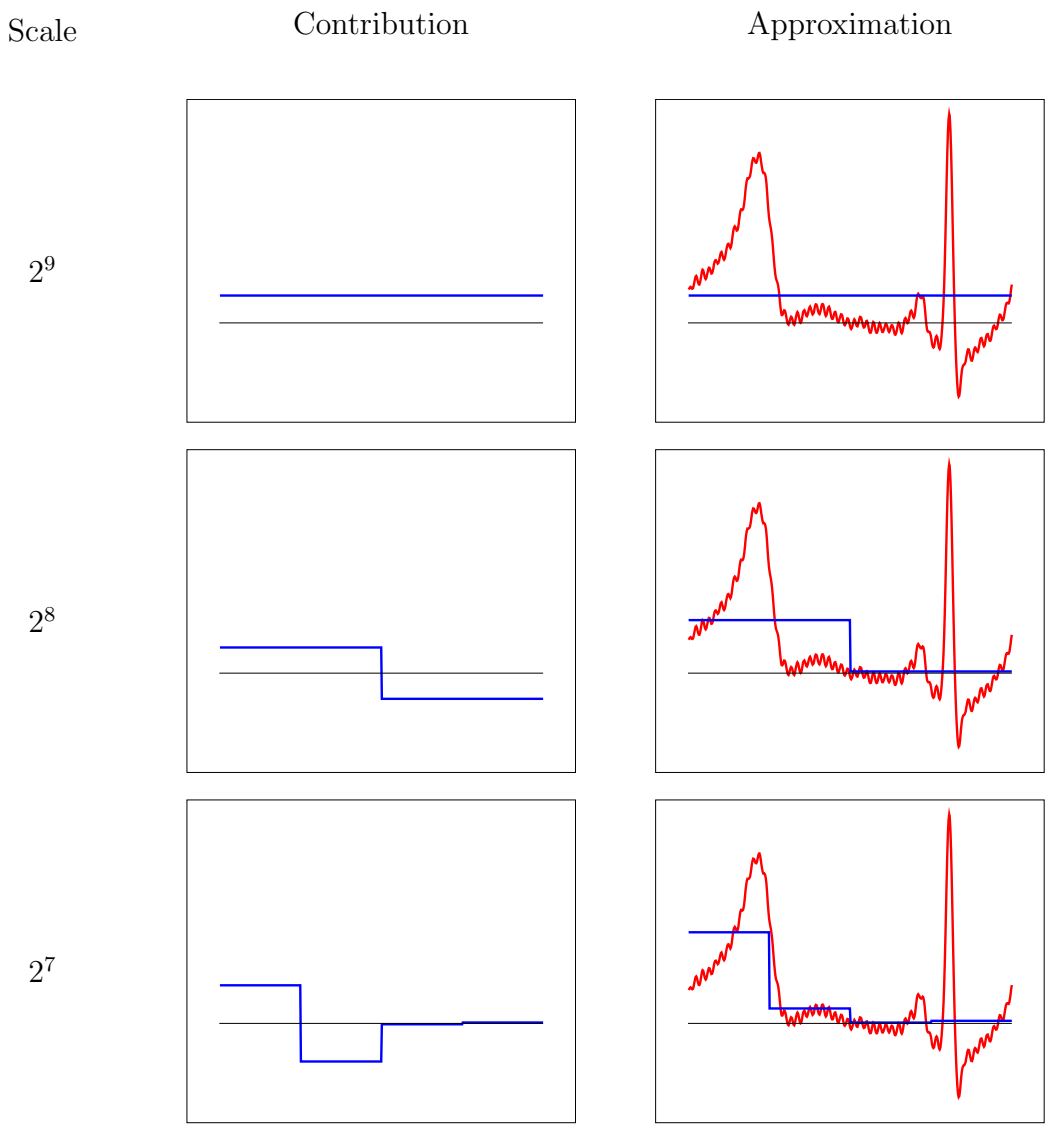

Figure 10: Approximation of the electrocardiogram signal in Figure 9 using a Haar multiresolution approximation at different scales (right). On the left, we can see the projection of the signal onto $\mathcal{V}_j \cap \mathcal{V}_{j+1}^{\perp}$, which captures the information in the signal at scale 2^j.

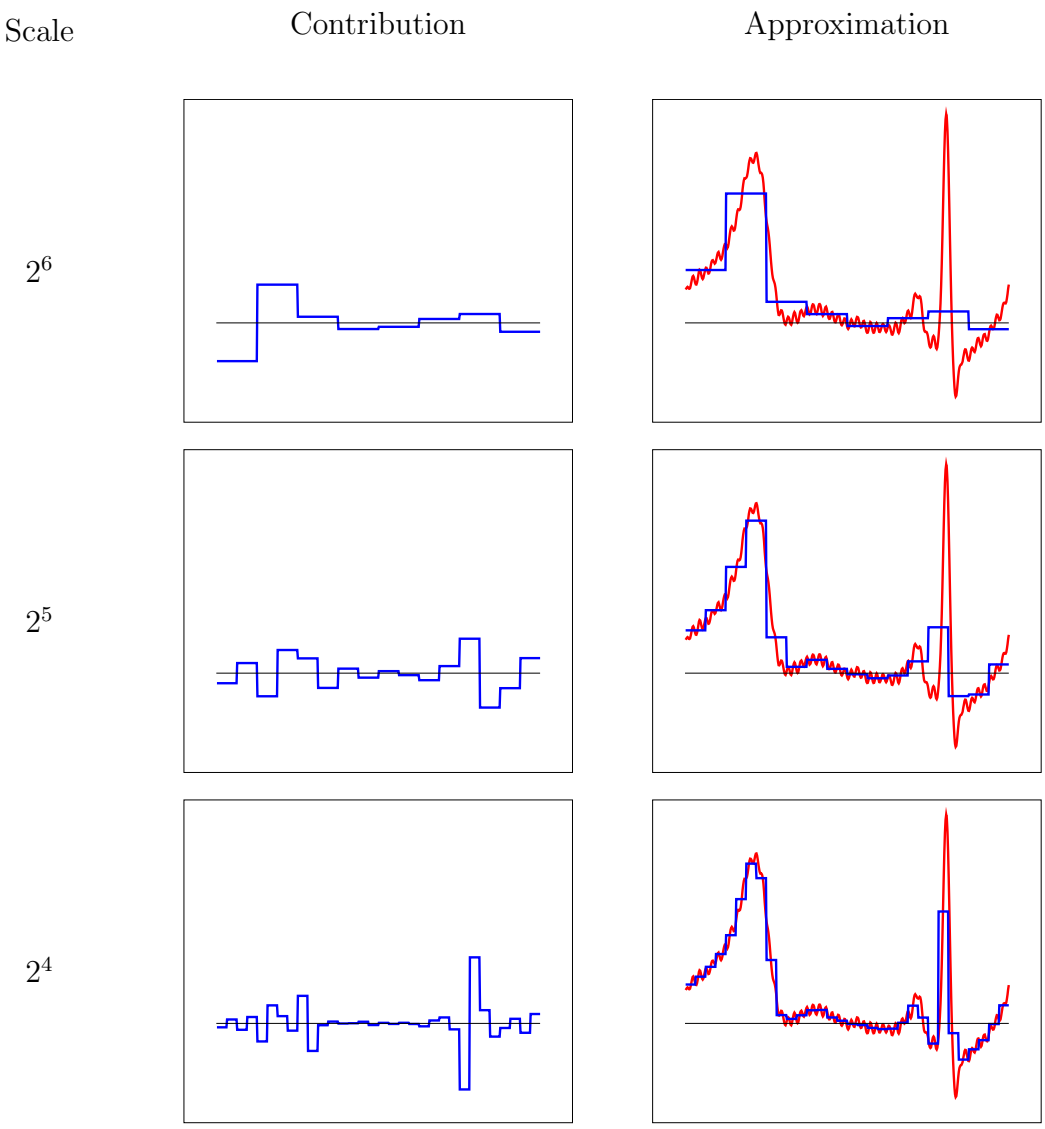

Figure 11: Approximation of the electrocardiogram signal in Figure 9 using a Haar multiresolution approximation at different scales (right). On the left, we can see the projection of the signal onto $\mathcal{V}_j \cap \mathcal{V}_{j+1}^{\perp}$, which captures the information in the signal at scale 2^j.

