Wavelet Methods for Curve and Surface Estimation

David Richard Mark Herrick

January 2000

A thesis submitted to the University of Bristol in accordance with the requirements of the degree of Doctor of Philosophy in the School of Mathematics of the Faculty of Science.

Abstract

Over the last decade wavelets have been successfully applied to many statistical smoothing problems. In this thesis two smoothing applications will be thoroughly examined. The first of these is probability density function estimation. Although the theoretical performance of wavelet density estimators has been studied, as yet the behaviour of non-linear wavelet density estimators has not been extensively evaluated in practice. An efficient algorithm for finding the empirical wavelet coefficients from the data is implemented, and several new ideas for thresholding these coefficients are investigated. The second application is the extension of two dimensional wavelet shrinkage to handle data with an irregularly spaced design. This is achieved by extrapolating the method proposed by Kovac & Silverman (2000) for extending the one dimensional wavelet shrinkage algorithm so that unequally spaced data can be smoothed. The method proposed is successfully implemented and is believed to be a useful addition to the library of spatial smoothing methods.

Acknowledgements

I would like to thank my advisor Guy Nason for all his help and encouragement over the last three (and a bit) years. Naturally, many of the ideas in this thesis originated with him. I would also like to thank Bernard Silverman, discussions with whom yielded some of the ideas about density estimation in Chapter 3, and all the other members of the statistics group. Particular thanks to Miles Harkness for sharing an office with me for all three years, especially for those four months he was in Australia ... I am sure that Miles will join me in thanking Jon and Marcel for setting such a sterling example to us youngsters.

Away from Bristol University Department of Mathematics, I would like to thank the EPSRC for supporting me with one of their studentships, and Jean Golding for being such an understanding employer whilst I finished writing up.

Thanks, Will, for those afternoons in the pub discussing what you were up to with your frozen birds. You may have got married a couple of months before me and bought a house a couple of months before me, but it looks like I've got my thesis in first!

Finally, I would like to thank Sarah, my wife, for all her support, especially during the final few months of writing up. No one will be as delighted to see this thesis finished as she will. I am also extremely grateful that she had the good sense to get a job rather than doing a PhD herself. I think I quite like being a kept man!

Author's Declaration

I declare that the work in this thesis was carried out in accordance with the Regulations of the University of Bristol. The work is original except where indicated by special reference in the text and no part of the thesis has been submitted for any other degree. Any views expressed in the thesis are those of the author and in no way represent those of the University of Bristol. The thesis has not been presented to any other University for examination either in the United Kingdom or overseas.

David R.M. Herrick

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The term "wavelet" originates from the early 1980s with the work of Morlet *et al.* (1982), in the context of analysis of seismic reflection data. Since then wavelets have found application in many different areas, such as in signal processing, for example Mallat (1989b), and image compression, for example Brislawn (1996) and Schomer *et al.* (1998). Their application in statistics stems from the early 1990's with the work of Donoho and Johnstone. Wavelets are a mathematical tool which can be thought of in many ways. One possibility is to view them as orthonormal basis functions that are localised in both time and scale. This means that they are able to model high frequency components of a signal, such as discontinuities, well. Another view of wavelets is as filters, and from this stems a fast, orthogonal, discrete wavelet transform. These key properties, combined with the fact that wavelets model smooth functions very efficiently, mean that wavelets provide an excellent tool for statistical denoising.

The standard wavelet shrinkage methodology, for example Donoho & Johnstone (1994), for denoising a univariate signal makes a number of assumptions. Firstly, the observed data values must have equally spaced sites and secondly the noise is assumed to be independent, identically distributed normal noise. Recent work, particularly Johnstone & Silverman (1997) and Kovac & Silverman (2000), has attempted to develop wavelet methodology so that data which do not meet these assumptions can be analysed. Some key points in the mathematical theory of wavelets, and their applications in statistics will be discussed in more detail in Chapter 2.

The application of wavelets to the estimation of probability density functions has already been discussed in the literature. These discussions have mainly concentrated on theoretical aspects, with practical considerations limited to linear estimators. Full non-linear, thresholded estimators have not been considered much in practice. This is one of the topics that will be considered in this thesis, and will be the subject of Chapter 3. The two main practical problems are how to estimate the wavelet coefficients from the data and how to threshold the noisy coefficients. In the standard regression methodology, the raw data values can be shown to approximate the scaling function coefficients at some high resolution, up to a factor of some power of two. However, such an approximation cannot be used with single sample data of the sort used for density estimation. Instead, the use of a direct projection method is proposed. This itself brings forward several problems of its own, mainly in the area of computational efficiency. Also, the resulting wavelet coefficients are correlated, and the nature of this correlation structure does not fit into the framework imposed in Johnstone & Silverman (1997).

Wavelet smoothing methods can be extended to two (and more) dimensions. The analogue of the equal spacing requirement in two dimensions is that the data sites must lie on a regularly spaced grid. The literature on 2D wavelet methods has mainly concentrated on image analysis, and image data are, by nature, regularly spaced. However, in surface estimation problems the data are unlikely to be observed over such a regular grid. A natural question to ask is whether the Kovac-Silverman method for handling unequally spaced data in 1D can be extended practically to two dimensions. This is the other main area that will be investigated in this thesis, in Chapter 4. As with the density estimation, many of the problems are of a computational nature. Interpolation in 2D is much less straightforward than in 1D. In addition, storage is an important consideration, since the arrays of variances and covariances of the data and wavelet coefficients, which are an important part of the algorithm, are potentially very large. In Chapter 5, the resulting 2D methodology will be compared with a standard method of spatial smoothing in a simulation study.

The software produced for both the density estimation and irregularly spaced 2D smoothing applications will be made available as a technical report from:

`http://www.stats.bris.ac.uk/pub/Reports.html`.

Chapter 2

Literature Review

In this chapter the literature relevant to the material that will be covered in Chapters 3 – 5 will be reviewed. The first two sections summarise some of the mathematical theory behind wavelets. §2.1 will look at the definition of wavelets, their properties and examine some examples. In §2.2 some wavelet transforms, which are the key to the use of wavelets as a mathematical tool, will be investigated. The uses of wavelets in statistics will then be introduced, with §2.3 looking at the standard application of wavelets to 1D smoothing problems. Some generalisations of the standard method will be considered in §2.4. The extension of wavelet theory to two dimensions will be discussed in §2.5. The final two sections will consider probability density function estimation, with §2.6 looking at non-parametric approaches in general and §2.7 reviewing some existing material on the application of wavelets to density estimation.

2.1 Wavelets and Multiresolution Analysis

2.1.1 Wavelets

Wavelets are orthonormal basis functions with many useful properties. They are dilations and translations of a *mother wavelet* ψ. Following Meyer (1992), a mother wavelet can be defined as follows:

Definition 2.1 *Let m be a non-negative integer. A function $\psi(x)$ of a real variable is called a* mother wavelet of class m *if the following properties hold:*

(a) if $m = 0$, $\psi(x) \in L^\infty(\mathbb{R})$*; if* $m \geqslant 1$, $\psi(x)$ *and all its derivatives up to order* m *belong to* $L^\infty(\mathbb{R})$*;*

(b) $\psi(x)$ *and all its derivatives up to order* m *decrease rapidly as* $x \to \pm\infty$*;*

(c) $\int_{-\infty}^{\infty} x^k \psi(x) = 0$ *for* $0 \leqslant k \leqslant m$*;*

(d) the collection of functions $2^{j/2}\psi(2^j x - k)$, $j,\ k \in \mathbb{Z}$, *is an orthonormal basis of* $L^2(\mathbb{R})$.

The functions $\psi_{jk}(x) = 2^{j/2}\psi(2^j x - k)$ are the wavelets. Condition (a) expresses the regularity of the wavelet and conditions (b) and (c) localisation and oscillatory character respectively. Condition (c) is known as the *vanishing moments* property.

Many different mother wavelets exist, e.g. the well known Haar basis, the Shannon, or Littlewood-Paley wavelet and the Meyer wavelet. These are all described in section 3.4 of Vidakovic (1999). More detailed and mathematical descriptions of wavelets can be found in Meyer (1992) and Daubechies (1992).

An important factor in the characterisation of wavelet bases is multiresolution analysis. It turns out that all wavelets of practical interest are associated with a multiresolution analysis (see chapter 5 of Daubechies (1992)). Indeed, Daubechies (1988) uses the multiresolution analysis framework to derive families of compactly supported wavelets.

2.1.2 Multiresolution Analysis

Multiresolution Analysis (MRA) provides a general approach to building wavelet orthonormal bases. Good introductions to multiresolution analysis and wavelets can be found in Mallat (1989a), Jawerth & Sweldens (1994), Daubechies (1988) and Daubechies (1992). Following Mallat (1989a) and Daubechies (1992) a multiresolution analysis can be defined as follows.

Definition 2.2 *A* multiresolution analysis *consists of a family of embedded closed subspaces* $V_j \subset L^2(\mathbb{R})$, $j \in \mathbb{Z}$,

$$\cdots \subset V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \cdots \tag{2.1}$$

such that

$$\bigcap_{j\in\mathbb{Z}} V_j = \{0\} \ , \tag{2.2}$$

$$\overline{\bigcup_{j\in\mathbb{Z}} V_j} = L^2(\mathbb{R}) \ , \tag{2.3}$$

and

$$f \in V_j \Leftrightarrow f(2\,\cdot) \in V_{j+1} \ , \tag{2.4}$$

and,

$$f \in V_0 \Leftrightarrow f(\cdot - k) \in V_0, \ \forall\, k \in \mathbb{Z} \ ; \tag{2.5}$$

moreover, there is a scaling function $\phi \in V_0$ *such that* $\{\phi_{0k} : k \in \mathbb{Z}\}$ *constitutes an orthonormal basis for* V_0. *Conditions (2.4) and (2.5) imply that,* $\forall j \in \mathbb{Z}$, *the* $\{\phi_{jk} : k \in \mathbb{Z}\}$ *constitute an orthonormal basis for* V_j, *where*

$$\phi_{jk}(x) = 2^{j/2}\phi(2^j x - k) \ .$$

The MRA and ϕ are called *regular*, with regularity r, if $\phi \in C^r$, and ϕ and every derivative up to order r are rapidly decreasing, i.e. for every integer $m \geqslant 0$ there exists a constant A_m such that

$$|\phi^{(j)}(x)| \leqslant \frac{A_m}{(1+|x|)^m} \quad \text{for } j = 0, 1, \ldots, r \ .$$

An example of spaces V_j that satisfy (2.1) – (2.5) is the set of spaces of functions that are piecewise constant over appropriate intervals:

$$V_j = \left\{ f \in L^2(\mathbb{R}) : \forall k \in \mathbb{Z}, \quad f|_{[2^{-j}k, 2^{-j}(k+1))} \text{ is constant} \right\} \ .$$

For reasons that will be given below, this is called the *Haar* multiresolution analysis. A possible choice for ϕ is the indicator function for $[0, 1)$:

$$\phi(x) = \begin{cases} 1 & \text{if } 0 \leqslant x < 1 \\ 0 & \text{otherwise} \end{cases} \ . \tag{2.6}$$

Let P_j be the orthogonal projection operator onto V_j. Then the approximation $P_j f$ of a function f at resolution 2^j is given by:

$$\big(P_j f\big)(x) = \sum_{k=-\infty}^{\infty} c_{jk}\phi_{jk}(x) \ ,$$

where

$$c_{jk} = \int_{-\infty}^{\infty} \phi_{jk}(x) f(x)\, dx \ . \tag{2.7}$$

Condition (2.3) insures that $\lim\limits_{j\to\infty} P_j f = f$ for all $f \in L^2(\mathbb{R})$. Figure 2.1 shows successive approximations of the sine curve at four resolutions using the Haar MRA.

2.1.3 Orthonormal Wavelet Bases

Consider the projection operator P_j. The additional information available in $P_{j+1}f$ as compared with $P_j f$, is given by an orthogonal projection on the orthogonal complement of V_j in V_{j+1}. Now define, for all $j \in \mathbb{Z}$, W_j to be this orthogonal complement. Then

$$\begin{aligned} & V_{j+1} = V_j \oplus W_j \ , && (2.8) \\ \text{with} \quad & W_j \perp V_j && \\ \text{and} \quad & W_j \perp W_{j'} \quad \text{if} \quad j \neq j' \ . && \end{aligned}$$

It follows that, for $j > L$:

$$V_j = V_L \oplus \bigoplus_{k=L}^{j-1} W_k \ , \tag{2.9}$$

where the subspaces $\{W_k\}_{k=L}^{j-1}$ are all mutually orthogonal. In conjunction with (2.2) and (2.3) this implies that

$$\bigoplus_{j\in\mathbb{Z}} W_j = L^2(\mathbb{R}) \ . \tag{2.10}$$

In other words $L^2(\mathbb{R})$ can be decomposed into mutually orthogonal subspaces. The W_j inherit the scaling property (2.4) from the V_j:

$$f \in W_j \Leftrightarrow f(2\,\cdot) \in W_{j+1} \ . \tag{2.11}$$

If a function ψ can be found, such that its integer translates form an orthonormal basis for W_0, then properties (2.10) and (2.11) mean that $\{\psi_{jk} : j,k \in \mathbb{Z}\}$ will form an orthonormal basis for $L^2(\mathbb{R})$, where

$$\psi_{jk}(x) = 2^{j/2}\psi(2^j x - k) \ .$$

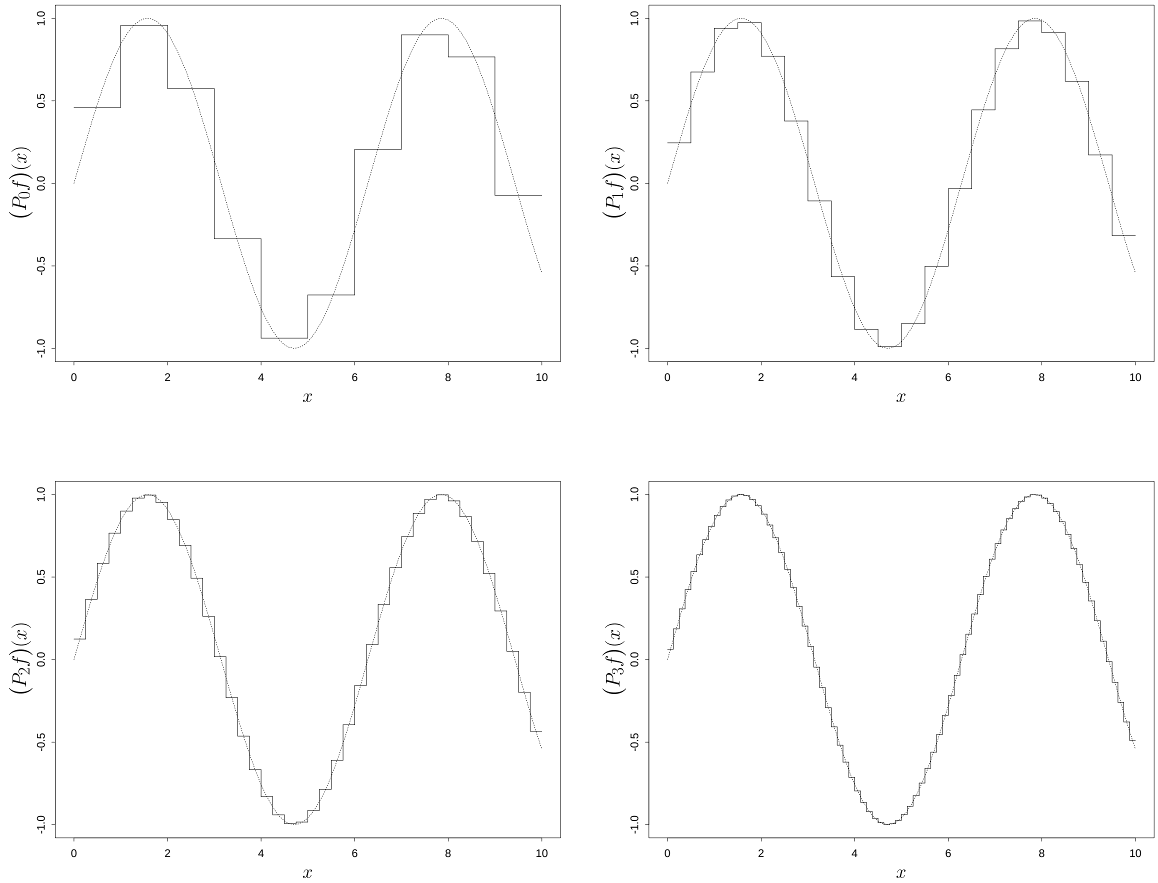


Figure 2.1: Successive approximations (solid lines) of the sine function (dotted lines) using the Haar MRA.