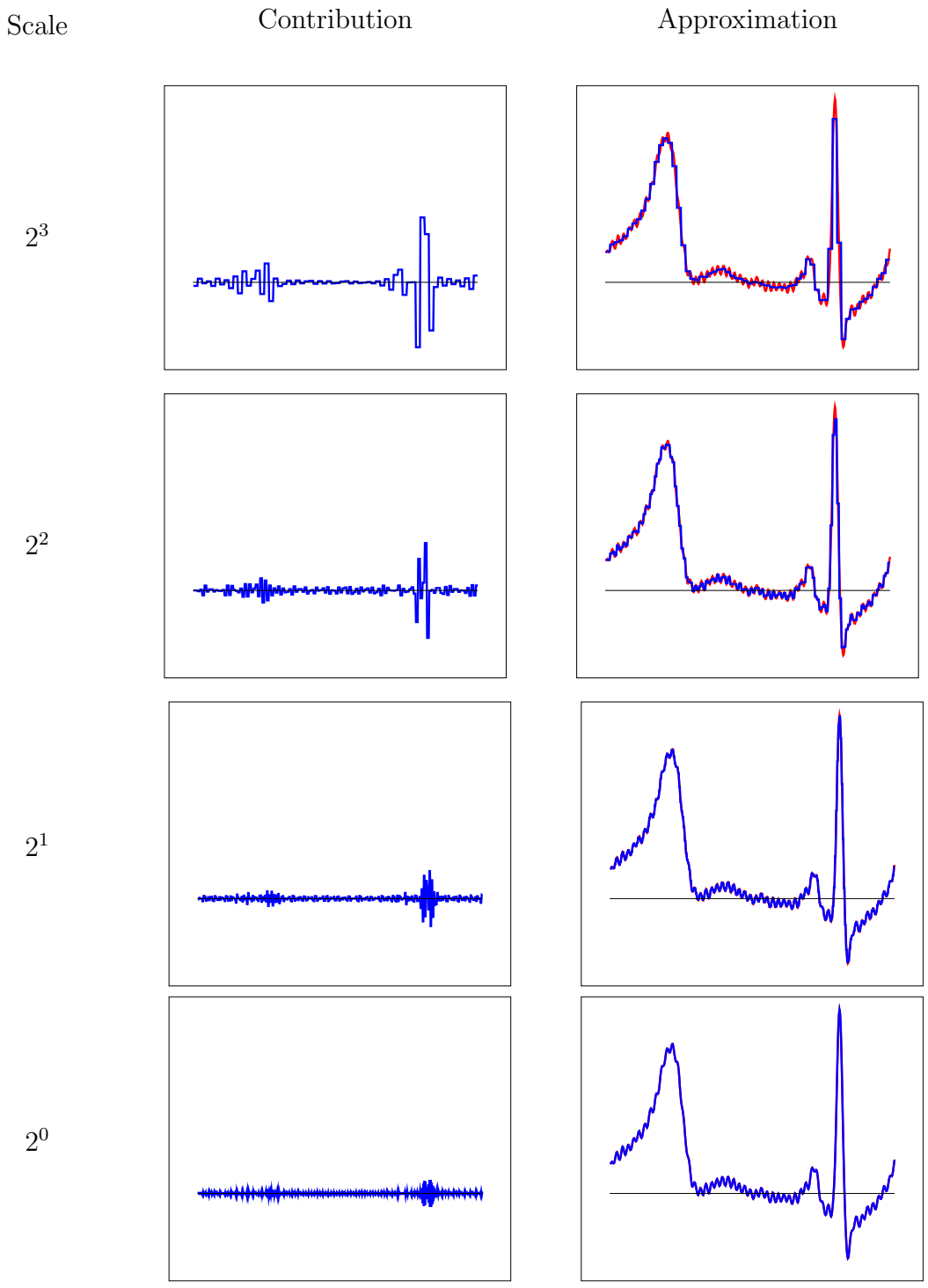

Figure 12: Approximation of the electrocardiogram signal in Figure 9 using a Haar multiresolution approximation at different scales (right). On the left, we can see the projection of the signal onto $\mathcal{V}_j \cap \mathcal{V}_{j+1}^{\perp}$, which captures the information in the signal at scale 2^j.

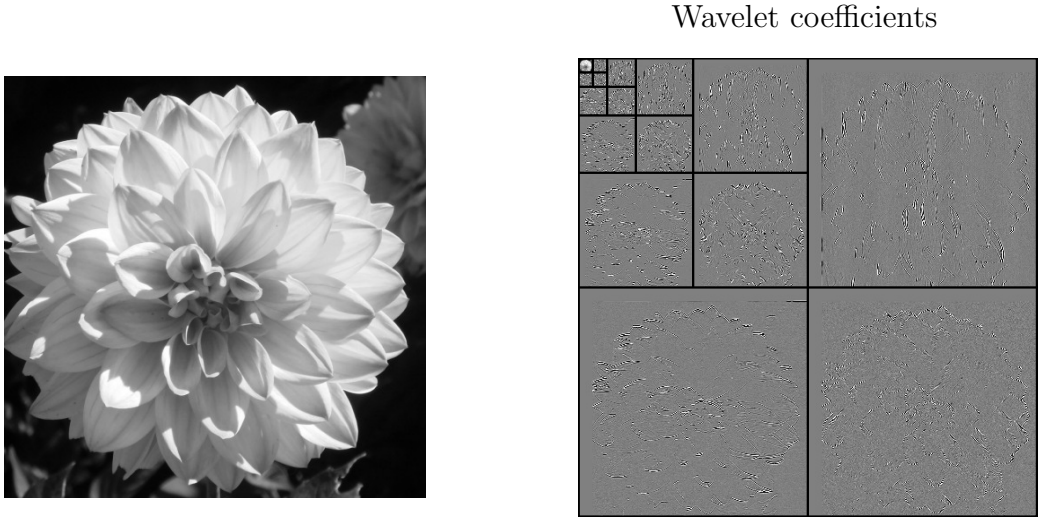

Figure 13: Coefficients in a wavelet basis (right) of a natural image (left).

for many years. Some of these extensions include the steerable pyramid, ridgelets, curvelets, and bandlets. We refer to Section 9.3 in [3] for more details.

3 The synthesis model

Overcomplete dictionaries provide greater flexibility for signal representation than orthonormal transforms, but this comes at a cost. Computing the corresponding coefficients and using the corresponding sparse models to process data is not as simple or computationally efficient. We consider an overcomplete dictionary D with linearly independent atoms $\{\phi_1, \ldots, \phi_m \in \mathbb{R}^m\}$ that are linearly independent

$$D := \begin{bmatrix} \phi_1 & \phi_2 & \cdots & \phi_m \end{bmatrix}, \qquad m > n. \tag{35}$$

The synthesis sparse model assumes that a signal $x \in \mathbb{R}^n$ can be represented as

$$x = Dc, \tag{36}$$

where c is sparse. Unfortunately, even if such a sparse c exists, it is not easy to compute it from x. There reason is that there are infinite choices of coefficient vectors c' such that $x = Dc'$ and most of them are not sparse at all!

A possible choice for the coefficient vector is given by the following lemma, which is proved in Section A.1 of the appendix.

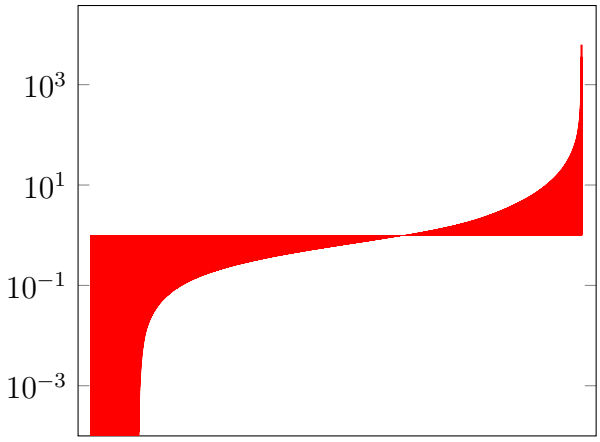

Figure 14: Sorted magnitude of the wavelet coefficients shown in Figure 13 plotted on a logarithmic scale. The image is highly compressible in the wavelet domain.

Lemma 3.1 (Minimum ℓ_2-norm solution)**.** *The coefficient vector with minimum ℓ_2 norm satisfying $x = Dc$ is*

$$c_{\ell_2} := D^T \left(D D^T \right)^{-1} x, \tag{37}$$

which is the projection of c onto the row space of D.

In the case of complex signals, the same result holds replacing D^T by D^*.

Unfortunately, the minimum ℓ_2-norm solution is often very dense. Let us consider a concrete example. We define the atoms of an overcomplete dictionary of sinusoids as

$$\phi_k(j) := \frac{1}{\sqrt{n}} e^{\frac{i 2\pi k j}{m}}, \quad 1 \leq k \leq m, \, 1 \leq j \leq n. \tag{38}$$

If $m = n$ this is an orthonormal DFT basis, but for $m > n$ the sinusoidal atoms are correlated and the dictionary is overcomplete. Figure 15 shows a signal that is sparse in this dictionary and the corresponding coefficient vector. Note that this signal would *not* be sparse in a DFT basis. As we can see in Figure 16, the minimum ℓ_2-norm coefficient vector is not sparse.

Ideally, we would like to find the coefficient vector with the smallest number of nonzeros that corresponds to the signal. More precisely, our aim is to solve

$$\text{minimize} \quad ||\tilde{c}||_0 \tag{39}$$

$$\text{subject to} \quad x = D\,\tilde{c} \tag{40}$$

Unfortunately, this optimization problem is computationally intractable. Intuitively, we cannot do much better than try out all the possibilities. However there are tractable methods that are able to produce sparse coefficients in practice. They are divided in two main classes.

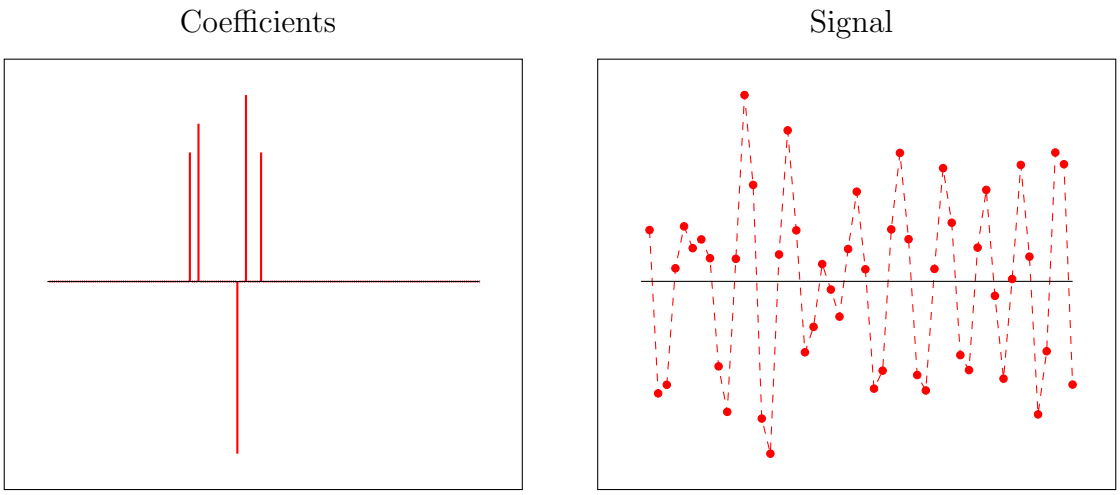

Figure 15: A signal that is sparse in an overcomplete dictionary of sinusoids (left) and its corresponding coefficients (right).

- Greedy methods which select atoms one by one.
- Methods based on solving a related convex program, usually ℓ_1-norm minimization.

In general, the performance of these methods deteriorates for dictionaries with more correlated atoms. This is not surprising because correlations *tangle* the contributions of the different atoms making more difficult to obtain a sparse solution efficiently.

3.1 Greedy methods

Matching pursuit (MP) [4] is a very simple method for obtaining a sparse coefficient vector. We initialize a residual vector to equal the signal. Then we iteratively choose the atom that is most correlated with the residual and subtract the component of the residual in that direction.

Algorithm 3.2 (Matching pursuit)**.** *Given a dictionary $D \in \mathbb{R}^{n\times m}$ (or $\mathbb{C}^{n\times m}$) and a signal $x \in \mathbb{R}^n$ (or $\mathbb{C}^n$), we initialize the residual and the approximation by setting,*

$$r^{(0)} := x, \tag{41}$$

$$\hat{x}^{(0)} := 0, \tag{42}$$

$$\mathcal{I}^{(0)} = \emptyset. \tag{43}$$

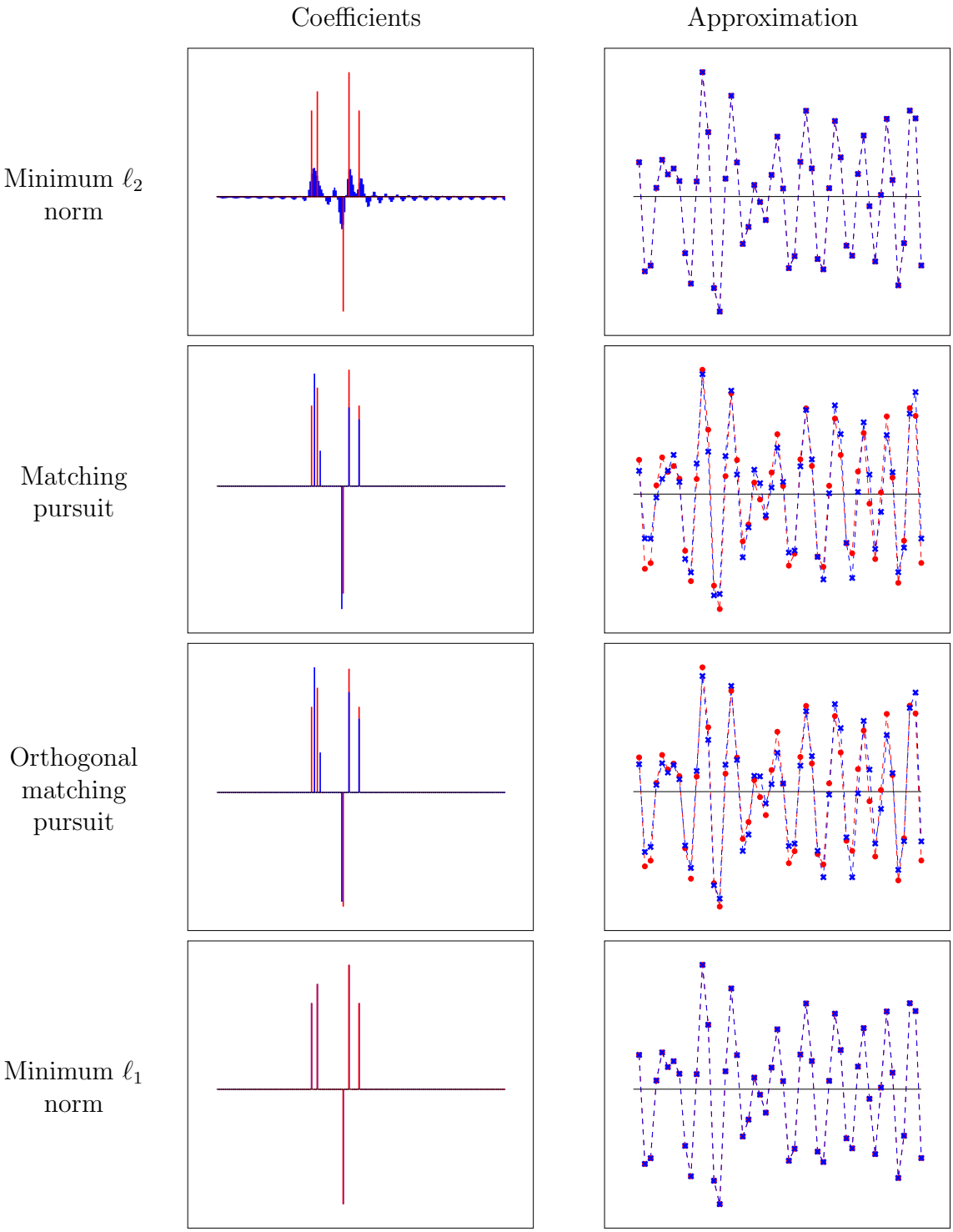
Coefficients
Approximation
Minimum ℓ_2 norm
Matching pursuit
Orthogonal matching pursuit
Minimum ℓ_1 norm

Figure 16: .

Then for a fixed number of iterations $k = 1, 2, \ldots, N$ we compute

$$\phi^{(k)} := \arg \max_{j \in \{1,2,\ldots,m\}/\mathcal{I}^{(k-1)}} \left| \left\langle r^{(k-1)}, \phi_j \right\rangle \right|, \tag{44}$$

$$\hat{x}^{(k)} := \hat{x}^{(k-1)} + \left\langle r^{(k-1)}, \phi^{(k)} \right\rangle \phi^{(k)}, \tag{45}$$

$$r^{(k)} := r^{(k-1)} - \left\langle r^{(k-1)}, \phi^{(k)} \right\rangle \phi^{(k)}, \tag{46}$$

$$\mathcal{I}^{(k)} := \mathcal{I}^{(k-1)} \cup \{j\} \quad \textit{where } j \textit{ is the index such that } \phi^{(k)} = \phi_j. \tag{47}$$

The output is a sparse approximation to the signal $\hat{x}^{(N)}$ and the corresponding coefficient indices $\mathcal{I}^{(N)}$.

If the atoms form an orthonormal basis, MP produces the exact coefficient vector in a number of iterations that is equal to its cardinality. However, for more correlated dictionaries it might choose wrong atoms (that may be very correlated with the actual atoms that form the signal), as shown in Figure 16.

The approximation obtained by MP at every iteration is not necessarily optimal in terms of ℓ_2-norm. Indeed, given a set of atoms

$$D_{\mathcal{I}^{(N)}} := \begin{bmatrix} \phi^{(1)} & \phi^{(2)} & \cdots & \phi^{(N)} \end{bmatrix} \tag{48}$$

where $N < m$, the coefficients that yield the best ℓ_2-norm approximation are equal to the least-squares estimate

$$c_{\text{ls}} := D^{\dagger}_{\mathcal{I}^{(k)}} x \tag{49}$$

$$= \left(D^T_{\mathcal{I}^{(N)}} D_{\mathcal{I}^{(N)}} \right)^{-1} D^T_{\mathcal{I}^{(N)}} x. \tag{50}$$

In contrast, MP computes

$$\left(c_{\text{MP}} \right)_k = \left\langle \phi^{(k)}, x - \sum_{j=1}^{k-1} \left(c_{\text{MP}} \right)_j \phi^{(j)} \right\rangle. \tag{51}$$

Setting the coefficients to equal (49) ensures that the residual and the approximation are orthogonal. This is the strategy followed by orthogonal matching pursuit (OMP) [5].