Such a ψ can be found as follows (see, for example, Daubechies (1992) or Mallat (1989a)), and is often called the *mother wavelet*. First consider some properties of ϕ. Since $\phi \in V_0 \subset V_1$, there exist h_n such that

$$\phi(x) = \sum_n h_n \phi_{1n}(x) \ , \tag{2.12}$$

$$\text{where} \qquad h_n = \langle \phi, \phi_{1n} \rangle = \int \phi(x)\phi_{1n}(x)\, dx \ .$$

Note that this means that for all $j, k \in \mathbb{Z}$,

$$\phi_{j-1,k}(x) = \sum_n h_{n-2k}\phi_{jn}(x) \ , \tag{2.13}$$

and that since the ϕ_{jk}'s are orthonormal:

$$\sum_{n\in\mathbb{Z}} |h_n|^2 = 1 \ .$$

Now define the Fourier transform $\hat{f}$ of a function f as:

$$\hat{f}(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-i\omega x} f(x)\, dx \ .$$

Taking the Fourier transform of (2.12) therefore gives:

$$\hat{\phi}(\omega) = H(\omega/2)\hat{\phi}(\omega/2) \ , \tag{2.14}$$

$$\text{where} \qquad H(\omega) = \frac{1}{\sqrt{2}} \sum_n h_n e^{-in\omega} \ . \tag{2.15}$$

The orthonormality of the $\phi(\cdot - k)$ leads to the condition (Daubechies (1992), p132):

$$|H(\omega)|^2 + |H(\omega + \pi)|^2 = 1 \ . \tag{2.16}$$

Mallat (1989a) proves that $|\hat{\phi}(0)| = 1$ and so from (2.14):

$$|H(0)| = 1 \ .$$

Similarly, since $\psi \in W_0 \subset V_1$, there exist g_n such that

$$\psi(x) = \sum_n g_n \phi_{1n}(x) \ . \tag{2.17}$$

Again, note that for all $j, k \in \mathbb{Z}$:

$$\psi_{j-1,k}(x) = \sum_n g_{n-2k}\phi_{jn}(x) \ . \tag{2.18}$$

Taking the Fourier transform of (2.17):

$$\hat{\psi}(\omega) = G(\omega/2)\hat{\phi}(\omega/2) \ ,$$

$$\text{where} \qquad G(\omega) = \frac{1}{\sqrt{2}} \sum_n g_n e^{-in\omega} \ .$$

By considering the decomposition of a function $f \in V_0$ into orthogonal components in V_{-1} and W_{-1}, Mallat (1989a) shows that for the decomposition to be orthogonal the following conditions must be satisfied:

$$|H(\omega)|^2 + |G(\omega)|^2 = 1$$

$$H(\omega)\overline{G(\omega)} + H(\omega+\pi)\overline{G(\omega+\pi)} = 0 \ .$$

He notes that one possible choice for G is:

$$G(\omega) = -e^{-i\omega}\overline{H(\omega+\pi)} \ ,$$

which leads to

$$g_n = (-1)^n h_{1-n} \ .$$

For the Haar MRA this leads to:

$$\psi(x) = \begin{cases} 1 & \text{if } 0 \leqslant x < 1/2 \\ -1 & \text{if } 1/2 \leqslant x < 1 \\ 0 & \text{otherwise} \end{cases} \ , \tag{2.19}$$

which is the well known Haar basis and the reason the Haar MRA is so called.

Daubechies (1992, p136) notes that every orthonormal wavelet basis of practical interest is associated with a MRA, but that it is possible to construct "pathological" ψ such that the ψ_{jk} are an orthonormal basis for $L^2(\mathbb{R})$ but are not derivable from any MRA. She goes on to note that such "pathology" cannot persist if some smoothness is imposed on $\hat{\psi}$, which corresponds to decay on ψ. In Corollary 5.5.4 she introduces a regularity condition which is stronger than is necessary to avoid the "pathology", showing that if $|\phi(x)|$, $|\psi(x)| \leqslant C(1+|x|)^{-m-1-\varepsilon}$ and $\psi \in C^m$ with $\psi^{(l)}$ bounded for $l \leqslant m$, then H factorises as

$$H(\omega) = \left(\frac{1+e^{-i\omega}}{2}\right)^{m+1} \mathcal{L}(\omega) \ , \tag{2.20}$$

where $\mathcal{L} \in C^m$ is 2π periodic. Note that by Theorem 5.5.1 on page 153 of Daubechies (1992) ψ satisfying the above regularity condition will also satisfy:

$$\int x^l \psi(x)\, dx = 0 \quad \text{for } l = 0, 1, \ldots, m \ . \tag{2.21}$$

Properties (2.20) & (2.21) will be important later on.

2.1.4 Daubechies Compactly Supported Wavelets

Daubechies (1988) investigates the construction of compactly supported wavelets. Note that if ϕ and ψ are compactly supported, there will be a finite number of non zero coefficients h_n and g_n. She proves in section 4.A that the only compactly supported wavelet basis for which the scaling function has an axis of symmetry is the Haar basis. She then characterises all orthonormal compactly supported wavelet bases, and illustrates this with a family of examples, known as the *extremal phase* wavelets. In chapter 8 of Daubechies (1992) she introduces another family known as the *least asymmetric* wavelets.

Following Daubechies (1992) for the rest of this section, consider how these wavelets are constructed. The aim is to construct H so that (2.16) is satisfied as well as (2.20) so that ψ has some regularity. Since for compactly supported ϕ there are only finitely many non-zero h_n, from (2.15) $H(\omega)$ will be a trigonometric polynomial. Motivated by the regularity condition (2.20) consider H of the form:

$$H(\omega) = \left(\frac{1+e^{-i\omega}}{2}\right)^{N} \mathcal{L}(\omega) \ , \tag{2.22}$$

with the *filter number* $N \geqslant 1$ and $\mathcal{L}$ a trigonometric polynomial. Therefore

$$|H(\omega)|^2 = \left(\cos^2 \frac{\omega}{2}\right)^N L(\omega) \ ,$$

where $L(\omega) = |\mathcal{L}(\omega)|^2$ is a polynomial in $\cos\omega$. It is convenient to write $L(\omega)$ as a polynomial in $\sin^2\omega/2 = (1-\cos\omega)/2$, so that

$$|H(\omega)|^2 = \left(\cos^2 \frac{\omega}{2}\right)^N P\left(\sin^2 \frac{\omega}{2}\right) \ .$$

For both the extremal phase and least asymmetric wavelets choose, to insure that condition (2.16) is satisfied, $P(y) = P_N(y)$ where

$$P_N(y) = \sum_{k=0}^{N-1} \binom{N-1+k}{k} y^k .$$

This choice completely characterises $L(\omega)$, from which it is necessary to find $\mathcal{L}(\omega)$. This step requires a lemma by Riesz, presented on page 172 of Daubechies (1992). Note that $L(\omega) \geqslant 0$, $\forall\omega$. Since L is a polynomial in $\cos\omega$ it is necessarily of the form

$$L(\omega) = \sum_{m=0}^{M} a_m \cos m\omega \ , \tag{2.23}$$

with $a_m \in \mathbb{R}$. Now write $L(\omega) = p_L(\cos\omega)$ where p_L is a polynomial of degree M with real coefficients. This polynomial can be factorised:

$$p_L(c) = \alpha \prod_{j=1}^{M} (c - c_j) \ , \tag{2.24}$$

where the zeros of p_L appear either in complex duplets $c_j, \overline{c}_j$ or in real singlets.

If c_j is real, consider the zeros of $\frac{1}{2} - c_j z + \frac{1}{2} z^2$, which are given by $c_j \pm \sqrt{c_j^2 - 1}$. For $|c_j| \geqslant 1$, these are two real zeros (degenerate if $c_j = \pm 1$) of the form r_j, r_j^{-1}. For $|c_j| < 1$, the two zeros are complex conjugate and of absolute value 1, i.e. they are of the form $e^{i\alpha_j}$, $e^{-i\alpha_j}$. Since $L \geqslant 0$, these zeros must have even multiplicity. If c_j is not real, then consider it together with $\overline{c}_j$. The polynomial $(\frac{1}{2} - c_j z + \frac{1}{2} z^2)(\frac{1}{2} - \overline{c}_j z + \frac{1}{2} z^2)$ has four zeros, $c_j \pm \sqrt{c_j^2 - 1}$ and $\overline{c}_j \pm \sqrt{\overline{c}_j^2 - 1}$. In fact these are all different and form a quadruplet z_j, z_j^{-1}, $\overline{z}_j$, $\overline{z}_j^{-1}$.

Having defined these zeros, it is now possible to write down $\mathcal{L}$:

$$\begin{aligned} \mathcal{L}(\omega) \quad = \quad & \left[\frac{1}{2} |a_M| \prod_{j=1}^{J} |z_j|^{-2} \prod_{l=1}^{L} |r_l|^{-1} \right]^{1/2} \cdot \prod_{j=1}^{J} (e^{-2i\omega} - 2e^{-i\omega} \mathrm{Re} z_j + |z_j|^2) \\ & \cdot \prod_{k=1}^{K} (e^{-2i\omega} - 2e^{-i\omega} \cos\alpha_j + 1) \cdot \prod_{l=1}^{L} (e^{-i\omega} - r_l) \ , \end{aligned} \tag{2.25}$$

where the different products group the three different types of zeros discussed above. This choice of $\mathcal{L}$ is not unique. For the real zeros either r_l or r_l^{-1} can be chosen to go in $\mathcal{L}$ and similarly either z_j, $\overline{z}_j$ or z_j^{-1}, $\overline{z}_j^{-1}$. For the extremal phase wavelets the zeros that lie outside the unit circle are used. For the least asymmetric wavelets the zeros are carefully chosen so as to maximise the symmetry. Note that for $N = 2, 3$ there

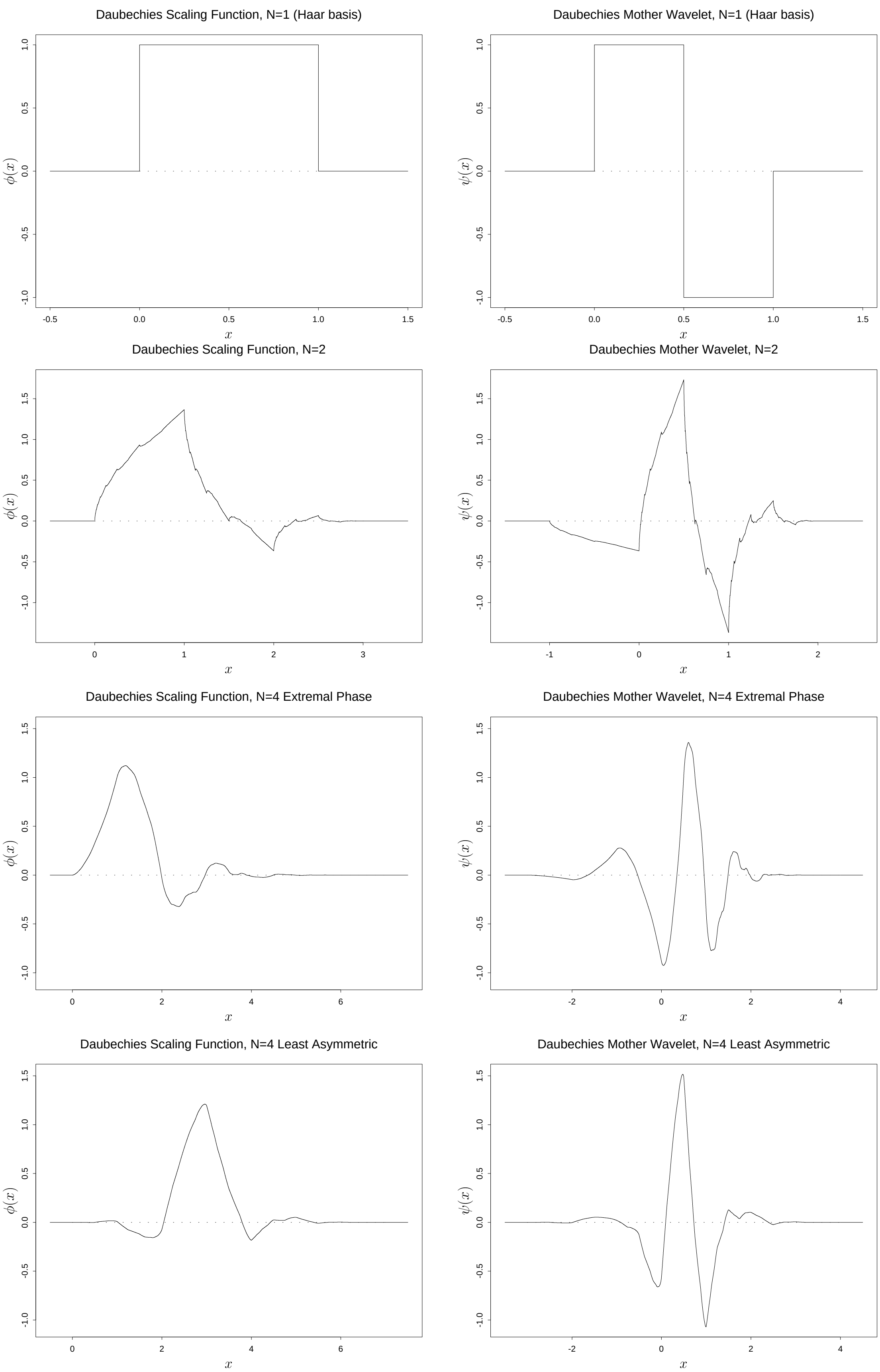


Figure 2.2: Mother wavelets and scaling functions for a selection of Daubechies bases.

is effectively only one pair ϕ, ψ as the alternative selection of zeros just leads to the mirror images.