Algorithm 3.3 (Orthogonal matching pursuit)**.** *Given a dictionary $D \in \mathbb{R}^{n \times m}$ (or $\mathbb{C}^{n \times m}$) and a signal $x \in \mathbb{R}^n$ (or $\mathbb{C}^n$), we initialize the residual and the approximation by setting,*

$$r^{(0)} := x, \tag{52}$$

$$\mathcal{I}^{(0)} = \emptyset. \tag{53}$$

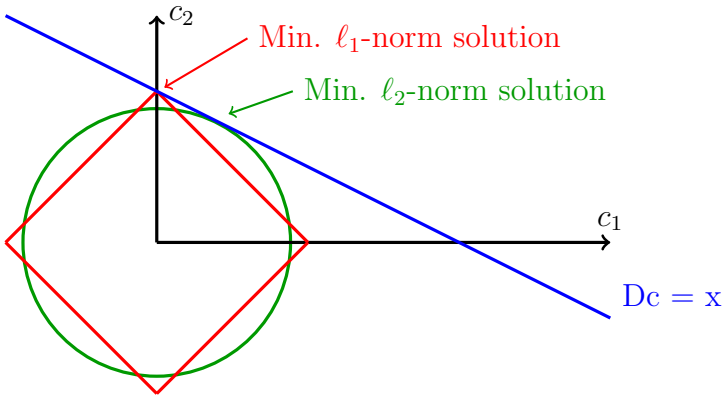

Figure 17: The minimum ℓ_1-norm solution is sparser than the minimum ℓ_2-norm solution because of the geometry of the ℓ_1-norm and ℓ_2-norm balls.

Then for a fixed number of iterations $k = 1, 2, \ldots, N$ *we compute*

$$\phi^{(k)} := \arg \max_{j \in \{1,2,\ldots,m\}/\mathcal{I}^{(k-1)}} \left| \left\langle r^{(k-1)}, \phi_j \right\rangle \right|, \tag{54}$$

$$\mathcal{I}^{(k)} := \mathcal{I}^{(k-1)} \cup \{j\} \quad \textit{where } j \textit{ is the index such that } \phi^{(k)} = \phi_j, \tag{55}$$

$$D_{\mathcal{I}^{(k)}} := \left[\phi^{(1)} \quad \phi^{(2)} \quad \cdots \quad \phi^{(k)} \right], \tag{56}$$

$$\hat{c}^{(k)} := D^{\dagger}_{\mathcal{I}^{(k)}} x, \tag{57}$$

$$\hat{x}^{(k)} := D_{\mathcal{I}^{(k)}} \hat{c}^{(k)}, \tag{58}$$

$$r^{(k)} := x - \hat{x}^{(k)}. \tag{59}$$

The output is a sparse approximation to the signal $\hat{x}^{(N)}$ *and the corresponding coefficient indices* $\mathcal{I}^{(N)}$.

OMP produces better approximations to the signal than MP, but it may still choose the wrong atoms if the dictionary is highly correlated, as is the case in Figure 16.

3.2 ℓ_1-norm minimization

As shown in Figure 16 minimizing the ℓ_2 norm of the coefficient vector does not tend to yield a sparse solution. However, minimizing the ℓ_1 norm, i.e. solving the problem

$$\text{minimize} \quad ||\tilde{c}||_1 \tag{60}$$

$$\text{subject to} \quad x = D\,\tilde{c} \tag{61}$$

often does. This approach is known as basis pursuit [2] in the literature. As we have seen in previous lectures, the optimization problem is convex and can be recast as a linear program.

It is therefore computationally tractable although significantly more computationally costly than greedy methods such as MP or OMP. ℓ_1-norm minimization achieves better results than these methods in some cases, see for instance Figure 16, but it may also fail to find sparse solutions in highly correlated dictionaries.

Figure 17 provides some geometric intuition as to why minimizing the ℓ_1 norm is better than minimizing the ℓ_2 norm when we aim to obtain sparse solutions under linear constraints. The ℓ_1-norm ball is more concentrated around the axes than the ℓ_2-norm ball. It is therefore more likely for the line representing the constraint $x = Dc$ to be tangent to the ball on an axis, where the solution has cardinality one instead of two. As a result, the minimum ℓ_1-norm solution is sparser than the minimum ℓ_2-norm solution. This intuition generalizes to higher dimensions.

4 Denoising

4.1 The denoising problem

The aim of denoising is to extract a signal from data that is corrupted by uninformative perturbations, which we call noise. We will focus on the additive noise model

$$\text{data} = \text{signal} + \text{noise}, \tag{62}$$

$$y = x + z. \tag{63}$$

In order to achieve denoising it is necessary to have some prior knowledge about the structure of the signal and the structure of the noise. As an example, in Figure 18, an electrocardiogram recording is corrupted by high-frequency perturbations. In the frequency domain, we can see a small peak at 60 Hz due to noise coming from the power grid. To eliminate this noise we enforce the prior that the signal of interest is essentially low pass by filtering out the high end of the spectrum.

4.2 Thresholding

Dictionaries and transforms such as the DFT, STFT and the wavelet transform allow to obtain sparse representations of images, music, speech and other signals. These dictionaries exploit structure that is expected to be present in the signal but not in the noise. As a result the noise component in the data will not have a sparse representation most of the time; it is *incoherent* with the dictionary atoms. This can be exploited to denoise the signal via thresholding.

Definition 4.1 (Hard thresholding)**.** *Let $x \in \mathbb{R}^n$. The hard-thresholding operator $\mathcal{H}_\eta : \mathbb{R}^n \to \mathbb{R}^n$ sets to zero any entries in x with magnitude smaller than a predefined real-valued*

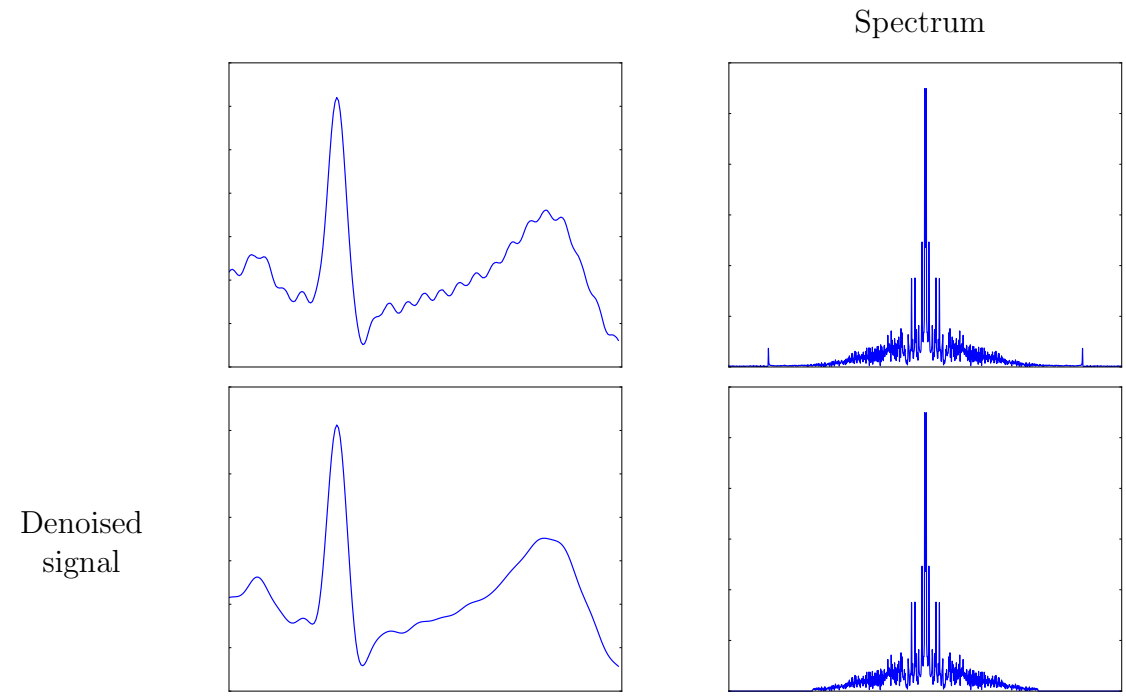

Figure 18: Electrocardiogram recording denoised via low-pass filtering.

threshold $\eta > 0$,

$$\mathcal{H}_\eta(x)_i := \begin{cases} x_i & \text{if } |x_i| > \eta, \\ 0 & \text{otherwise.} \end{cases} \tag{64}$$

The idea is very simple. We first map the signal to a domain where it is sparse, but the noise is dense. Then we discard all the coefficients below a certain threshold, as illustrated in Figure 19. In particular, if the signal is sparse in a basis, we threshold the coefficients $B^{-1}y$,

$$\hat{c} = \mathcal{H}_\eta\left(B^{-1}y\right) \tag{65}$$

$$= \mathcal{H}_\eta\left(c + B^{-1}z\right), \tag{66}$$

$$\hat{x} = B\hat{c}. \tag{67}$$

For the method to work $B^{-1}z$ must not be sparse. We can actually prove that this is the case if the basis is orthonormal and the noise is iid Gaussian.

Lemma 4.2 (Thresholding Gaussian noise)**.** *If z is an n-dimensional iid Gaussian with zero mean and variance σ^2, then for any orthogonal matrix $U \in \mathbb{R}^n$ $U^T z$ is iid Gaussian with zero mean and variance σ^2.*

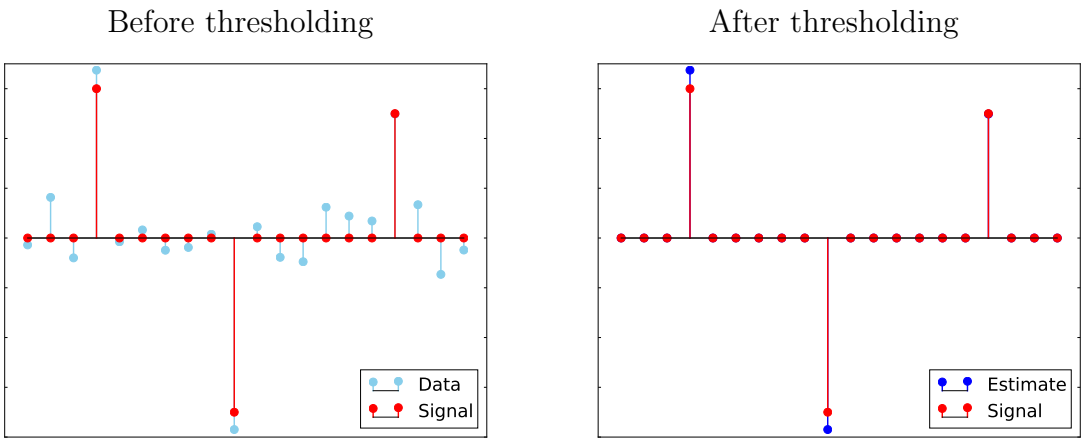

Figure 19: Denoising via hard thresholding.

Proof. By elementary properties of Gaussian random vectors, $U^T z$ is Gaussian with zero mean and covariance matrix

$$\Sigma = U^T \left(\sigma^2 \, \mathrm{I}\right) U = \sigma^2 U^T U = \sigma^2 \, \mathrm{I} \, . \tag{68}$$

□

Figures 20 and 21 show the results of applying this denoising method to the signals in Figures 1 and 13. Both signals are corrupted by additive Gaussian noise. In both cases, exploiting the sparse decomposition allows us to denoise the data very effectively.

Thresholding can also be applied in conjunction with a sparse analysis model. If we suspect that the inner products between a signal x and the atoms in a dictionary D are mostly zero then we can threshold $D^T y$ to denoise the coefficients,

$$\hat{c} = \mathcal{H}_\eta \left(D^T y\right) \tag{69}$$

$$= \mathcal{H}_\eta \left(D^T x + D^T z\right) , \tag{70}$$

$$\tag{71}$$

This method will be effective if $D^T z$ is dense (recall that z denotes the noise component). Recovering an approximation to the signal from the thresholded vector $\mathcal{H}_\eta \left(D^T y\right)$ requires applying a left inverse L that might introduce some distortion,

$$\hat{x} = L\hat{c}, \qquad LD^T = \mathrm{I} \, . \tag{72}$$

An alternative approach is to enforce the analysis model within an optimization problem, as we explain in Section **??**. Thresholding STFT coefficients is a popular denoising technique in speech processing. Figures 24 and **??** show an example with real data.

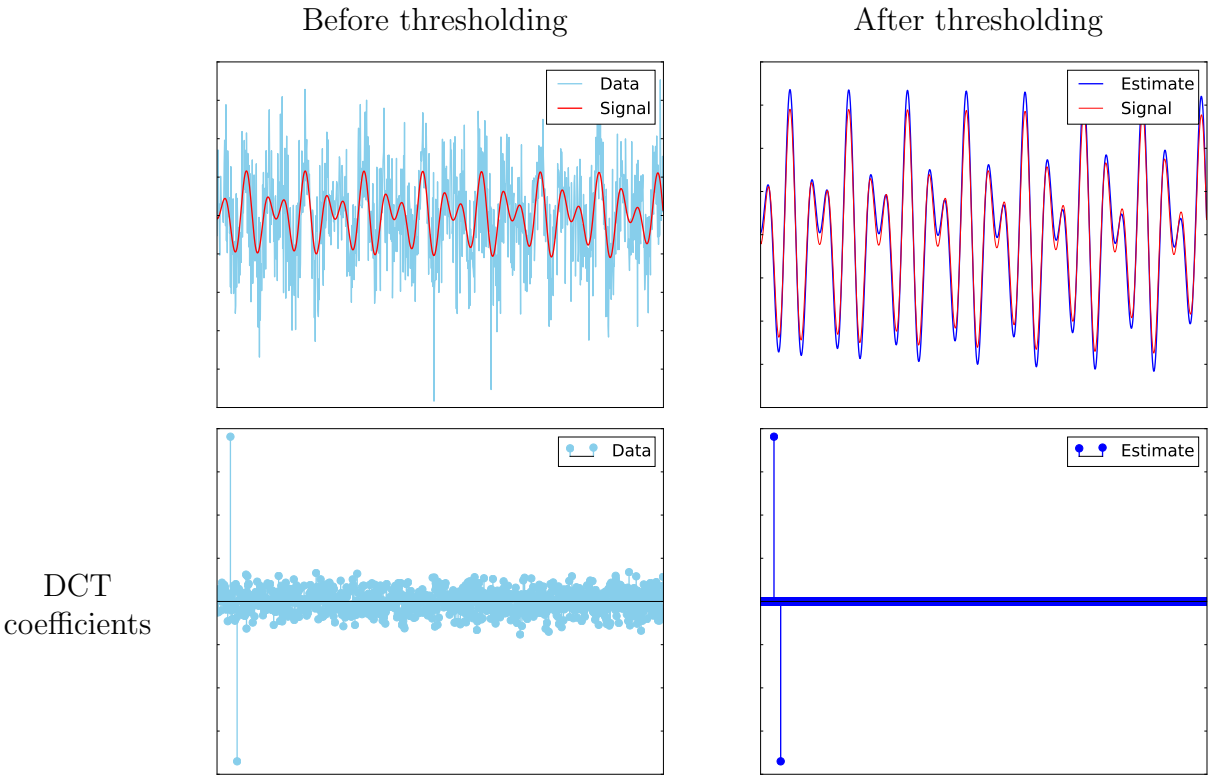

Figure 20: Denoising via hard thresholding in the DCT basis.