Example 2.1 *For small values of N the above procedure can be followed analytically. Consider $N=2$. Note that:*

$$P_2(y) = 1 + 2y$$

$$\Rightarrow \qquad L(\omega) = 2 - \cos(\omega) \ ,$$

so that in (2.23) $M = 1$ with $a_0 = 2$ and $a_1 = -1$. This gives $\alpha = -1$ and $c_1 = 2$ in (2.24) and so the zeros of $\frac{1}{2}(z^2 - 4z + 1)$ must be found. These are $2 \pm \sqrt{3}$, so setting $r_1 = 2 + \sqrt{3}$, (2.25) gives:

$$\mathcal{L}(\omega) = \frac{1}{2}\left((1+\sqrt{3}) + (1-\sqrt{3})e^{-i\omega}\right) \ ,$$

so that $$H(\omega) = \frac{1}{8}\left((1+\sqrt{3}) + (3+\sqrt{3})e^{-i\omega} + (3-\sqrt{3})e^{-2i\omega} + (1-\sqrt{3})e^{-3i\omega}\right) \ .$$

Therefore, by comparison with (2.15):

$$h_0 = \sqrt{2}(1+\sqrt{3})/8 \approx \quad 0.4830$$

$$h_1 = \sqrt{2}(3+\sqrt{3})/8 \approx \quad 0.8365$$

$$h_2 = \sqrt{2}(3-\sqrt{3})/8 \approx \quad 0.2241$$

$$h_3 = \sqrt{2}(1-\sqrt{3})/8 \approx -0.1294 \ .$$

Figure 2.2 displays the scaling functions and mother wavelets for the extremal phase families with $N=1,2,4$ and the least asymmetric family with $N=4$. Notice that the least asymmetric scaling function with $N=4$ is clearly closer to being symmetric than the extremal phase wavelet with $N=4$.

Evaluating the Daubechies Wavelets

The families of wavelets defined above have no closed form formula, except for the Haar wavelet. Although for most applications it is not necessary to evaluate the wavelet, there are some occasions when this is important. The `WaveThresh` program `draw.default`, which was used to construct figure 2.2,uses the *cascade* algorithm, see for example section 6.5 of Daubechies (1992). Essentially this involves taking the inverse DWT of a wavelet series which has only one non-zero coefficient, in a carefully chosen position.

A more flexible method for evaluating $\phi(x)$ for a particular value of x is the *local pyramidal algorithm* of Daubechies & Lagarias (1992). Consider a Daubechies wavelet basis with filter number N. Then ϕ has support $[0, 2N-1]$. Let $x \in (0,1)$ and let $dyad(x) = \{\delta_1, \delta_2, \dots, \delta_n, \dots\}$ be the set of 0-1 digits in the dyadic representation of x:

$$x = \sum_{j=1}^{\infty} \delta_j 2^{-j} .$$

The subset of the first n digits from $dyad(x)$ is denoted by $dyad(x,n) = \{\delta_1, \delta_2, \dots, \delta_n\}$. Let $\boldsymbol{h} = (h_0, h_1, \dots, h_{2N-1})$ be the vector of wavelet filter coefficients. Define two $(2N-1) \times (2N-1)$ matrices as

$$\left.\begin{aligned} T_0 &= \sqrt{2}(h_{2i-j-1}) \\ T_1 &= \sqrt{2}(h_{2i-j}) \end{aligned}\right\} 1 \leqslant i,j \leqslant 2N-1 .$$

Then Daubechies & Lagarias (1992) show that

$$\lim_{n\to\infty} T_{\delta_1} \cdot T_{\delta_2} \cdot \ldots \cdot T_{\delta_n} = \begin{bmatrix} \phi(x) & \phi(x) & \cdots & \phi(x) \\ \phi(x+1) & \phi(x+1) & \cdots & \phi(x+1) \\ \vdots & \vdots & & \vdots \\ \phi(x+2N-2) & \phi(x+2N-2) & \cdots & \phi(x+2N-2) \end{bmatrix} \tag{2.26}$$

To evaluate ϕ for values of x other than $(0,1)$, put the decimal part of x into the algorithm and read off the appropriate row from (2.26). Note that the algorithm yields $\phi_{0k}(x)$ for all values of k for which it is non-zero.

Pinheiro & Vidakovic (1995) suggest the following variant of this result. Define the $2N-1$ vector $\boldsymbol{u}$ as:

$$\boldsymbol{u}(x) = \left\{(-1)^{[2x]-i} h_{1+i-[2x]} \quad , \ i = 0, \dots, 2N-2\right\} ,$$

where $[x]$ is the integer part of x. If for some i the index $i+1-[2x]$ is < 0 or $> 2N-1$ then the corresponding component of $\boldsymbol{u}$ is set to zero. Define the vector $\boldsymbol{v}$ as

$$\boldsymbol{v}(x,n) = \frac{\sqrt{2}}{2N-1} \mathbf{1}^{\mathrm{T}} \prod_{i \in dyad(\{2x\},n)} T_i ,$$

where $\{x\}$ is the decimal part of x (so $[x]+\{x\}=x$). Then

$$\psi(x) = \lim_{n\to\infty} \boldsymbol{u}(x)^{\mathrm{T}}\boldsymbol{v}(x,n) \ .$$

This result is a simple rearrangement of (2.17), where values of the ϕ_{1k} are obtained by averaging the columns of (2.26). Unfortunately there are errors, possibly typographical, in the presentation of this result in Pinheiro & Vidakovic (1995) and in Vidakovic (1999).

2.1.5 Besov Spaces

Besov spaces are extensively used in the wavelet literature when considering the theoretical performance of wavelet estimators. These can be characterised in several different ways. Donoho *et al.* (1996) define the spaces in terms of wavelet coefficients, as follows. Suppose ϕ has regularity $r > \sigma$, and recall that P_j is the associated projection operator onto V_j and $Q_j = P_{j+1} - P_j$. Besov spaces depend on three parameters $\sigma > 0$, $1 \leqslant p \leqslant \infty$ and $1 \leqslant q \leqslant \infty$ and are denoted B^{σ}_{pq}. A function $f \in B^{\sigma}_{pq}$ if and only if the norm

$$J^{\sigma}_{pq}(f) = \|P_0(f)\|_{L^p(\mathbb{R})} + \left(\sum_{j\geqslant 0}(2^{j\sigma}\|Q_j f\|_{L^p(\mathbb{R})})^q\right)^{1/q} < \infty \ ,$$

with the usual modification for $q=\infty$. Using now the decomposition of f:

$$\begin{aligned} P_0 f &= \sum_{k\in\mathbb{Z}} c_{0k}\phi_{0k} \ , \\ Q_j f &= \sum_{k\in\mathbb{Z}} d_{jk}\psi_{jk} \ , \end{aligned}$$

it can also be said that $f \in B^{\sigma}_{pq}$ if and only if the equivalent norm

$$J'^{\sigma}_{pq}(f) = \|c_{0\cdot}\|_{l_p} + \left(\sum_{j\geqslant 0}(2^{j(\sigma+1/2-1/p)}\|d_{j\cdot}\|_{l_p})^q\right)^{1/q} < \infty \ , \tag{2.27}$$

where

$$\|d_{j\cdot}\|_{l_p} = \left(\sum_{k\in\mathbb{Z}} |d_{jk}|^p\right)^{1/p} \ .$$

Abusing the notation slightly, write $\|\boldsymbol{d}\|^{\sigma}_{pq}$ for the above norm applied to the coefficients $\big((c_{jk}),(d_{jk})\big)$. Also, set $B^{\sigma}_{pq}(M) = \{\boldsymbol{d} : \|\boldsymbol{d}\|^{\sigma}_{pq} \leqslant M\}$.

Abramovich *et al.* (1998) present a definition of B^σ_{pq} in terms of the smoothness of f. They note the equivalence of this to the above definition and that σ measures the number of derivatives, where the existence of derivatives is required in an L^p sense, while q provides a further, finer, gradation. See also Abramovich *et al.* (2000a). For a more detailed, mathematical study see, for example, Triebel (1992).

2.2 Wavelet Transforms

2.2.1 Wavelets as Basis Functions

Using (2.3) and (2.9), and letting $j \to \infty$, it can be seen that, for some $L \in \mathbb{Z}$, $\{\phi_{Lk}, \psi_{jl} : k, l \in \mathbb{Z},\ j \in \mathbb{J}_L\}$ forms an orthonormal basis for $L^2(\mathbb{R})$, where

$$\mathbb{J}_L = \{m \in \mathbb{Z} : m \geqslant L\}\ . \tag{2.28}$$

Using this basis, the wavelet representation of a function f is:

$$f(x) \sim \sum_{k\in\mathbb{Z}} c_{Lk}\phi_{Lk}(x) + \sum_{j\in\mathbb{J}_L}\sum_{k\in\mathbb{Z}} d_{jk}\psi_{jk}(x)\ , \tag{2.29}$$

where the wavelet coefficients are

$$d_{jk} = \int_{-\infty}^{\infty} \psi_{jk}(x) f(x) dx \tag{2.30}$$

and the scaling function coefficients are given in (2.7).

The first term in (2.29) is the approximation $P_L f$ of f at resolution 2^L. From (2.8) and since $\{\psi_{jk} : k \in \mathbb{Z}\}$ is a basis for W_j, $\sum_{k\in\mathbb{Z}} d_{jk}\psi_{jk}(x)$ is the difference between $P_j f$ and the finer resolution approximation $P_{j+1} f$. So for each value of j, the second term in (2.29) adds another level of detail into the representation.

There are two properties of the wavelet coefficients that will be important later on. Firstly, because of the vanishing moments property (2.21), if f is smooth the wavelet representation is very economical because there will be few non-zero wavelet coefficients d_{jk}. Also, because wavelets are localised in time and scale, a discontinuity, or other high frequency feature, in f will only result in large wavelet coefficients for values of k corresponding to the location of the feature. Many functions can therefore be represented by a small number of wavelet coefficients. This property explains the application of wavelets to data compression and is also important in statistical applications.

The second important property of the coefficients is that there is a recursive relationship between scaling function and wavelet coefficients at successive levels. From (2.7) and (2.13):

$$\begin{aligned} c_{j-1,k} &= \int f(x) \left\{ \sum_n h_{n-2k} \phi_{jn}(x) \right\} dx \\ &= \sum_n h_{n-2k} \left\{ \int f(x) \phi_{jn}(x)\, dx \right\} \\ &= \sum_n h_{n-2k} c_{jn} \ . \end{aligned} \tag{2.31}$$

Similarly, from (2.30) and (2.18):

$$d_{j-1,k} = \sum_n g_{n-2k} c_{jn} \ . \tag{2.32}$$

Note that the *Parseval identity* relates the wavelet and scaling function coefficients to the "energy" of f:

$$\int |f(x)|^2 \, dx = \sum_k |c_{Lk}|^2 + \sum_{j=L}^{\infty} \sum_k |d_{jk}|^2 \ . \tag{2.33}$$

2.2.2 Discrete Wavelet Transform

The wavelet representation (2.29) is of limited practical use, because functions are almost always observed at only a finite number of discrete time points, so a discrete analogue is required. The DWT is due to Mallat (1989b), but similar algorithms without the wavelet motivation were developed earlier in the filtering literature, for example the subband coders of Smith & Barnwell (1986). Vaidyanathan (1990) provides a survey of many filtering methods, including subband coders and Smith & Eddins (1990) consider subband coding for images.

Suppose the function f is observed at a dyadic number of equally spaced time points, so the observations are $f(t_i)$ for $i = 0, \dots, 2^J - 1$ for some $J \in \mathbb{N}$, where $t_i = t_0 + i\delta$ for some $t_0, \delta \in \mathbb{R}$. A function $\bar{f}$ can be constructed:

$$\bar{f}(x) = \sum_k \widetilde{c}_{Jk} \phi_{Jk}(x) \ ,$$

where $\widetilde{c}_{Jk} = f(t_k)$. This function is clearly an element of V_J and the whole multiresolution analysis apparatus can be used on it. Delyon & Juditski (1995) note that since

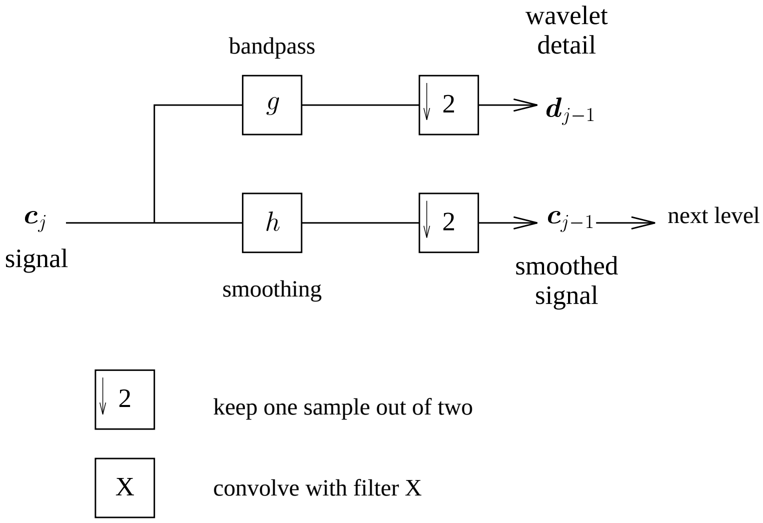


Figure 2.3: Decomposition of $\boldsymbol{c}_j$ into (smoothed) scaling function coefficients $\boldsymbol{c}_{j-1}$ and (detail) wavelet coefficients $\boldsymbol{d}_{j-1}$.

$2^{J/2}\phi_{Jk}$ is a function with integral 1 that peaks severely on $2^{-J}k$, $2^{J/2}c_{Jk} \approx \widetilde{c}_{Jk}$, and go on to suggest other methods of estimating the coefficients c_{Jk}.

The formulae (2.31) and (2.32) can now be applied recursively to the $\widetilde{c}_{Jk}$ to find the coefficients $\widetilde{c}_{0k}$ and $\widetilde{d}_{jk}$ for $0 \leqslant j \leqslant J-1$. These coefficients will be referred to as the *empirical* scaling function and wavelet coefficients. This is often thought of as a filtering operation, with a high pass (or bandpass) filter g and a low pass (or smoothing) filter h. Each stage of the transform consists of a convolution of the $\widetilde{c}_{jk}$ with each of the filters, follow by a dyadic decimation (every other element is discarded) to give the lower resolution coefficients $\widetilde{c}_{j-1,k}$ and $\widetilde{d}_{j-1,k}$. This is illustrated in figure 2.3. Figures 2.3 & 2.4 originated in Mallat (1989b). The versions displayed here are reproduced with permission from Nason & Silverman (1994).