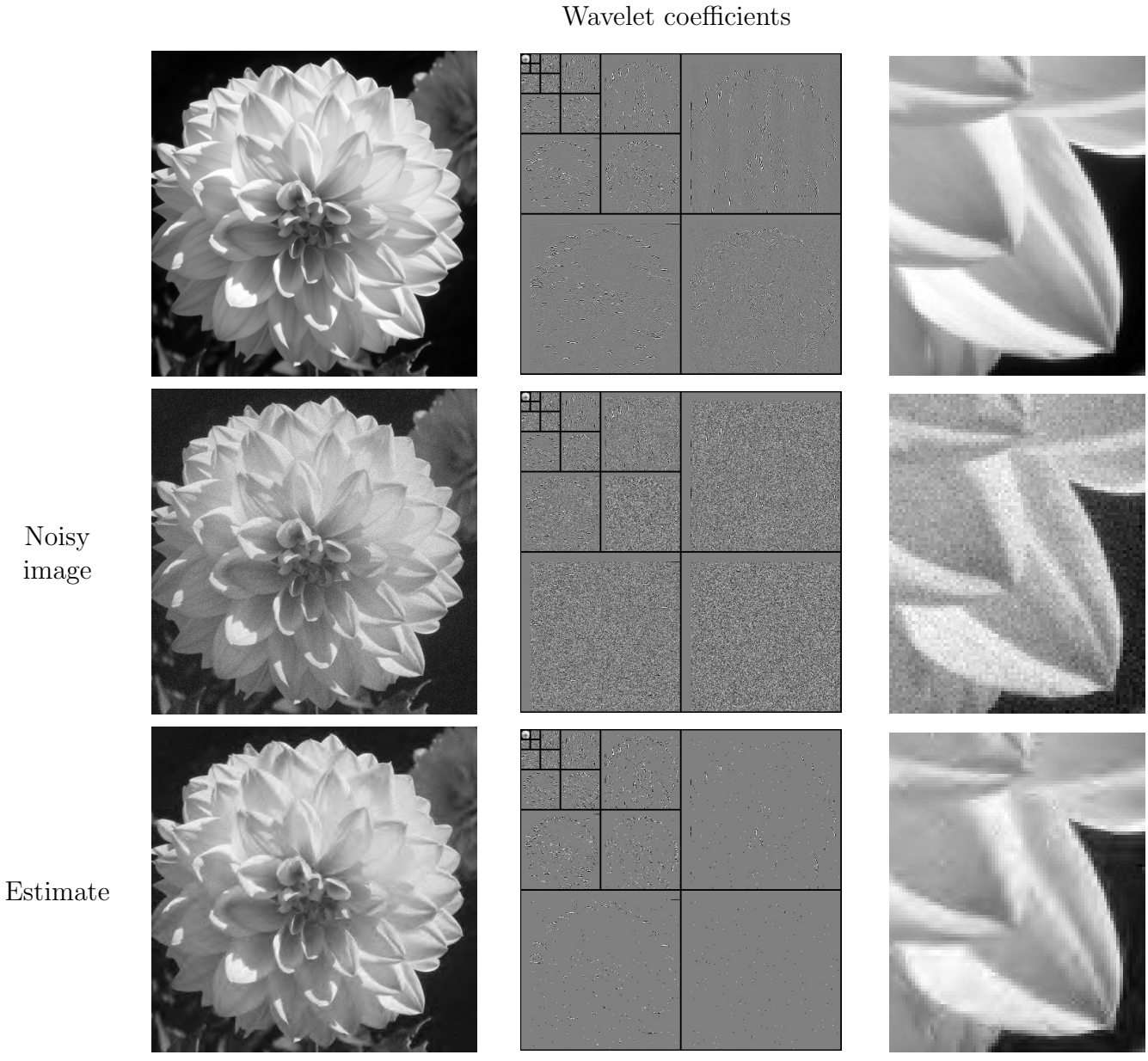

Figure 21: Denoising via hard thresholding in a biorthogonal wavelet basis.

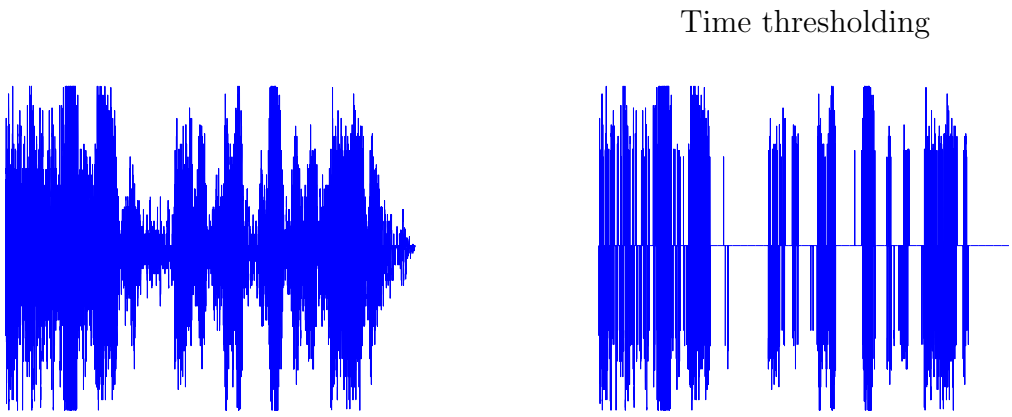

Figure 22: Time thresholding applied to the noisy data shown in Figure 6. The result sounds terrible because the thresholding eliminates parts of the speech.

4.3 Block thresholding

When we apply transforms that capture localized details of signals, such as the wavelet transform or the STFT, sparse representations tend to be highly structured. For example, nonzero wavelet coefficients are often clustered around edges. This is apparent in Figure 13. The reason is that several localized atoms are needed to reproduce sharp variations, whereas a small number of coarse-scale atoms suffice to represent smooth areas of the image.

The assumption that the coefficients of a signal are grouped together is called *group sparsity*. Thresholding-based denoising of group-sparse signals should take into account this structure. It is probably a good idea to threshold an isolated coefficient that is not too large, but a similar coefficient that lies near large nonzero coefficients is likely to contain useful information and should not be discarded.

Block thresholding exploits group sparsity by thresholding the ℓ_2 norm of groups of coefficients. The coefficients are partitioned into blocks $\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_k$. The blocks are then thresholded one by one.

$$\mathcal{B}_\eta\left(x\right)_i := \begin{cases} x_i & \text{if } i \in \mathcal{I}_j \quad \text{such that } \left|\left|x_{\mathcal{I}_j}\right|\right|_2 > \eta, \\ 0 & \text{otherwise.} \end{cases} \tag{73}$$

4.4 Speech denoising

We use a real speech denoising example to compare the effect of thresholding in different domains and of applying block thresholding. The recording shown in Figures 6 and 7 is a short snippet from the movie *Apocalypse Now* where one of the character talks over the noise of a helicopter. We denoise the data using the following methods (click on the links to hear

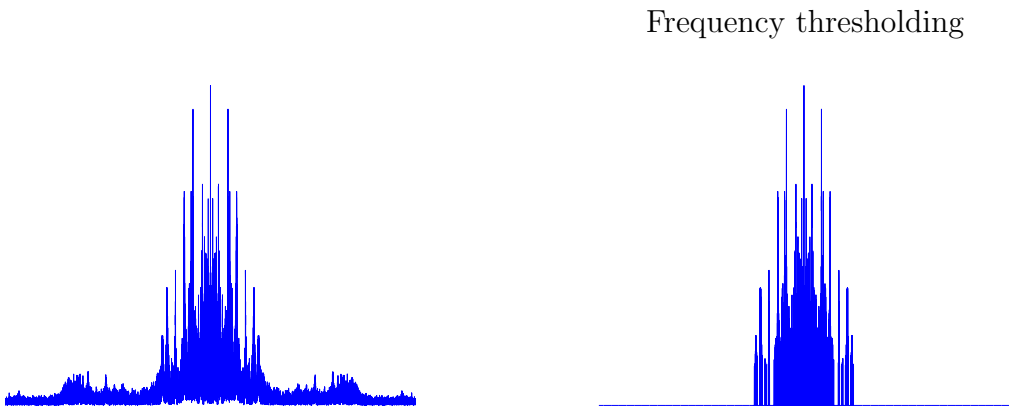

Figure 23: Frequency thresholding applied to the noisy data shown in Figure 6. The result is very low pitch because the thresholding eliminates the high frequencies of both the speech and the noise.

the result):

- Time thresholding: The result, which is plotted in Figure 22, sounds terrible because the thresholding eliminates parts of the speech.
- Frequency thresholding: The result has very low pitch because the thresholding eliminates the high frequencies of both the speech and the noise. The spectrum is shown in Figure 22 before and after thresholding.
- STFT thresholding: The result is significantly better but isolated STFT coefficients that are not discarded produce *musical noise* artifacts. The corresponding spectrogram is shown in Figure 24.
- STFT block thresholding: The result does not suffer from musical noise and retains some of the high-pitch speech. The corresponding spectrogram is shown in Figure 24.

The results are compared visually for a small time segment of the data in Figure 25.

4.5 Synthesis model

A discussed previously, overcomplete dictionaries are not invertible. This means that we cannot apply a linear transform to the noisy signal and threshold in order to exploit the synthesis model. However, we can apply the methods we studied in Section 3 for estimating synthesis coefficients in the noiseless case. In particular, we can adapt the approach based on ℓ_1-norm minimization by eliminating the equality constraint in Problem **??** and adding a