Mallat (1989b) also derives a reconstruction algorithm as follows. Note that at each level of the decomposition coarser scale coefficients are obtained from finer ones as illustrated by:

$$\begin{aligned}\sum_k \widetilde{c}_{j+1,k}\phi_{j+1,k} &= P_{j+1}\bar{f}\\ &= P_j\bar{f} + Q_j\bar{f}\\ &= \sum_k \widetilde{c}_{jk}\phi_{jk} + \sum_k \widetilde{d}_{jk}\psi_{jk}\ ,\end{aligned}$$

where Q_j is the orthogonal projection operator onto W_j. Hence, because of the mutual

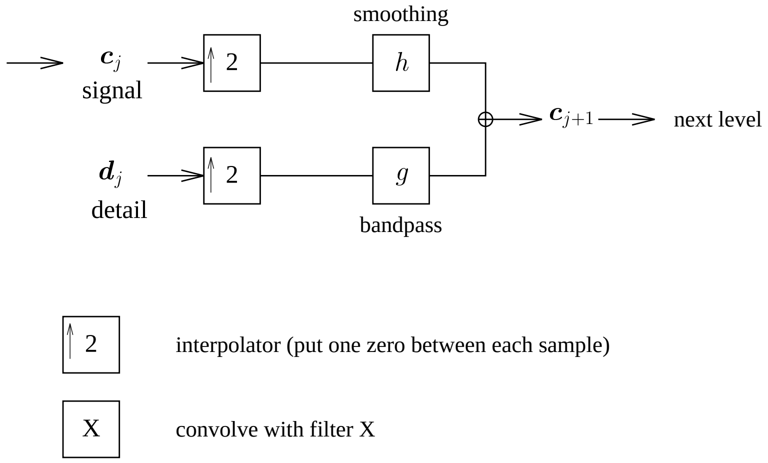


Figure 2.4: Reconstruction of $\boldsymbol{c}_{j+1}$ from $\boldsymbol{c}_j$ and $\boldsymbol{d}_j$.

orthogonality of the ϕ_{jk}:

$$\begin{aligned}
\widetilde{c}_{j+1,n} &= \langle \phi_{j+1,n}, P_{j+1}\bar{f} \rangle \\
&= \sum_k \widetilde{c}_{jk} \langle \phi_{j+1,n}, \phi_{jk} \rangle + \sum_k \widetilde{d}_{jk} \langle \phi_{j+1,n}, \psi_{jk} \rangle \\
&= \sum_k h_{n-2k}\, \widetilde{c}_{jk} + \sum_k g_{n-2k}\, \widetilde{d}_{jk} \ ,
\end{aligned}$$

since we know from (2.13) and (2.18) that

$$\begin{aligned}
\langle \phi_{j+1,n}, \phi_{jk} \rangle &= h_{n-2k} \ , \\
\langle \phi_{j+1,n}, \psi_{jk} \rangle &= g_{n-2k} \ .
\end{aligned}$$

This gives Mallat's reconstruction formulae. This filtering operation is illustrated in figure 2.4.

The DWT is implemented in the `S-Plus` package `WaveThresh`, as well as several other packages. The `WaveThresh` software was used for this thesis, and is described in Nason & Silverman (1994).

Boundary Conditions

Consider the convolutions in (2.31) and (2.32). The filters g and h usually have the same length. This is certainly true for the Daubechies wavelets, which have lengths $N_h = N_g = 2N$, where N is the filter number, introduced on page 10. It is very important to consider what happens when the filters overlap with the ends of the vector of coefficients. It is usually assumed that the function f satisfies some type of boundary conditions and the algorithm is then adapted to take account of this. The `WaveThresh` package can do this in two ways. *Symmetric* boundary conditions reflect the ends of the vector. *Periodic* boundary conditions wrap round the last few coefficients to the start of the vector. Other methods are possible, such as putting in zeros beyond the range of the vector of coefficients.

2.2.3 Non-Decimated Wavelet Transform

The non-decimated wavelet transform (NWT), which is sometimes called the stationary wavelet transform, was introduced in a statistical context by Nason & Silverman (1995). Recall that the DWT consists of a convolution followed by a dyadic decimation. In the NWT the decimation stage is omitted, resulting in (roughly) the same number of coefficients at each level. Note that in the decimation stage of the DWT, usually every even member of the vector is retained. However, it is equally possible to retain every odd member, so there are two possible sets of coefficients at each stage. Since this is true at every level, there are 2^J different, but equally valid, sets of DWT coefficients. The NWT contains all these coefficients.

2.2.4 Lifting

Sweldens (1996) gives an introduction to the *lifting scheme*. Lifting provides a faster alternative to the DWT by exploiting the connection between the filters h and g. It also allows the construction of *second generation* wavelets, which are not necessarily dilates and translates of a mother function (see Sweldens (1998)). It can be used to define wavelet transforms on irregularly sampled data, a subject which is dealt with in more detail in Daubechies *et al.* (1999). The statistical applications of lifting have not yet been well studied.

2.3 Wavelet Shrinkage and Thresholding

Wavelet shrinkage was introduced to the literature by Donoho & Johnstone (1994). The situation is the usual regression problem of trying to recover a signal in the presence of noise. Suppose n noisy samples of a function f are available:

$$y_i = f(x_i) + \varepsilon_i \ , \qquad i = 1, \dots , n \ , \tag{2.34}$$

with $x_i = (i-1)/n$, ε_i i.i.d. $N(0, \sigma^2)$ and n a power of two. The aim is to estimate the unknown vector $\boldsymbol{f} = (f(x_1), \dots , f(x_n))$.

The method of wavelet shrinkage consists of three main steps. First the data is decomposed using the DWT. Then the wavelet coefficients are modified in some way. Finally the inverse transform is applied to the modified coefficients to give an estimate of $\boldsymbol{f}$.

The coefficient modification can be carried out in many ways. A popular method, used in Donoho & Johnstone (1994) is *thresholding*. This is motivated by two factors. Firstly, as discussed in §2.2.1, the wavelet representation of many kinds of function is very economical, so it is expected that there will be few large true wavelet coefficients and these will be concentrated near the areas of major spatial activity (e.g. discontinuities). Secondly, under the noise model, all wavelet coefficients will be contaminated by the noise. In fact, since the wavelet transform is orthogonal, if the ε_i are assumed to be independent Gaussian noise, then wavelet coefficients will also be contaminated with independent Gaussian noise. So, in this case, the empirical wavelet coefficients can be written as

$$\widetilde{d}_{jk} = d_{jk} + \varepsilon_{jk} \ ,$$

where the ε_{jk} are independent Gaussian noise, and relatively few of the d_{jk} are expected to be non zero. Thresholding attempts to determine which of the $\widetilde{d}_{jk}$ are just noise, and which are really non zero.

Donoho & Johnstone (1994) define two types of thresholding. *Hard* thresholding removes all the coefficients that are smaller than the *threshold* τ:

$$\widehat{d}_{jk} = T_{\text{hard}}(\widetilde{d}_{jk}; \tau) = \widetilde{d}_{jk} I(|\widetilde{d}_{jk}| > \tau) \ . \tag{2.35}$$

Soft thresholding takes account of the fact that the true function coefficients are also contaminated with noise, so the empirical coefficients that are larger than the threshold are shrunk by τ:

$$\widehat{d}_{jk} = T_{\text{soft}}(\widetilde{d}_{jk}; \tau) = \text{sgn}(\widetilde{d}_{jk})(|\widetilde{d}_{jk}| - \tau) I(|\widetilde{d}_{jk}| > \tau) \ . \tag{2.36}$$

Universal Threshold Threshold Selection

There are many different possibilities for choosing the threshold level τ. Donoho & Johnstone (1994) propose the *universal threshold*:

$$\tau^u = \sqrt{2\log n}\;\sigma\ ,$$

where the noise variance σ usually needs to be estimated. Donoho & Johnstone (1994) suggest using the MAD of the finest level coefficients, divided by 0.6745. The rationale for this choice of threshold is that if $\{Z_i\}_{i=1}^n$ is a white noise sequence, i.i.d. $N(0,1)$, then as $n \to \infty$

$$P\left(\max_i |Z_i| > \sqrt{2\log n}\right) \to 0\ .$$

This means that, with high probability, all the pure noise coefficients will be thresholded to zero. This results in reconstructions that have a "noise-free" character, but a consequence of this is that they will tend to be over smoothed.

Donoho *et al.* (1995) study the performance of this threshold selection method, in conjunction with a soft threshold. They conclude that the resulting function estimate is asymptotically minimax for a wide variety of loss functions l and functions f belonging to a wide range of smoothness classes (e.g. B^σ_{pq}). The minimax problem for non-parametric regression is described in section 2.2 of Donoho *et al.* (1995). More importantly they show that the wavelet estimator is nearly optimal for a wide variety of objectives.

SURE Threshold Selection

Another method, suggested by Donoho & Johnstone (1995), is based on Stein's unbiased risk estimation. Stein (1981) introduces, as Theorem 1, the following method for estimating the risk of an estimate of the mean of a multivariate normal distribution. Let $\mathbf{x} \sim N_p(\mu, \mathbf{1})$ be multivariate normal observations with mean vector μ. Suppose we have a fixed estimator $\widehat{\mu}(\mathbf{x})$ of μ that can be written as:

$$\widehat{\mu}(\mathbf{x}) = \mathbf{x} + \mathbf{g}(\mathbf{x})\ ,$$

where $\mathbf{g} = (g_i)_{i=1}^p$ is a function from $\mathbb{R}^p$ to $\mathbb{R}^p$. When $\mathbf{g}(\mathbf{x})$ is weakly differentiable, Stein shows that

$$E_\mu \|\widehat{\mu}(\mathbf{x}) - \mu\|^2 = p + E_\mu \left\{ \|\mathbf{g}(\mathbf{x})\|^2 + 2\nabla \cdot \mathbf{g}(\mathbf{x}) \right\} ,$$

where
$$\nabla \cdot \mathbf{g} \equiv \sum_i \frac{\partial}{\partial x_i} g_i .$$

Now, notice that the soft threshold can be written as:

$$\widehat{\mu}(\mathbf{x}) = \mathbf{x} - \operatorname{sgn}(\mathbf{x}) \min(|\mathbf{x}|, \tau) .$$

So, using Stein's result, the quantity

$$\mathrm{SURE}(\tau; \mathbf{x}) = p - 2 \cdot \#\{i : |x_i| \leqslant \tau\} + \sum_{i=1}^{p} (|x_i| \wedge \tau)^2$$

is an unbiased estimate of the risk $E_\mu \|\widehat{\mu}(\mathbf{x}) - \mu\|^2$. Donoho & Johnstone (1995) choose the threshold for each level, τ_j^s to minimise $\mathrm{SURE}(\tau; \boldsymbol{d}_j)$. They show that the threshold is guaranteed to occur at one of the raw coefficients, and that minimisation is computationally straightforward. In fact their SureShrink procedure uses a hybrid method: the universal threshold is used on those levels where there is little evidence of anything other than noise, otherwise the SURE threshold is used. This is because in such situations the noise will swamp the information contributed to SURE by the few non zero signal coordinates.

Threshold Selection by Cross-validation

Nason (1996) considers a different approach altogether and uses cross-validation methods to choose the threshold. The aim is to choose τ to minimise the mean integrated square error (MISE) between the true function and the wavelet shrinkage estimator $\hat{f}_\tau(x)$:

$$M(\hat{f}_\tau) = E\left[\int \left\{ \hat{f}_\tau(x) - f(x) \right\}^2 \mathrm{d}x \right] . \tag{2.37}$$

Since f is unknown, an estimate of M must be devised. Two methods of cross-validation are considered.

Firstly twofold cross-validation, in which half the data points are used to form a wavelet shrinkage estimator using a particular threshold. The remaining points are used to estimate the MISE at that threshold, by comparing them with the estimator. The error is then minimised over the threshold. First take all the evenly indexed y_i and form the shrinkage estimator $\hat{f}^E_\tau$ using a particular threshold τ. This is interpolated, so that it can be compared with the left out, noisy data:

$$\bar{f}^E_{\tau,j} = \frac{1}{2}\left(\hat{f}^E_{\tau,j+1} + \hat{f}^E_{\tau,j}\right) \qquad j = 1, \ldots, n/2\,,$$

where periodicity is assumed at the end points. This repeated with the odd indexed points to give the interpolant $\bar{f}^O_\tau$. Then the following estimate of M is minimised over τ:

$$\hat{M}(\tau) = \sum_{j=1}^{n/2} \left\{ \left(\bar{f}^E_{\tau,j} - y_{2j+1}\right)^2 + \left(\bar{f}^O_{\tau,j} - y_{2j}\right)^2 \right\} \,. \tag{2.38}$$

Since this will yield a threshold based on $n/2$ points, a correction is proposed to give a threshold for n points.

Nason (1996) also suggests a computationally inefficient leave-one-out cross-validation method. Each point y_i for $1 < i < n$ is left out in turn. The remaining points are split into two groups:

$$G_L = \{y_1, \ldots, y_{i-1}\}\,,$$
$$G_R = \{y_{i+1}, \ldots, y_n\}\,.$$

These groups are reflected and then extended to dyadic length and wavelet estimators $\hat{f}_{L,\tau}$ and $\hat{f}_{R,\tau}$ formed. The removed point is predicted by averaging the the rightmost point of $\hat{f}_{L,\tau}$ and the leftmost point of $\hat{f}_{R,\tau}$ to give $\hat{y}_{\tau,-i}$. The estimated MISE to be minimised over τ is:

$$\hat{M}(\tau) = \sum_{i=2}^{n-1} (y_i - \hat{y}_{\tau,-i})^2\,.$$

It is well known that cross-validation methods do not work well with serially correlated data. See for example Diggle (1990).

Thresholding Correlated Data

So far, the methods of thresholding discussed assume that the noise process is uncorrelated. Johnstone & Silverman (1997) consider the case when the noise is correlated, but stationary. They present two important properties of the wavelet transform as applied to correlated data. Firstly, the autocorrelation of the d_{jk} within each level tends to die away rapidly, and secondly there tends to be little or no correlation between levels. So the wavelet transform acts to remove the correlation from the data. Because of this they propose using a level-dependent threshold. It turns out that the SURE quantity remains unbiased for the risk, even under a correlated model, so it is sensible to use this.

Lorentz Thresholding

Goel & Vidakovic (1995) introduce Lorentz thresholding, which is hard thresholding with a threshold selection procedure based on a consideration of the energies contained in the coefficients. Only those coefficients whose energies are κ times the average energy are retained. The thresholded coefficients can be written as:

$$\widehat{d}_{jk} = \widetilde{d}_{jk} \cdot I(\widetilde{d}^2_{jk} > \kappa\overline{\widetilde{d}^2}) \ ,$$

where

$$\overline{\widetilde{d}^2} = \frac{1}{m}\sum_{j,k}\widetilde{d}^2_{jk} \ ,$$

with m being the total number of coefficients to which the thresholding procedure is applied. Usually $m = 2^J - 2^{j_0}$, where J is the resolution level of the original data (so $n = 2^J$) and j_0 is the coarsest level that is thresholded.

Restriction to Integer Resolution Level

Hall & Nason (1997) discuss the effect of restricting the resolution level J to an integer. They argue that this restriction does not allow as much adaptivity to variations in a "smooth" target function as kernel methods do, and note that it can be likened to demanding that the smoothing parameter of a kernel estimator be a power of two. The restriction of integer J is necessary for the use of Mallat's DWT algorithm. Although Hall & Nason note that the computational effort of smoothing with non-integer J is computationally insignificant in the density estimation case, there are no details of how this should be done and the extension to regression is not discussed.