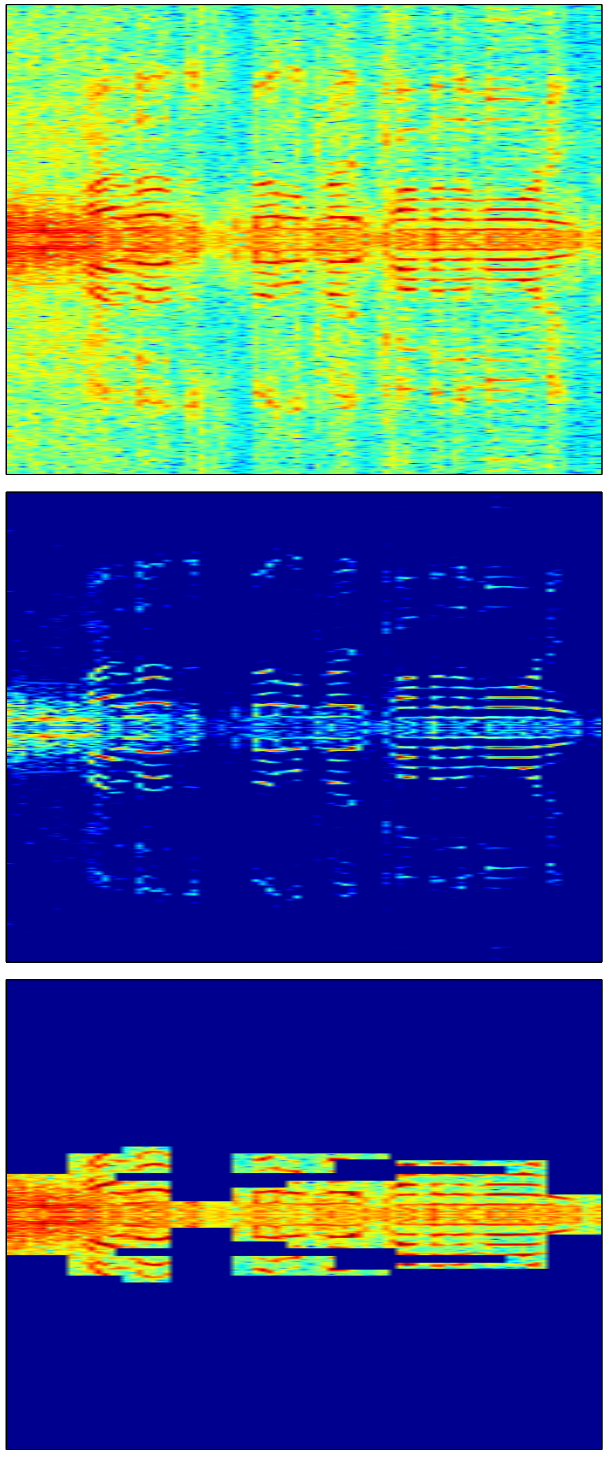

Figure 24: Spectrograms of the noisy signal (above) compared to the estimates obtained by simple thresholding (center) and block thresholding (bottom). The result of simple thresholding contains musical noise caused by particularly large STFT coefficients caused by the noise that were not thresholded. The result of block thresholding does not suffer from these artifacts.

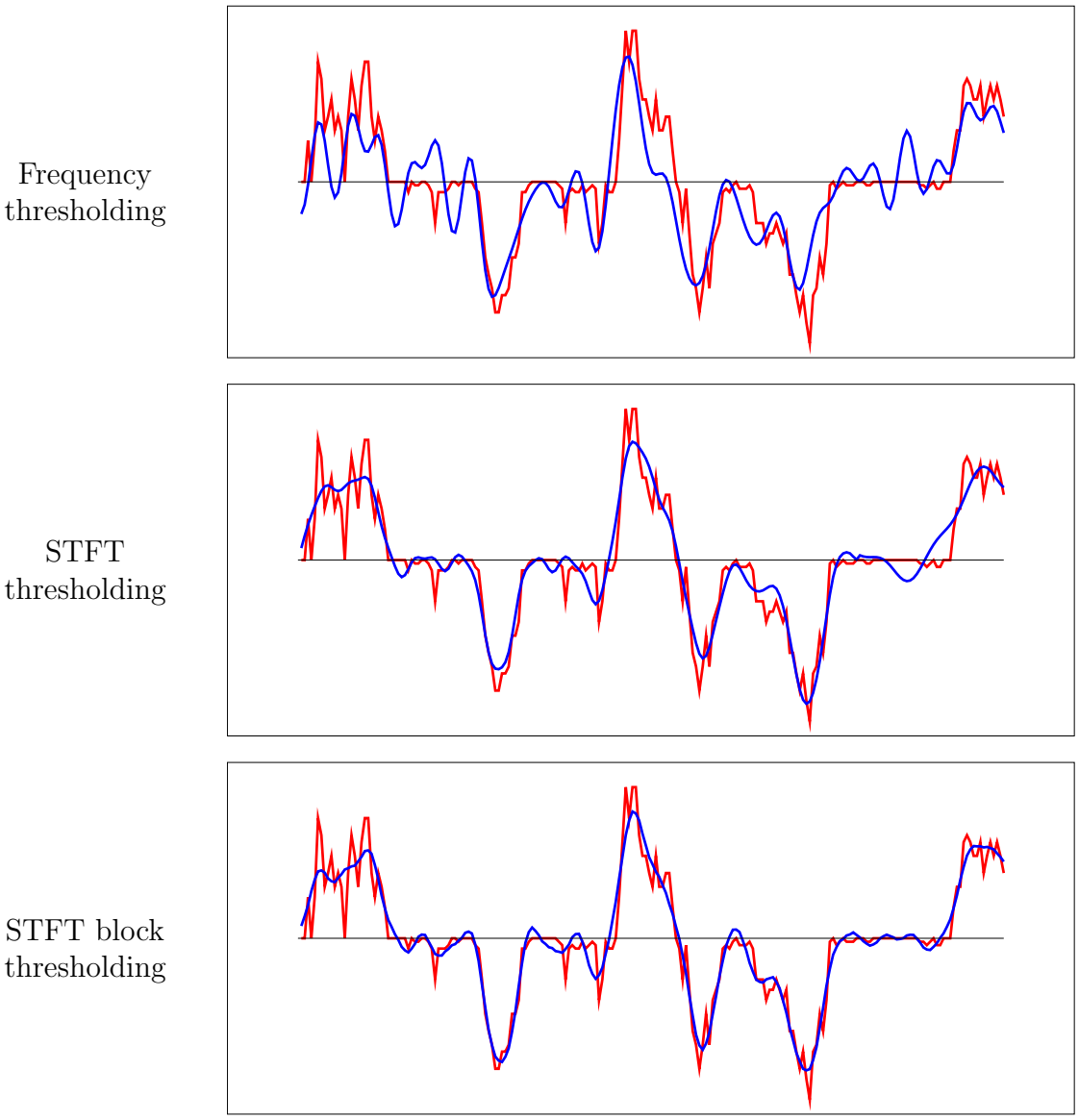

Figure 25: Comparison of the original noisy data (blue) with the denoised signal for the data shown in Figure 6. We compare frequency thresholding (above) and thresholding (center) and block thresholding (below) of STFT coefficients.

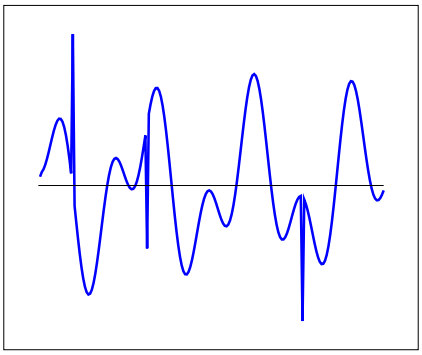

Figure 26: A signal consisting of a superposition of spikes and sinusoids.

data-fidelity term to the cost function. This is known as basis-pursuit denoising [2]. Thus, we estimate the coefficients by solving

$$\hat{c} = \arg\min_{\tilde{x} \in \mathbb{R}^m} ||y - D\tilde{c}||_2^2 + \lambda \, ||\tilde{c}||_1 \tag{74}$$

$$\hat{x} = D\hat{c}, \tag{75}$$

where $\lambda > 0$ is a regularization parameter that determines the tradeoff between the term that promotes sparsity and the term that promotes data fidelity.

The signal in Figure 26 is not sparse either in a basis of spiky atoms or sinusoidal atoms. However, it is sparse in a dictionary that contains *both* sinusoids and spikes,

$$x = Dc = \begin{bmatrix} I & F \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = a + Fb, \tag{76}$$

where $I \in \mathbb{R}^{n \times n}$ is the identity matrix and $F \in \mathbb{R}^{n \times n}$ is a DCT matrix. Figure 27 shows the result of applying ℓ_1-norm regularization to denoise a noisy version of the signal.

4.6 Analysis model

In Section 4.2 we explained how to apply thresholding to the coefficients of a noisy signal in an overcomplete dictionary. To retrieve an approximation of the signal we were forced to apply a left inverse matrix that could distort the result. A more principled way to enforce a sparse analysis model while denoising is to solve the ℓ_1-norm regularized problem

$$\hat{x} = \arg\min_{\tilde{x} \in \mathbb{R}^m} ||y - \tilde{x}||_2^2 + \lambda \left|\left|A^T \tilde{x}\right|\right|_1, \tag{77}$$

which can be interpreted as a tractable relaxation of the problem

$$\text{minimize} \quad \left|\left|D^T \tilde{x}\right|\right|_0 \tag{78}$$

$$\text{subject to} \quad y \approx \tilde{x}. \tag{79}$$

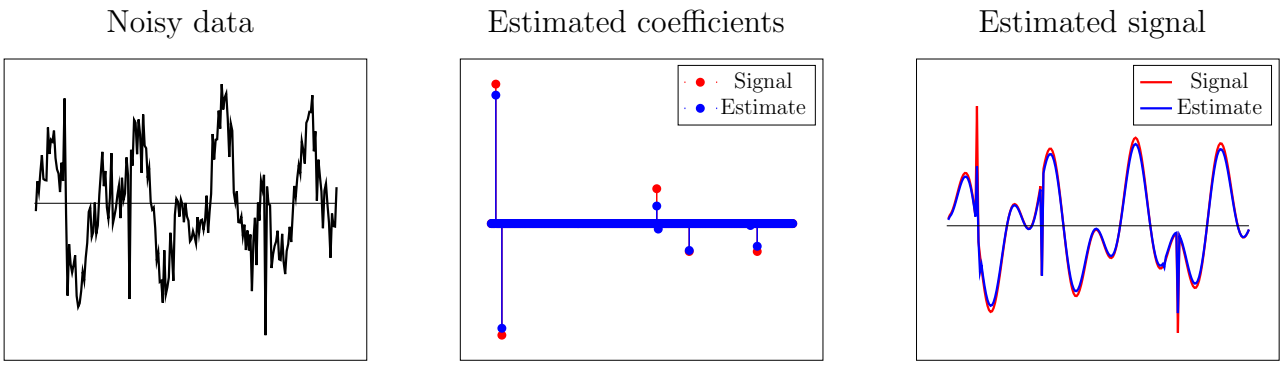

Figure 27: Denoising via ℓ_1-norm-regularized least squares.