2.4 Wavelet Shrinkage for Unequally Spaced Data

Suppose data are observed as in (2.34), with the locations x_i not equally spaced. Without loss of generality, suppose the data are observed on the interval $[\,0, 1\,]$. Kovac & Silverman (2000) propose dealing with this situation by interpolating the data onto an evenly spaced grid of dyadic length. They define a grid of length 2^J, for some positive integer J, by:

$$\check{x}_k = (k + \tfrac{1}{2})2^{-J} \ , \qquad 0 \leqslant k \leqslant 2^J - 1 \ .$$

The choice of the parameter J is not discussed in much detail by Kovac & Silverman. Their only suggestion is to choose J to be the smallest value that gives more grid points than data points:

$$J = \min\{j \in \mathbb{Z} : 2^j > n\} \ .$$

Linear interpolation is then used to give new "data" $\check{y}_k$ at the grid points $\check{x}_k$. The $\check{y}_k$ are calculated by evaluating the straight line that interpolates the nearest data points on the left and on the right at $\check{x}_k$. Outside the range of the data, lines of gradient zero are fitted. So, the interpolated data can be written:

$$\check{y}_k = \begin{cases} y_1 & \text{if } \check{x}_k \leqslant x_1 \\ y_n & \text{if } \check{x}_k \geqslant x_n \\ \alpha y_i + (1-\alpha) y_{i+1} & \text{if } x_i \leqslant \check{x}_k \leqslant x_{i+1} \ , \end{cases} \tag{2.39}$$

where α depends only on x_i, x_{i+1} and $\check{x}_k$. In fact,

$$\alpha = (x_{i+1} - \check{x}_k)/(x_{i+1} - x_i) \ .$$

For an example, see figure 2.5. There are five data points and $J = 3$ is chosen, giving eight grid points.

There are a dyadic number of evenly spaced grid points, so the DWT can be used on the interpolated data, which will yield empirical wavelet coefficients $\widetilde{d}_{jk}$. However, these coefficients cannot be thresholded using standard methods. Most thresholding methods assume that the data are independent and identically distributed. Whilst this may be true of the original data y_i it will certainly not be true of the interpolated data $\check{y}_i$. Most importantly, the wavelet coefficients can no longer be assumed to have equal variances.

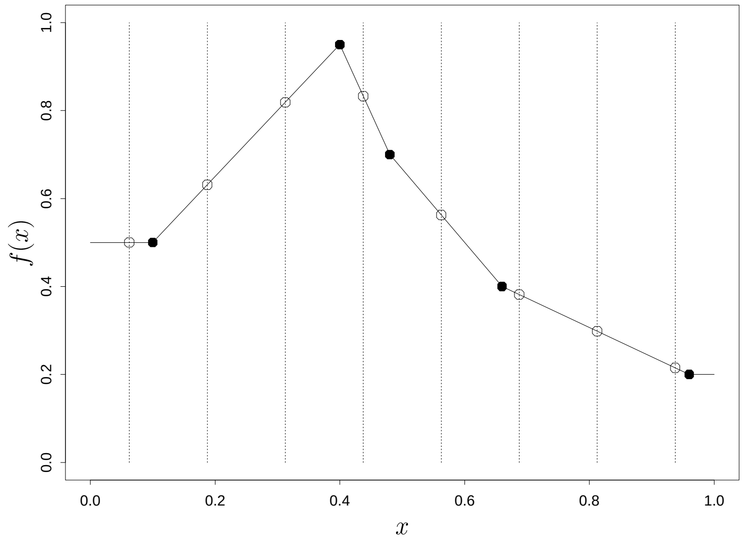


Figure 2.5: Linear interpolation of unevenly spaced data. Closed circles represent the original data and open circles the interpolated data.

Thresholding Heteroscedastic Coefficients

A natural extension of standard thresholding methods to the case of heteroscedastic coefficients is to use a different threshold for each coefficient. Kovac & Silverman (2000) propose using such coordinate-wise thresholds here. The extension of the hard threshold will look like:

$$\widehat{d}_{jk} = \widetilde{d}_{jk} I(|\widetilde{d}_{jk}| > \tau_{jk}) \ .$$

The thresholds τ_{jk} can be chosen by using extensions of standard techniques. If the variances of all the coefficients are known, then we can use an extension of the universal threshold, which chooses the threshold to be proportional to the standard deviations σ_{jk} of the empirical coefficients:

$$\tau^u_{jk} = \sqrt{2 \log n}\ \sigma_{jk} \ , \tag{2.40}$$

where $\sigma^2_{jk} = \mathrm{Var}(\widetilde{d}_{jk})$.

In the case of unequally spaced data, if the original data y_i are assumed to be independent and identically distributed, with variance σ^2, then:

$$\sigma^2_{jk} = \sigma^2 \gamma_{jk} \ ,$$

where the variance factors γ_{jk} can be calculated, as described in the next section. Usually the noise level σ is unknown. By analogy with the standard method for equally spaced data, Kovac & Silverman (2000) suggest estimating σ by

$$\widehat{\sigma} = \mathrm{MAD}\left\{\widetilde{d}_{J-1,k}/\sqrt{\gamma_{J-1,k}} : \gamma_{J-1,k} > 0.0001\right\}/0.6745 \ . \tag{2.41}$$

They then set

$$\widehat{\sigma^2_{jk}} = \widehat{\sigma}^2\gamma_{jk} \ .$$

Calculation of Variances

First consider calculating the variance factors of the interpolated data. The linear interpolation can be written as a matrix operation:

$$\check{\boldsymbol{y}} = R\boldsymbol{y}$$

where R is $2^J \times n$, $\check{\boldsymbol{y}}$ is the vector of interpolated values and $\boldsymbol{y}$ the vector of observed values. Note that, from (2.39), each row of R will have only two non-zero entries (or one if the corresponding grid point is outside the range of the data) and these will sum to one.

Using standard results, and assuming that the original data has covariance matrix $\sigma^2 I$, it is easy to see that the covariance matrix of the interpolated data is given by:

$$\mathrm{Var}(\check{\boldsymbol{y}}) = \sigma^2 RR^{\mathrm{T}} \ .$$

Now consider how to calculate the variances of the individual wavelet coefficients. Since the DWT is linear, it can be written in terms of matrices, so the σ^2_{jk} can easily be found by repeated matrix multiplications. However, this will be quite slow, as matrix multiplication is an $O(n^2)$ operation.

Consider for a moment the structure of $\Gamma = RR^{\mathrm{T}}$. Since the non-zero entries of each row of R are consecutive, and the entries on one row are always directly below or to the right of the entries on the row above, it is clear that R, and hence Γ has a band structure. In the example in Figure 2.5 the structure of R is:

$$R = \begin{pmatrix} * & 0 & 0 & 0 & 0 \\ * & * & 0 & 0 & 0 \\ * & * & 0 & 0 & 0 \\ 0 & * & * & 0 & 0 \\ 0 & 0 & * & * & 0 \\ 0 & 0 & 0 & * & * \\ 0 & 0 & 0 & * & * \\ 0 & 0 & 0 & * & * \end{pmatrix},$$

and the structure of RR^{T} is:

$$RR^{\mathrm{T}} = \begin{pmatrix} * & * & * & 0 & 0 & 0 & 0 & 0 \\ * & * & * & * & 0 & 0 & 0 & 0 \\ * & * & * & * & 0 & 0 & 0 & 0 \\ 0 & * & * & * & * & 0 & 0 & 0 \\ 0 & 0 & 0 & * & * & * & * & * \\ 0 & 0 & 0 & 0 & * & * & * & * \\ 0 & 0 & 0 & 0 & * & * & * & * \\ 0 & 0 & 0 & 0 & * & * & * & * \end{pmatrix}.$$

The asterisks represent the non-zero entries.

Kovac & Silverman (2000) develop a fast algorithm for such a case. Their algorithm for calculating the σ_{jk}^2 when RR^{T} is a *band matrix*, is linear in both the bandwidth and the number of data points. As well as exploiting the band structure, they note that at each level from the covariance matrix of the c_{jk} the full covariance matrix of the $c_{j-1,k}$ needs to be calculated, but only the *variances* of the wavelet coefficients $d_{j-1,k}$. The full covariance matrix of the $d_{j-1,k}$ is not required. Using this fact saves calculating numbers that are not needed.

Other Approaches

Pensky & Vidakovic (1998) propose using a linear wavelet estimator of the form $\sum_k \widehat{c}_{Jk}\phi_{Jk}(x)$ - i.e. they estimate a projection of the function on the multiresolution subspace V_J. The coefficients are estimated using a prestimator of the density of the locations x_i. They investigate mean squared error convergence rates. This includes some discussion of the value of J, but no practical method of choosing this parameter is proposed.

Sardy *et al.* (1999) investigate four schemes based on the Haar wavelet transform, all using piecewise constant interpolation. Their preferred method is isometric wavelets, which amounts to ignoring the irregular spacing and applying the standard DWT directly to the data. They note that these isometric wavelets are a special case of lifting and that they do not form an orthonormal set of functions. Some theoretical justification is provided in terms of a non-Euclidean norm and it is noted that the isometric wavelets produce noisy coefficients that are i.i.d.. However, the interpretation of these coefficients is not clear. Their relationship to the true (ordinary Haar) wavelet coefficients of the function is not discussed. Also, it is not clear whether the isometric wavelets form a basis of $L^2(\mathbb{R})$, despite not being orthonormal. They discuss very briefly the extension of this idea to two dimensions.

2.5 Wavelets in Two Dimensions

2.5.1 Multiresolution Analysis in Two Dimensions

Mallat (1989b) notes that multiresolution analysis and the wavelet model can readily be extended to two (or more) dimensions. Most of the literature relating to the 2D wavelet transform considers it in the context of image analysis, rather than surface estimation. The signal is now a function of two variables:

$$f(x, y) \in L^2(\mathbb{R}^2) \ .$$

Following the notation of Daubechies (1988), a multiresolution approximation of $L^2(\mathbb{R}^2)$ is a sequence $\mathbf{V}_j$ of subspaces of $L^2(\mathbb{R}^2)$ which satisfies a straightforward two-dimensional extension of the properties of the 1D MRA (2.1) – (2.5). Again, there is a scaling function $\Phi(x, y) \in \mathbf{V}_0$ such that $\{\Phi^j_{kl} : k, l \in \mathbb{Z}\}$ constitute an orthonormal basis for $\mathbf{V}_j$, where

$$\Phi^j_{kl} = 2^j \Phi(2^j x - k, 2^j y - l) \ . \tag{2.42}$$

In fact the particular case of separable multiresolution analysis is usually used, where $\mathbf{V}_j$ is taken to be the tensor product of V_j with itself:

$$\mathbf{V}_j = V_j \otimes V_j \ .$$

(So $g \in \mathbf{V}_j \implies g = g_1 g_2$ where $g_1, g_2 \in V_j$.) It can be shown that the scaling function can be written as

$$\Phi(x, y) = \phi(x)\phi(y) \ .$$

The orthogonal complement of $\mathbf{V}_j$ in $\mathbf{V}_{j+1}$ is denoted $\mathbf{W}_j$ (so that $\mathbf{V}_{j+1} = \mathbf{V}_j \oplus \mathbf{W}_j$). An expression for $\mathbf{W}_j$ can be derived as follows:

$$\begin{aligned}
\mathbf{V}_{j+1} &= V_{j+1} \otimes V_{j+1} \\
&= (V_j \oplus W_j) \otimes (V_j \oplus W_j) \\
&= \mathbf{V}_j \oplus [(V_j \otimes W_j) \oplus (W_j \otimes V_j) \oplus (W_j \otimes W_j)] \ .
\end{aligned}$$

From this it can be seen that an orthonormal basis for $\mathbf{W}_j$ is given by

$$\left\{ \Psi^{ij}_{kl} : i \in \{1, 2, 3\}, k, l \in \mathbb{Z} \right\} \ ,$$

where the two dimensional wavelets Ψ^1, Ψ^2, Ψ^3 are given by:

$$\begin{aligned}
\Psi^1(x, y) &= \phi(x)\psi(y) \\
\Psi^2(x, y) &= \psi(x)\phi(y) \\
\Psi^3(x, y) &= \psi(x)\psi(y)
\end{aligned}$$

and the dilations and translations are found using a formula analogous to (2.42).

2.5.2 The Discrete Wavelet Transform in Two Dimensions

For practical purposes an extension of the fast DWT to two dimensions is needed. Recall that Mallat's one dimensional algorithm requires a dyadic number of equally spaced data points. Mallat (1989b) also describes a two dimensional algorithm which requires data on an evenly spaced grid with the same, dyadic, number of grid points along each axis. Such evenly spaced data can be stored in a matrix:

$$z_{ij} = f(x_{ij}, y_{ij}) + \varepsilon_{ij} \ , \qquad \text{for } i = 1, \ldots, n \text{ and } j = 1, \ldots, n \ .$$

If the grid is aligned parallel to the axes then the grid points can be written:

$$x_{ij} = x_i \qquad y_{ij} = y_j \ ,$$

where $\boldsymbol{x}$ is a vector containing the x coordinates of the vertical grid lines and $\boldsymbol{y}$ is a vector containing the y coordinates of the horizontal grid lines.

To calculate the DWT of the data matrix Z, denote the data also by C^J, for some large J. At each level the decomposition to the next level is accomplished in two stages. First the rows of C^j are convolved with the one dimensional filters h and g, and alternate columns are retained. Then the columns of each resulting matrix are convolved with h and g, and alternate rows are retained. This gives the scaling function coefficients at the next level (C^{j-1}) and three types of wavelet coefficients ($D^{1,j-1}$, $D^{2,j-1}$, $D^{3,j-1}$). C^{j-1} is a coarser scale representation of the data and $D^{1,j-1}$, $D^{2,j-1}$ and $D^{3,j-1}$ give the detail of the "difference" between the two scales. Roughly speaking, $D^{1,j-1}$ contains horizontal detail, $D^{2,j-1}$ vertical detail and $D^{3,j-1}$ diagonal detail. The transform is illustrated in Figure 2.6. The formulae for calculating the coefficients are given below:

$$c_{kl}^{j-1} = \sum_m \sum_n h_{m-2k} h_{n-2l} c_{mn}^j \tag{2.43}$$

$$d_{kl}^{1,j-1} = \sum_m \sum_n g_{m-2k} h_{n-2l} c_{mn}^j \tag{2.44}$$

$$d_{kl}^{2,j-1} = \sum_m \sum_n h_{m-2k} g_{n-2l} c_{mn}^j \tag{2.45}$$

$$d_{kl}^{3,j-1} = \sum_m \sum_n g_{m-2k} g_{n-2l} c_{mn}^j \ . \tag{2.46}$$

The reconstruction algorithm is illustrated in Figure 2.7. Figures 2.6 and 2.7 are reproduced, with permission, from Nason & Silverman (1994).