Although this problem is very similar to the ℓ_1-norm regularized synthesis formulation (74) it is significantly more challenging to solve.

In image processing, an extremely popular analysis operator is the finite-differences operator. The reason is that images are often well approximated as piecewise constant, which means that they have sparse gradients. We define the total variation of an image Im as the ℓ_1-norm of the horizontal and vertical components of its gradient

$$\text{TV}\left(\text{Im}\right) := \left|\left|\nabla_x \, \text{Im}\right|\right|_1 + \left|\left|\nabla_y \, \text{Im}\right|\right|_1 . \tag{80}$$

If the image is corrupted by noise that is not piecewise constant, we can enforce the prior that the gradient is sparse by penalizing the total variation of the estimate. This is equivalent to applying ℓ_1-norm regularization with an analysis operator that computes the discretized gradient of the image (i.e. a finite-differences operator),

$$\widehat{\text{Im}} = \arg \min_{\widetilde{\text{Im}} \in \mathbb{R}^{n \times n}} \left|\left|Y - \widetilde{\text{Im}}\right|\right|_{\text{F}}^2 + \lambda \, \text{TV}\left(\widetilde{\text{Im}}\right) . \tag{81}$$

Figures 29, 30 and **??** display the results of applying total-variation denoising to a one-dimensional piecewise constant signal and to a real image. Small values of the regularization parameter do not denoise well, whereas large values produce cartoonish estimates. Medium values however allow to denoise quite effectively. We refer the interested reader to [1, 6] for more details on total-variation regularization.

References

The book *A wavelet tour of signal processing* by Mallat [3] is a great reference for the topics discussed in these notes. Numerical experiments by Gabriel Peyré illustrating many of the ideas that we have discussed are available here.

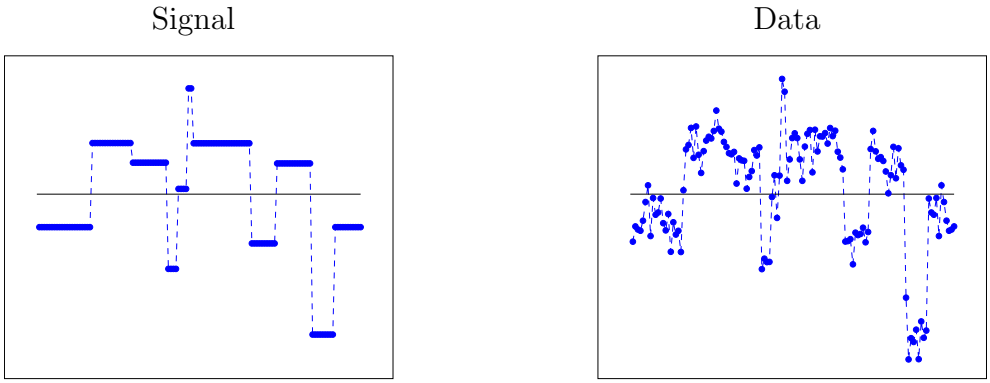

Figure 28: One-dimensional signal (left) and the corresponding noisy data (right).

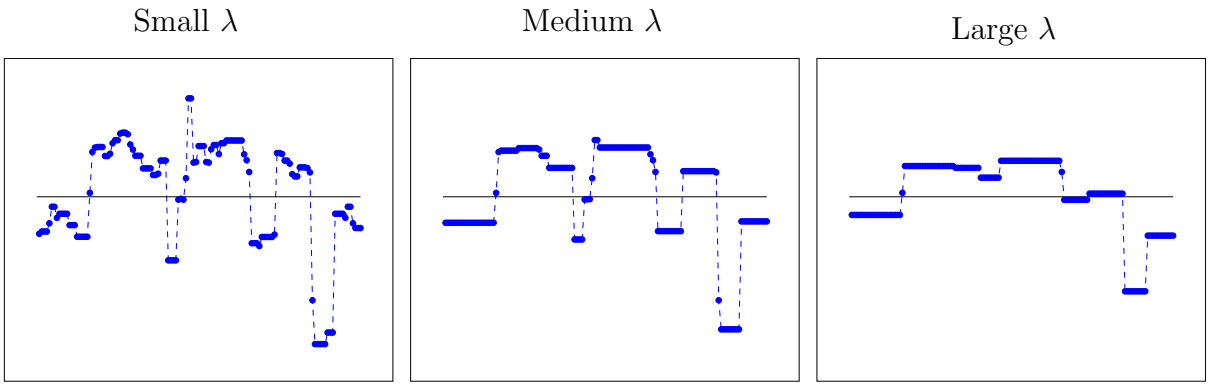

Figure 29: Total-variation denoising applied to the data in Figure 28 for different values of the regularization parameter.

Figure 30: Total-variation denoising applied to the image in Figure 21 for different values of the regularization parameter.

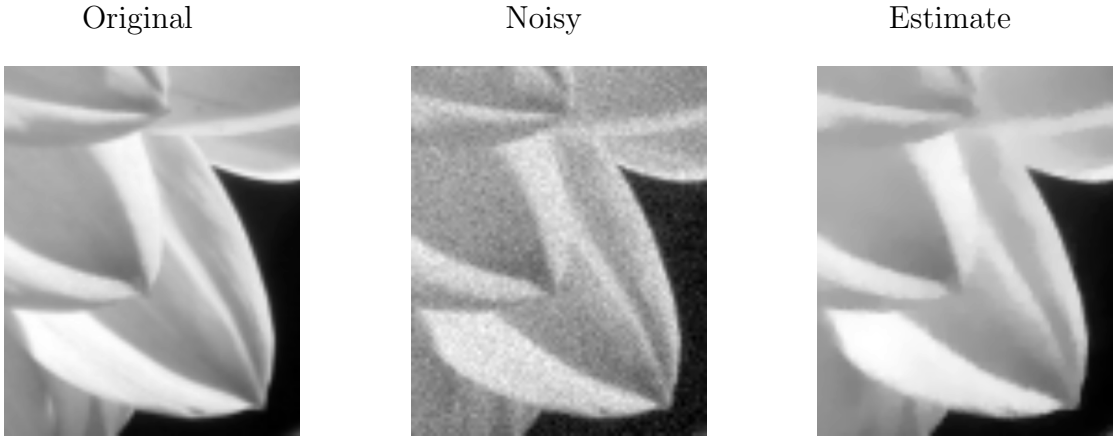

Figure 31: Total-variation denoising applied to the image in Figure 21 for different values of the regularization parameter.

[1] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical imaging and vision*, 20(1-2):89–97, 2004.

[2] S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Review*, 43(1):129–159, 2001.

[3] S. Mallat. *A wavelet tour of signal processing: the sparse way*. Academic press, 2008.

[4] S. G. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415, 1993.

[5] Y. C. Pati, R. Rezaiifar, and P. Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *27th Asilomar Conference on Signals, Systems and Computers*, pages 40–44. IEEE, 1993.

[6] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.

A Proofs

A.1 Proof of Lemma 3.1

Consider the SVD of $D = USV^*$. The columns of $V \in \mathbb{R}^{m \times n}$ are an orthonormal basis for the row space of D row (D). Let us decompose any coefficient vector c' such that $x = Dc'$ as

$$c' = Vb + \mathcal{P}_{\text{row}(D)^\perp}\ (c') \tag{82}$$

where b is an n-dimensional vector. If $x = Dc'$, then

$$\begin{aligned} S^{-1}U^T x &= V^* c' && (83)\\ &= V^* \left(Vb + \mathcal{P}_{\text{row}(D)^\perp}\ (c') \right) && (84)\\ &= b, && (85) \end{aligned}$$

so b has the same value for any c' such that $x = Dc'$. We can decompose the norm of c' in the following way by Pythagoras's Theorem,

$$\begin{aligned} ||c'||_2^2 &= \left|\left|\mathcal{P}_{\text{row}(D)}\ (c')\right|\right|_2^2 + \left|\left|\mathcal{P}_{\text{row}(D)^\perp}\ (c')\right|\right|_2^2 && (86)\\ &= ||b||_2^2 + \mathcal{P}_{\text{row}(D)^\perp}\ (c')\,. && (87) \end{aligned}$$

Any solution such that $\mathcal{P}_{\text{row}(D)^\perp}\ (c') \neq 0$ will have norm greater than $||b||_2^2$ so the minimum norm solution is

$$\begin{aligned} Vb &= VS^{-1}U^T x && (88)\\ & D^T \left(DD^T\right)^{-1} x. && (89) \end{aligned}$$