The 2D DWT is implemented in `WaveThresh`. There is one apparent bug, in that the type 1 and type 2 wavelet coefficients are returned the wrong way round. This is corrected for in the coefficient plotting routines, so that they appear the right way round. Since the code is consistent within itself this does not cause any problems, unless the user decides to access the coefficients directly and modify them in different ways.

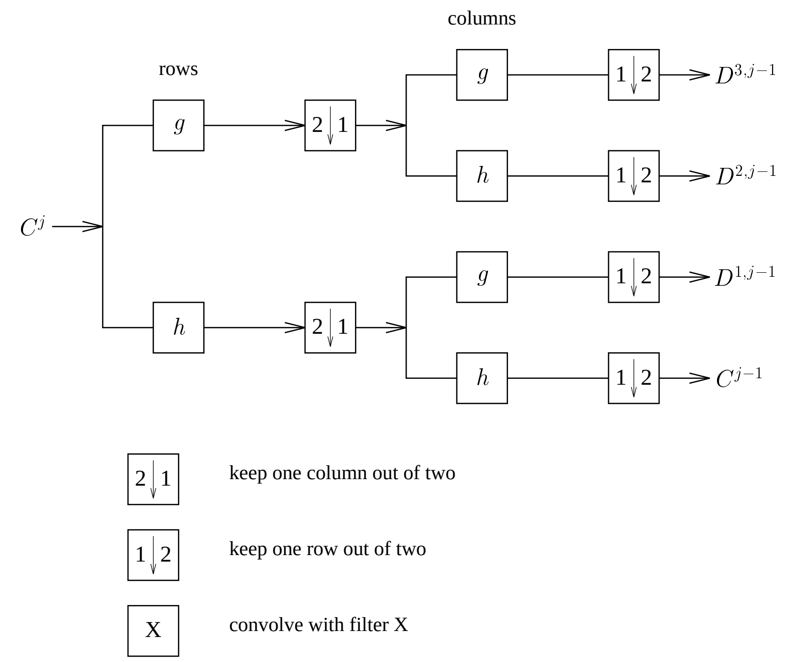


Figure 2.6: Two dimensional wavelet decomposition algorithm.

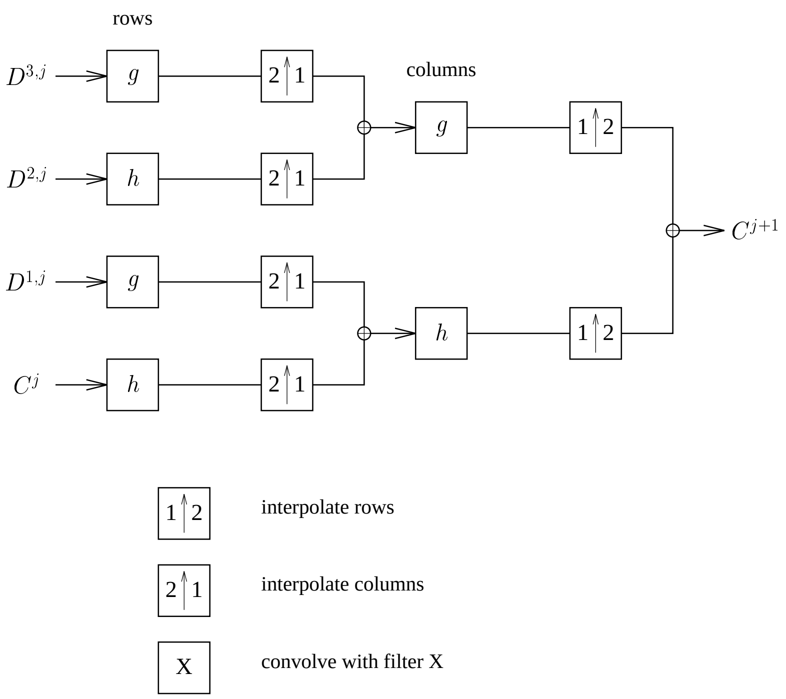


Figure 2.7: Two dimensional wavelet reconstruction algorithm.

2.6 Nonparametric Density Estimation

Given observations $x_1, \ldots, x_n$ from some density function f, density estimation aims to reconstruct f. There are several well established methods of doing this. Kernel methods and orthogonal series estimates will be considered here, as well as a brief description of maximum likelihood estimation.

The classic reference for density estimation is Silverman (1986), which concentrates mainly on kernel methods. Useful summaries can also be found in Bean & Tsokos (1980), Izenman (1991) and Tarter & Kronmal (1976). Scott (1992) provides a thorough discussion of density estimation in multidimensional settings.

2.6.1 Kernel Density Estimates

Kernel density estimation is probably the commonest, best understood method of density estimation. More recent discussions of kernel methods include Wand & Jones (1995) and Simonoff (1996).

The kernel estimator, based on a random sample $X_1, \ldots, X_n$, is

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right) ,$$

where h is the smoothing parameter (or window width, or bandwidth) and K is the kernel function, which satisfies

$$\int_{-\infty}^{\infty} K(x)\, dx = 1 .$$

If, in addition, K is everywhere non negative, so is itself a probability density function, then $\hat{f}$ will be a density function. Furthermore, $\hat{f}$ inherits the properties of differentiability and continuity from K. Often K is taken to be the density function of the $N(0,1)$ distribution, but there are many other possibilities.

Smoothing Parameter Selection

The choice of the smoothing parameter is crucial in density estimation, and most research has considered automatic methods for choosing the smoothing parameter. Although sometimes smoothing can be done by eye, automatic methods are useful as a starting point, and also invaluable for the inexperienced user. Several methods are discussed in section 3.4 of Silverman (1986). Amongst them, the following rule of thumb is derived:

$$h = 1.06 \min(s, R/1.34) n^{-1/5} , \tag{2.47}$$

where s is the sample standard deviation and R is the interquartile range. This choice minimises the approximate mean integrated square error for a normal kernel when the density from which the sample is drawn is $N(0, \sigma^2)$. Note that 1.34 is approximately the IQR of the standard normal, so that $R/1.34$ is a robust estimate of σ. This robust measure tends to oversmooth less than s for data exhibiting skewness or kurtosis, but tends to oversmooth more for multimodal distributions. Thus taking the minimum gives the best of both worlds.

The `S-Plus` function `density` performs kernel density estimation as above, but note that its smoothing parameter `width` is actually $4h$. The default bandwidth selection chooses (on the same scale as (2.47)):

$$h = \frac{\text{range}(x)}{8(1 + \log_2 n)} . \tag{2.48}$$

The theoretical justification for this choice is unclear and it is known to have poor practical performance.

Sheather & Jones (1991) develop another method of bandwidth selection, which is based on the work of Park & Marron (1990). Both methods minimise a kernel-based estimate of the MISE, but Sheather and Jones' version has superior theoretical performance (in terms of rates of convergence) and also computational gains in that the functions involved are better behaved, making the numerical solution to certain equations easier to compute. In addition, the practical performance has been shown by thorough simulation studies to be excellent. This selection procedure has been implemented by Venables & Ripley (1994) and the function `width.SJ` is available in library of functions that goes with the book.

Variable Bandwidth Kernel Density Estimates

There is a drawback to kernel density estimation, which becomes apparent when the method is applied to data from long tailed distributions. Since the smoothing parameter is fixed across the whole sample, it is likely that spurious, noisy, bumps will appear in the tails of the distribution, where observations are sparse. If the estimate was smoothed to deal with this, then the main part of the distribution would probably be over-smoothed and detail lost. This leads to the consideration of variable bandwidth kernel density estimates.

Silverman (1986) defines, on page 21, the *variable kernel estimate* with smoothing parameter h by

$$\hat{f}(x) = \frac{1}{n}\sum_{j=1}^{n} \frac{1}{h\delta_{j,k}} K\left(\frac{x - X_j}{h\delta_{j,k}}\right) \quad ,$$

where $\delta_{j,k}$ is the distance from X_j to the k^{th} nearest point in the set comprising the other $n-1$ data points. The choice of k determines how dependent the smoothing parameter choice will be on very local detail. The variable kernel method is a special case of the adaptive kernel method. This is discussed in section 5.3 of Silverman (1986). Terrell & Scott (1992), and Lepski *et al.* (1997) discuss variable kernels in the context of regression.

Variable bandwidth estimates are competitors to wavelet methods, but it is difficult to obtain good public domain software.

2.6.2 Maximum Likelihood Estimates

Good & Gaskins (1971) consider a maximum likelihood approach to density estimation. The *maximum likelihood density estimate* is defined as the solution to the following constrained optimisation problem:

$$\text{maximise } L(f) \text{ subject to } \int f(x)\,dx = 1 \text{ and } f(x) \geqslant 0 \quad \forall\, x \quad ,$$

where the functional $L(f)$ is the log-likelihood:

$$L(f) = \sum_{i=1}^{n} \log f(X_i) \quad .$$

However, naïve application of this method would result in an estimate that was the mean of a set of Dirac functions at the n observations, i.e. the empirical density function:

$$\widetilde{f}(x) = \frac{1}{n}\sum_{i=1}^{n} \delta(x - X_i) \ . \tag{2.49}$$

This is of no interest, and the practice is to introduce a penalty function $F(f)$, which penalises roughness, and maximise:

$$\widetilde{L}(f) = \sum_{i=1}^{n} \log f(X_i) - F(f) \ .$$

They consider several penalty functions, including the use of Fisher information which leads to the following penalty functional:

$$F(f) = \alpha \int \frac{f'^2(x)}{f(x)} \, dx \ . \tag{2.50}$$

They also propose using the transformation $g^2 = f$, which avoids the non-negativity constraint. The problem now becomes:

$$\text{maximise } \widetilde{L}(g^2) \text{ subject to } \int g^2(x) \, dx = 1 \ ,$$

where

$$F(g^2) = 4\alpha \int g'^2(x) \, dx \ .$$

2.6.3 Orthogonal Series Density Estimates

OSDEs were introduced by Čencov (1962). The unknown density is represented as a convergent orthogonal series expansion

$$f(x) = \sum_{j \in \mathbb{I}} a_j \xi_j(x)$$

where $\{\xi_j, j \in \mathbb{I}\}$, with $\mathbb{I} \subseteq \mathbb{Z}$, is a complete orthonormal system of functions in $L_2(\mathbb{D})$, where $\mathbb{D} \subseteq \mathbb{R}$. There are many possible choices of basis functions. Tarter & Lock (1993) consider the use of Fourier Series in Chapter 3 of their book. Other choices of basis functions include the Hermite functions (Schwartz (1967)). Engel (1990) considers the use of Haar series.

Note that whatever the particular set of basis functions it is always true that

$$\begin{aligned} a_j &= \langle f, \xi_j \rangle \\ &= \int_{\mathbb{D}} f(x) \xi_j(x) \, dx \\ &= E\left[\xi_j(X)\right] \ , \end{aligned}$$

since f is a p.d.f.. The usual estimator of the $\{a_j\}_{j\in\mathbb{I}}$ is

$$\widehat{a}_j = \frac{1}{n}\sum_{i=1}^{n}\xi_j(X_i)$$

and the resulting density estimate is then

$$\widehat{f}(x) = \sum_{j\in\mathbb{M}}\widehat{a}_j\xi_j(x) \ ,$$

where $\mathbb{M} \subseteq \mathbb{I}$. The choice of $\mathbb{M}$ determines how smooth the estimate is.

Kronmal & Tarter (1968) introduce a method for choosing $\mathbb{M}$ in the case of Fourier Series estimates. Diggle & Hall (1986) show that the method breaks down in some cases and suggest a different method. Clutton-Brock (1990) considers using the exponential of a series of polynomials to represent the density. This results in a simple likelihood and a complete set of sufficient statistics. Rafajlowicz (1997) shows that OSDEs retain the MISE consistency when the observations are grouped to the points of a uniform grid. Watson (1969) compares kernel and orthogonal series methods, and decides that there is little to recommend one method over the other. Hall (1983) discusses orthogonal series methods for densities of qualitative, quantitative and mixed data

2.7 Wavelet Density Estimation

In section 2.6.3 the use of orthogonal series in the estimation of density functions was considered. Wavelets provide an orthogonal basis with many attractive properties, so now look at using them in density estimation. Provided here is a review of work already done in this area; some new ideas will be introduced in chapter 3. Vannucci (1995) provides a useful summary of the univariate case and also explores the multivariate case, as does Masry (1997). The latter considers the multivariate case for stationary random processes. Vidakovic (1999) provides in chapter 7 an account of the use of wavelets in density estimation. See also the review papers by Antoniadis (2000) and Abramovich *et al.* (2000b).

The wavelet expansion of a function f is given in (2.29). If f is a density function then the coefficients defined in (2.7) and (2.30) can be written as

$$\begin{aligned} c_{jk} &= E\left[\phi_{jk}(X)\right] \ , \\ d_{jk} &= E\left[\psi_{jk}(X)\right] \ . \end{aligned}$$

The obvious estimators of these coefficients are

$$\widehat{c}_{jk} = \frac{1}{n}\sum_{i=1}^{n} \phi_{jk}(X_i) \;, \tag{2.51}$$

$$\widehat{d}_{jk} = \frac{1}{n}\sum_{i=1}^{n} \psi_{jk}(X_i) \;. \tag{2.52}$$

2.7.1 Linear Estimators

A number of different density estimators are considered in the literature. The simplest is the truncated scaling function estimator, which is linear:

$$\widehat{f}(x) = \widehat{f}_n(x) = \sum_{k=-K}^{K} \widehat{c}_{jk}\phi_{jk}(x) \;, \tag{2.53}$$

for some resolution j.

Masry (1994) considers using an estimator of the form (2.53) for estimating the pdf of a stationary mixing process $\{X_i\}_{i=-\infty}^{\infty}$ from observations $\{x_i\}_{i=0}^{n-1}$, and derives precise asymptotic MISE formulae. Densities f belonging to the Sobolev space H_2^σ are considered ($H_2^\sigma \equiv B_{22}^\sigma$). It is assumed that $j = j_n \to \infty$ and $K = K_n \to \infty$ as $n \to \infty$. Masry defines ε_n^2 to be the MISE (2.37),

$$\varepsilon_n^2 \triangleq M(\widehat{f}_n) \;,$$

and assumes that

$$f(x) \leqslant D|x|^{-\beta} \quad \text{for } |x| \geqslant 1 \text{ and some } \beta > 1 \;,$$

where D is some positive constant.

He first shows that, for i.i.d. observations, if $f \in L_2(-\infty,\infty)$ and is of bounded variation on $(-\infty,\infty)$ and $f \in H_2^\sigma$ and the MRA is r-regular with $0 < \sigma < r$ then with j_n and K_n given by:

$$\begin{aligned} 2^{j_n} &= n^{1/(2\sigma+1)} \;, \\ K_n &= 2^{2[(\beta+\sigma-0.5)/(2\beta-1)]j_n} \log(10+n) \;, \end{aligned}$$

then

$$n^a \varepsilon_n^2 \to 1 \quad \text{as } n \to \infty, \text{ where } a = \frac{2\sigma}{2\sigma+1} \;. \tag{2.54}$$

If the same assumptions on f are kept, but in addition insisting that $\beta > \sigma + 0.5$, then, with some restrictions on the mixing properties of the process, the result (2.54) holds for dependent observations. The rate is best possible for $f \in H_2^{\sigma}$. The restrictions on the process $\{X_i\}_{i=-\infty}^{\infty}$ are satisfied by strongly mixing processes with exponential decay, for example ARMA processes.

Safavi *et al.* (1997) consider a truncated wavelet density estimator in their application of density estimation to process monitoring. This is again a linear estimate, of the form

$$\widehat{f}(x) = \sum_k \widehat{c}_{Lk}\phi_{Lk}(x) + \sum_{j=L}^{J}\sum_k \widehat{d}_{jk}\psi_{jk}(x) \quad ,$$

where the empirical coefficients are given by (2.51) and (2.52) and J is the cut-off chosen to achieve a certain degree of smoothness in the estimate, based on the measure:

$$\sup|\widehat{f}(x+h) - \widehat{f}(x)| < \alpha \qquad \forall h > 0 \quad ,$$

where α and h are some constants that determine the degree of smoothness. The empirical coefficients are not thresholded, and in fact the estimator is linear and equivalent to (2.53) with $j = J$.

The on-line monitoring and diagnosis of process operating performance are very important parts of the strategies aimed at improving a process and the quality of its products in the long term. The idea is that the behaviour of the process can be characterised using data obtained when the process is operating in a "normal region" (in a state of control). Then future unusual events can be detected by referencing new data to the control model. A basic tool is to estimate the p.d.f. of the control model – then the definition of a normal region is straightforward. The practical example considered in Safavi *et al.* (1997) is of a process with 6 output variables. The dimensionality is reduced to 2 by partial least squares. The normal operating region is defined by considering a 90% confidence level. Since this is in 2 dimensions it can be thought of as a contour on the density estimate. Then any new data falling outside of the region defined by this contour is indicative of a change in the operating conditions.

They also propose a data-selection method for dealing with short-range stationary dependence in the data. Given a set of observations $\{x_i\}_{i=1}^{n}$ they first perform wavelet density estimation on the data to give the estimator

$$\widehat{f}^1(x) = \sum_k \widehat{c}^1_{Lk}\phi_{Lk}(x) + \sum_{j=L}^{J}\sum_k \widehat{d}^1_{jk}\psi_{jk}(x) \ .$$

Then they use the set of odd indexed data to give the estimator

$$\widehat{f}^2(x) = \sum_k \widehat{c}^2_{Lk}\phi_{Lk}(x) + \sum_{j=L}^{J}\sum_k \widehat{d}^2_{jk}\psi_{jk}(x) \ .$$

This procedure could continue with further subsets (for example every third data point). Here the final estimate is obtained by averaging over the two previous ones,

$$\bar{f}(x) = \sum_k \bar{c}_{Lk}\phi_{Lk}(x) + \sum_{j=L}^{J}\sum_k \bar{d}_{jk}\psi_{jk}(x) \ ,$$

where

$$\bar{c}_{Lk} = \frac{1}{2}(\widehat{c}^1_{Lk} + \widehat{c}^2_{Lk}) \ ,$$

and similarly for the $\bar{d}_{jk}$.

Vannucci & Vidakovic (1995) consider the linear scaling function estimate (2.53). They do not truncate the estimator (i.e. they take $K = \infty$) but address the problem of the optimal number of basis functions by examining the choice of j. They propose a method for choosing the optimal value of j, say j^o. They use the method of Pinheiro & Vidakovic (1995) to estimate f by first estimating $\sqrt{f}$. They consider the results of Good & Gaskins (1971) to motivate the use of the Fisher information as a penalty. Good & Gaskins (1971) show that in the family of density functions with support $[a, b]$, such that $f(a) = f(b) = 0$, the function that minimises the Fisher information is

$$\widetilde{f}(x) = \frac{1}{m}\cos^2\frac{\pi x}{2m} \ ,$$

where $m = (b-a)/2$. The minimum information achieved by $\widetilde{f}$ is

$$F(\widetilde{f}) = \frac{\pi^2}{m^2} \ .$$

Beylkin (1992) shows that the wavelet estimator of f', the derivative of f, can be expressed as

$$\widehat{f'}(x) = \sum_k \widehat{r}_{jk}\phi_{jk}(x) \ ,$$

where

$$\widehat{r}_{jk} = 2^{-j} \sum_l r_l \widehat{c}_{j,k-l} \ , \tag{2.55}$$

with the coefficients $\widehat{c}_{jk}$ given by (2.51). The values of the weights $\{r_l\}$ are obtained by solving a system of equations and are tabulated in Beylkin (1992).

Vannucci & Vidakovic (1995) follow the same procedure for the case of the square root of f. The estimator becomes

$$\widehat{(\sqrt{f})'}(x) = \sum_k \widehat{r}_{jk} \phi_{jk}(x) \ ,$$

where the $\widehat{r}_{jk}$ are of the form (2.55), but where the $\widehat{c}_{jk}$ are given by (2.62). Then they use the Parseval identity to show that

$$\left\| \widehat{(\sqrt{f})'} \right\|^2 = \sum_k |\widehat{r}_{jk}|^2 \ ,$$

and then say that since

$$\left\| \widehat{(\sqrt{f})'} \right\|^2 = \frac{1}{4} \int \frac{\widehat{f'}^2(x)}{\widehat{f}(x)} \, dx \tag{2.56}$$

one gets

$$\frac{1}{4} \int \frac{\widehat{f'}^2(x)}{\widehat{f}(x)} \, dx = \sum_k |\widehat{r}_{jk}|^2 \ . \tag{2.57}$$

The left hand side of (2.57) can be recognised as the Fisher Information integral of $\widehat{f}$, $F(\widehat{f})$, as defined in (2.50). They then write

$$F(j) = \sum_k |\widehat{r}_{jk}|^2 \ ,$$

and claim that it can be proved that $F(j)$ decreases to zero for coarse levels and increases exponentially for fine levels.

They choose the optimal level j^o as follows. Let $[a, b]$ be the range of the observations $x_1, \ldots, x_n$. Calculate $F(j)$ for different values of j and choose a level j^o such that $4 \cdot F(j^o)$ is close to the minimum Fisher Information in the family of density functions with compact support $[a, b]$. This means that one should choose the smallest j^o such that

$$F(j^o) \geqslant \frac{\pi^2}{4m^2} \ .$$

They claim that this method is robust with respect to the value of k in the pre-estimator (2.63).

However, whilst it is true that

$$\left\| (\sqrt{f})' \right\|^2 = \frac{1}{4}\int \frac{f'^2(x)}{f(x)}\,dx$$

this does not mean that (2.56) is true. In particular,

$$\widehat{\left(\frac{f'}{2\sqrt{f}}\right)} = \frac{\widehat{f'}}{2\widehat{\sqrt{f}}}$$

is not, in general, true. Of course, (2.56) might hold in some approximate sense.

2.7.2 Non-Linear Estimators

It is only by considering non-linear wavelet estimators that the localised properties of wavelets can be fully exploited. Donoho *et al.* (1996) explore the asymptotic properties of estimators based on thresholding empirical wavelet coefficients. They explore the minimax rates of convergence over a large range of Besov function classes B^{σ}_{pq} and a range of global $L_{p'}$ error measures $1 \leqslant p' \leqslant \infty$. They define the mean $L_{p'}$ error by

$$R_n(\widehat{f}_n, f) = E\|\widehat{f}_n - f\|_{p'}^{p'} = E\int |\widehat{f}_n(x) - f(x)|^{p'}\,dx \quad \text{for } 1 \leqslant p' < \infty$$

and

$$R_n(\widehat{f}_n, f) = E\|\widehat{f}_n - f\|_{\infty} \quad \text{for } p' = \infty\ ,$$

where $\widehat{f}_n$ can be any density estimator based on a sample of size n. They look at the worst case performance over a variety of functional spaces

$$R_n(\widehat{f}; \mathcal{F}) = \sup_{f\in\mathcal{F}} R_n(\widehat{f}, f)\ ,$$

where $\mathcal{F}$ is a subset of densities with fixed compact support and bounded in the norm of one of the Besov spaces B^{σ}_{pq}. In fact the spaces they use are defined by

$$\begin{aligned} \mathcal{D}^{\sigma}_{pq}(M) &= \left\{ f : \int f = 1,\ f \geqslant 0,\ J'^{\sigma}_{pq} \leqslant M \right\} \\ \mathcal{D}^{\sigma}_{pq}(M,T) &= \left\{ f \in \mathcal{D}^{\sigma}_{pq}(M) : \operatorname{supp} f \subset [-T, +T] \right\}\ , \end{aligned}$$

where the norm J'^{σ}_{pq} is defined in (2.27) on page 15.

They derive rates of convergence for the classes of linear and non-linear density estimates. Since the class of non-linear estimators include all estimates, this rate is taken as the optimal one. They show that for $p' > p$ (also if $p' \geqslant 1$ & $\sigma > 1/p$) linear density estimates have sub-optimal rates of convergence.

Then they consider the non-linear truncated thresholded wavelet estimators. They define the empirical coefficients in the usual way and employ level dependent hard thresholding (2.35) with thresholds

$$t_j = KC(j)n^{-1/2} \ . \tag{2.58}$$

Then the estimator TW associated with the functions $j_0(n), L(n), C(j)$ and constant K is

$$\mathrm{TW}(x) = \sum_{k\in\mathbb{Z}} \widetilde{c}_{L(n)k}\phi_{Lk}(x) + \sum_{j=L(n)}^{j_0(n)} \sum_{k\in\mathbb{Z}} \widehat{d}_{jk}\psi_{jk}(x) \ . \tag{2.59}$$

Their theorem 3 describes the behaviour of TW when p, q, σ are known, for the case $C(j) = \sqrt{j}$. They show that TW attains the optimal rate of convergence to within logarithmic terms, and attain the exactly optimal rate in the "sparse" case. Defining $\varepsilon = \sigma p - (p' - p)/2$, the sparse case occurs when $\varepsilon > 0$.

Their theorem 5 shows that a slight modification of TW renders it adaptive, in the sense that it either exactly or approximately achieves the rates of convergence of theorem 3 without the need to specify σ, p, q. However, it still depends on p' and M. The modification, denoted ATM, is obtained from compactly supported and $(r_0 + 1)$ regular ϕ, ψ in (2.59) simply by specifying $C(j) = \sqrt{j}$ as before and

$$\begin{aligned} 2^{L(n)} &\leqslant n^{\frac{1}{(1+2r_0)}} < 2^{L(n)+1} \\ 2^{j_0(n)} &\leqslant \frac{n}{\log n} < 2^{j_0(n)+1} \ . \end{aligned}$$

The constant K in (2.58) is chosen as $c(M, \phi)r_0 p'$.

What does this mean in practice? One simply uses thresholds $t_j = K\sqrt{j/n}$, for some constant K on the levels which satisfy

$$\begin{aligned} & n^{\frac{1}{(1+2r_0)}} \leqslant 2^j \leqslant \frac{n}{\log n} \\ \Rightarrow \quad & \frac{1}{(1+2r_0)} \log_2 n \leqslant j \leqslant \log_2 n - \log_2(\log n) \ . \end{aligned}$$

How K should be chosen in practice is not clear.

Hall & Patil (1995) develop mean integrated squared error (MISE) formulae for nonlinear, thresholded wavelet based density estimators. They introduce an extra parameter p into the wavelet set up, so that the dilation/translation equations become

$$\begin{aligned} \phi_{jk}(x) &= p_j^{1/2}\phi(p_j x - k) \\ \psi_{jk}(x) &= p_j^{1/2}\psi(p_j x - k) \ , \end{aligned}$$

for arbitrary $p > 0$ and $p_j = p2^j$. Their notation for the wavelet density estimate is

$$\widehat{f}(x) = \sum_k \widehat{c}_{0k}\phi_{0k}(x) + \sum_{j=0}^{q-1}\sum_k \widetilde{d}_{jk}\psi_{jk}(x) \ ,$$

where $\widetilde{d}_{jk}$ are thresholded coefficients and $q \geqslant 1$ is another smoothing parameter. A hard threshold (2.35) is used with threshold τ.

They start by considering smooth f, assuming that f is r times differentiable, where ψ has $r-1$ vanishing moments.

Assume that $f^{(r)}$ exists in a piecewise sense, and is bounded and piecewise continuous on $(-\infty, \infty)$, with finite and well-defined left and right hand limits, and monotone on $(-\infty, -u)$ and (u, ∞) for sufficiently large u; and assume that $p \to \infty$, $q \to \infty$, $p_q\tau^2 \to 0$, $p_q^{2r+1}\tau^2 \to \infty$ and $\tau \geqslant C(n^{-1}\log n)^{1/2}$, where $C > C_0 \equiv 2\{r(\sup f)/(2r+1)\}^{1/2}$, they show that the MISE can be expressed as follows:

$$\int E\left[(\widehat{f} - f)^2\right] \sim n^{-1}p + p^{-2r}\kappa^2(1-2^{-2r})^{-1}\int f^{(r)2} \ , \tag{2.60}$$

where "$\sim$" means that the ratio of the left and right hand sides converges to 1 as $n \to \infty$, and the constant κ is given by

$$\kappa = (r!)^{-1}\int y^r\psi(y)\,dy = (r!)^{-1}(-\frac{1}{2})^{r+1}\sqrt{2}\sum(-1)^n n^r h(n) \ .$$

Comparing (2.60) with the MISE in the kernel case, we can see that the two are of the same basic form

$$\text{MISE} \sim C_1(nh)^{-1} + C_2h^{2r} \ ,$$

where p^{-1} is the wavelet analogue of the bandwidth h in the kernel case.

The right hand side of (2.60) is minimised by taking

$$p \sim an^{1/(2r+1)} \ ,$$

where

$$a = \left\{ 2r\kappa^2 (1 - 2^{-2r})^{-1} \int f^{(r)2} \right\}^{1/(2r+1)} \tag{2.61}$$

and the minimum size of (2.60) is const $\cdot\, n^{-2r/(2r+1)}$.

Although the theorem from which (2.60) is derived is only valid for "sufficiently large" p, by considering $p = O(\tau^{-2/(2r+1)})$ they show that nevertheless the *overall* minimum of the MISE does occur at $p \sim an^{1/(2r+1)}$. The practical implication of this is that the wavelet estimator is relatively robust against over smoothing, that is, against choosing p too small. In addition, the smaller the threshold, the greater the level of robustness. They remark that (2.60) holds also for soft thresholding (2.36).

They then move on to consider piecewise continuous f and show that if we impose the extra conditions that f is r-times differentiable in a piecewise sense and that $p_q^{2r+1} n^{-2r} \to \infty$ then the formula (2.60) still holds. They note that this result is in stark contrast to the kernel case, where the presence of discontinuities can dramatically increase the order of magnitude of the MISE.

They consider the estimation of derivatives of f. They note that wavelet coefficients of $f^{(s)}$ are

$$\begin{aligned} c_{jk}^{(s)} &= \int f^{(s)} \phi_{jk} = (-1)^s p_j^{s+1/2} \int f(x) \phi^{(s)}(p_j x - k)\, dx \\ \text{and} \qquad d_{jk}^{(s)} &= \int f^{(s)} \psi_{jk} = (-1)^s p_j^{s+1/2} \int f(x) \psi^{(s)}(p_j x - k)\, dx \end{aligned}$$

and so unbiased estimators of these quantities are

$$\begin{aligned} \widetilde{c_{jk}^{(s)}} &= (-1)^s p_j^{s+1/2} \frac{1}{n} \sum_{i=1}^{n} \phi^{(s)}(p_j X_i - k) \\ \text{and} \qquad \widetilde{d_{jk}^{(s)}} &= (-1)^s p_j^{s+1/2} \frac{1}{n} \sum_{i=1}^{n} \psi^{(s)}(p_j X_i - k)\ . \end{aligned}$$

The corresponding estimator of $f^{(s)}$ is

$$\widehat{f^{(s)}}(x) = \sum_k \widetilde{c_{0k}^{(s)}} \phi_{0k}(x) + \sum_{j=0}^{q-1} \sum_k \widetilde{d_{jk}^{(s)}} I\left(|\widetilde{d_{jk}^{(s)}}| > p_j^s \delta \right) \psi_{jk}(x)\ .$$

They claim that versions of their results for $\widehat{f}$ may be proved for $\widehat{f}^{(s)}$.

The final thing they consider is to use some of the above results to give a plug-in rule for choosing p. Let $\widehat{J}$ denote an estimator of $J = \int f^{(r)2}$, weakly consistent in the sense that $\widehat{J} \to J$ in probability. Let a be as defined in (2.61) and define

$$\widehat{a} = \left\{2r\kappa^2(1-2^{-2r})^{-1}\widehat{J}\right\}^{1/(2r+1)} .$$

Write $p_0 = an^{1/(2r+1)}$ and $\widehat{p}_0 = \widehat{a}n^{1/(2r+1)}$. Then the empirically chosen bandwidth $\widehat{p}_0$ asymptotically achieves the MISE associated with p_0. They claim that cross validation also produces an asymptotically optimal value of p.

2.7.3 Bona Fide Density Estimators

It is possible for a general orthogonal series estimator to take negative values. Although this phenomenon is usually confined to tails and sparsely represented regions of the density, reporting an estimator with negative values might be disturbing to the user. One way of dealing with this problem is to truncate the estimator to be non-negative and re-normalise, another is estimate a transformation of the density, for example $\log f$ or $\sqrt{f}$.

Pinheiro & Vidakovic (1995) consider estimating the square root of the density, which allows you to control the positiveness of the density estimate. They obtain a bona fide density estimate (i.e. one that integrates to 1) by scaling the coefficients to have l^2 norm one. The Parseval identity (2.33) then ensures that f integrates to one.

They note that $\sqrt{f}$ can be expanded as

$$\sqrt{f}(x) = \sum_k c_{Lk}\phi_{Lk}(x) + \sum_{j \geqslant L}\sum_k d_{jk}\psi_{jk}(x) \quad ,$$

with sample coefficients defined by:

$$\begin{aligned} \widetilde{c}_{jk} &= \sum_{i=1}^{n} \omega_i(X_1,\ldots,X_n)\phi_{jk}(X_i) \qquad (2.62) \\ \widetilde{d}_{jk} &= \sum_{i=1}^{n} \omega_i(X_1,\ldots,X_n)\psi_{jk}(X_i) \ , \end{aligned}$$

where

$$\omega_i(X_1,\ldots,X_n) = \frac{1}{n}\frac{1}{\sqrt{\widehat{f}_n(X_i)}}$$

for some pre-estimator of the unknown density $\widehat{f}_n$. The simplest pre-estimator is the *neighbourhood count* estimator, which defines the weights to be

$$\omega_i(X_1,\ldots,X_n) = \frac{1}{\sqrt{\sum_{j=1}^n I(X_i - \frac{\lambda}{n} \leqslant X_j \leqslant X_i + \frac{\lambda}{n})}} \qquad (2.63)$$

for some integer $\lambda < n$. The choice of this parameter is not discussed. Note that the scaling of the weights is unimportant as the empirical coefficients will be normalised later. The motivation for this choice of $\widetilde{c}_{jk}$ and $\widetilde{d}_{jk}$ comes from observing that

$$c_{jk} = \int \psi_{jk}(x)\sqrt{f}(x)\,dx = \int \frac{\psi_{jk}(x)}{\sqrt{f}(x)} f(x)\,dx$$

and similarly for d_{jk}.

To decide the number of levels of wavelet coefficients that should be included in the estimate they look at the *empirical scalogram*. This is defined as

$$\widetilde{\mathcal{E}}(j) = \sum_k \widetilde{d}_{jk}^2$$

By considering the behaviour of this for large j, and also the behaviour of the *scalogram*

$$\mathcal{E}(j) = \sum_k |\langle f, \psi_{jk}\rangle|^2 = \sum_k d_{jk}^2$$

they argue that the empirical scalogram can be expected to decrease with j at first, before starting an exponential increase. They define the VISUSTOP procedure for choosing the maximum level to choose the level after which the empirical scalogram commences its exponential growth.

Having selected the number of levels, the next stage is to threshold the remaining coefficients. They use the Lorentz thresholding procedure, as described in §2.3. As mentioned before, having selected the maximum number of levels and thresholded the remaining coefficients, the thresholded coefficients must be normalised to ensure that the resulting density integrates to one.

Penev & Dechevsky (1997) also consider using $g = \sqrt{f}$ in density estimation, but propose using the bivariate wavelet decomposition on $g(x)g(y)$ and then setting $x = y$. This approach avoids "mixing levels" in the final estimate of f. This means that each additive term in the estimate of f depends on at most one value of j. Koo & Kim (1996) consider estimating $\log f$ to ensure a bona-fide density estimate, although the normalisation used to ensure that the estimate of f integrates to 1 would be difficult to evaluate in practice. They consider the rates of convergence of such an estimator when $\log f \in B^{\sigma}_{pq}$.

Chapter 3

Some New Methods for Wavelet Density Estimation

The aim of the research recorded in this chapter is to investigate non-linear, thresholded wavelet density estimators, particularly the practical problems involved in the implementation thereof. §3.1 will introduce the dataset that will be used as an example in this chapter, and then discuss the basic estimation scheme that will be used. The next two sections will examine the computation of the wavelet estimator, with §3.2 concentrating on the calculation of the wavelet coefficients and §3.3 concentrating on thresholding the noisy coefficients. In §3.4 simulated data from a known density will be used to investigate the performance of the methodology. The choice of one of the main parameters will be considered in §3.5, and §3.6 will present some concluding remarks.

3.1 Introduction

Before introducing the estimation scheme that will be investigated, the example dataset will be described.

3.1.1 Example Data Set

The data, obtained from David Balding of Reading University, consist of measurements of the lengths of gene fragments, which are used as markers for forensic identification. The idea is that the corresponding sets of fragment lengths are likely to be different for

Table 3.1: The number of observations in each cell of the gene fragment data.

Ethnic Group	Locus 1	Locus 2	Locus 3	Locus 4
Afro-Caribbean	1050	1068	1043	420
Northern European	2401	2398	2377	1080

every person. There are measurements corresponding to four loci (the position of the fragment on the gene) for each of two ethnic groups (Afro-Caribbean and Northern-European). Table 3.1 shows the number of observations in each data set, and table 3.2 the range of values in each set. Note that the observations are recorded to 2 decimal places.

The underlying distribution is discrete, but the lengths are measured with a large amount of error so the actual distribution is best thought of as smooth but spiky. The spikyness will give some very steep sections to the density functions, which is why wavelet methods may well be suitable. For many applications, the underlying discrete distribution is required, a problem which does not concern us here. However in some cases the actual sampling density is needed, for example to calculate a likelihood ratio for comparing whether two sets of gene fragments are independent measurements of the same gene or independent measurements of two independent genes. Kernel methods will underestimate the peaks of the sampling density and so will bias the likelihood ratio. This forensic application is discussed in more detail in Roeder (1994) and Aitken (1995).

Whilst this sort of data poses a statistically interesting problem and will crop up in many applications, the immediate interest of this particular dataset from the forensic perspective has declined because new technology has been developed which reduces the measurement error to near zero.

Kernel Density Estimates

Kernel density estimation is discussed in §2.6.1. From the description of the data, and simple summary plots like stem and leaf or histograms (see figure 3.1 on page 52), it is obvious that the data are not normal, and in general the distributions are skewed to the left. Three bandwidth selection procedures were used (the `S-Plus` default (2.48), "rule

Table 3.2: The range of observations in each cell of the gene fragment data.

Ethnic Group	Locus 1	Locus 2	Locus 3	Locus 4
Afro-Caribbean	$(1.18, 19.93)$	$(1.57, 16.14)$	$(2.87, 19.08)$	$(1.13, 8.79)$
Northern European	$(1.11, 19.90)$	$(1.74, 14.89)$	$(2.58, 17.67)$	$(1.54, 9.18)$

of thumb" method (2.47) and the Sheather-Jones method, with no sub-sampling) and from there the pictures were smoothed by eye. Note that since from the summaries the data appear non-normal the rule of thumb method cannot be expected to give good results. The results are shown in figure 3.2 on page 53, and the values of the smoothing parameter chosen by the three methods are in table 3.3 on the same page. The values chosen are all reported on the same scale and all need to be multiplied by 4 for use in the `S-Plus` function `density`. It can be seen that the two ethnic groups have very similar distributions at locus 1 but quite different distributions at locus 2 and 3. At locus 4, the distributions of both groups peak in roughly the same area, but the Northern European data has an extra peak around $x = 4$ which is not present in the Afro-Caribbean data.

3.1.2 Basic Estimation Scheme and Notation

Consider estimating a density function f from i.i.d. data $x_1, \ldots, x_n$, using wavelets. Recall from (2.29), (2.28), (2.7) and (2.30) that the formal wavelet expansion of the density function is:

$$f(x) \sim \sum_{k \in \mathbb{Z}} c_{Lk} \phi_{Lk}(x) + \sum_{j \in \mathbb{J}_L} \sum_{k \in \mathbb{Z}} d_{jk} \psi_{jk}(x) \ , \tag{3.1}$$

where $\mathbb{J}_L = \{m \in \mathbb{Z} : m \geqslant L\}$ and the coefficients c_{Lk} and d_{jk} are given by:

$$\begin{aligned} c_{Lk} &= \int_{-\infty}^{\infty} \phi_{Lk}(x) f(x)\, dx \ , \\ d_{jk} &= \int_{-\infty}^{\infty} \psi_{jk}(x) f(x)\, dx \ . \end{aligned}$$

Following Hall & Patil (1995) (see §2.7.2), although with different notation, non dyadic resolutions will be considered. The standard dilations and translations will be

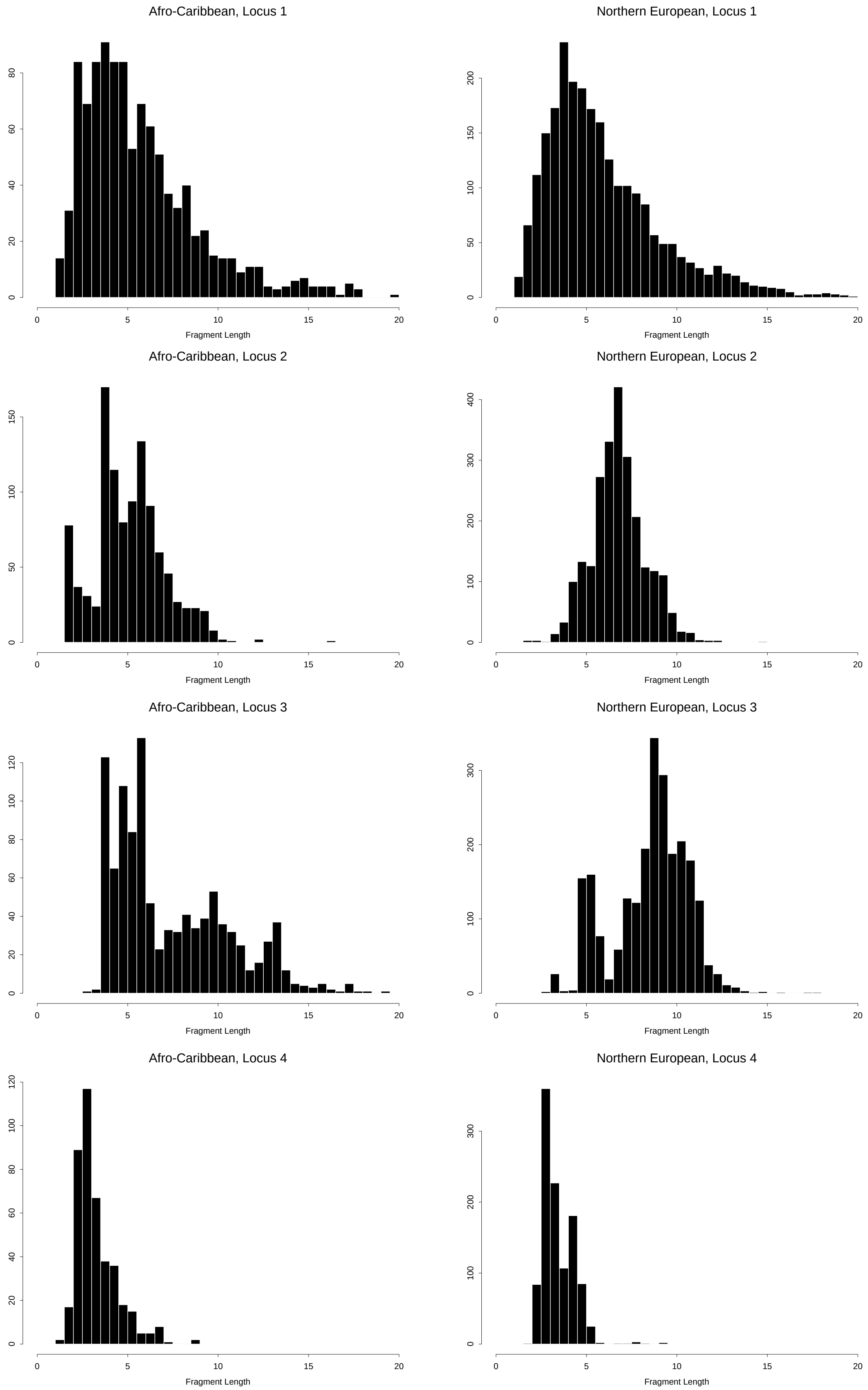


Figure 3.1: Histograms of the gene fragment data. The left-hand column is for the Afro-Caribbean data and the right-hand column for the Northern European data. The bin width in all plots is 0.5.

Table 3.3: Smoothing parameters chosen by three methods for the gene fragment data The results are all on the same scale and are reported to 3 decimal places.

Ethnic Group	Method	Locus 1	Locus 2	Locus 3	Locus 4
Afro-Caribbean	`S-Plus` default	0.212	0.165	0.184	0.099
	Sheather-Jones	0.487	0.112	0.309	0.138
	Rule of thumb	0.729	0.434	0.849	0.304
Northern European	`S-Plus` default	0.192	0.134	0.154	0.086
	Sheather-Jones	0.433	0.186	0.155	0.074
	Rule of thumb	0.639	0.290	0.458	0.224

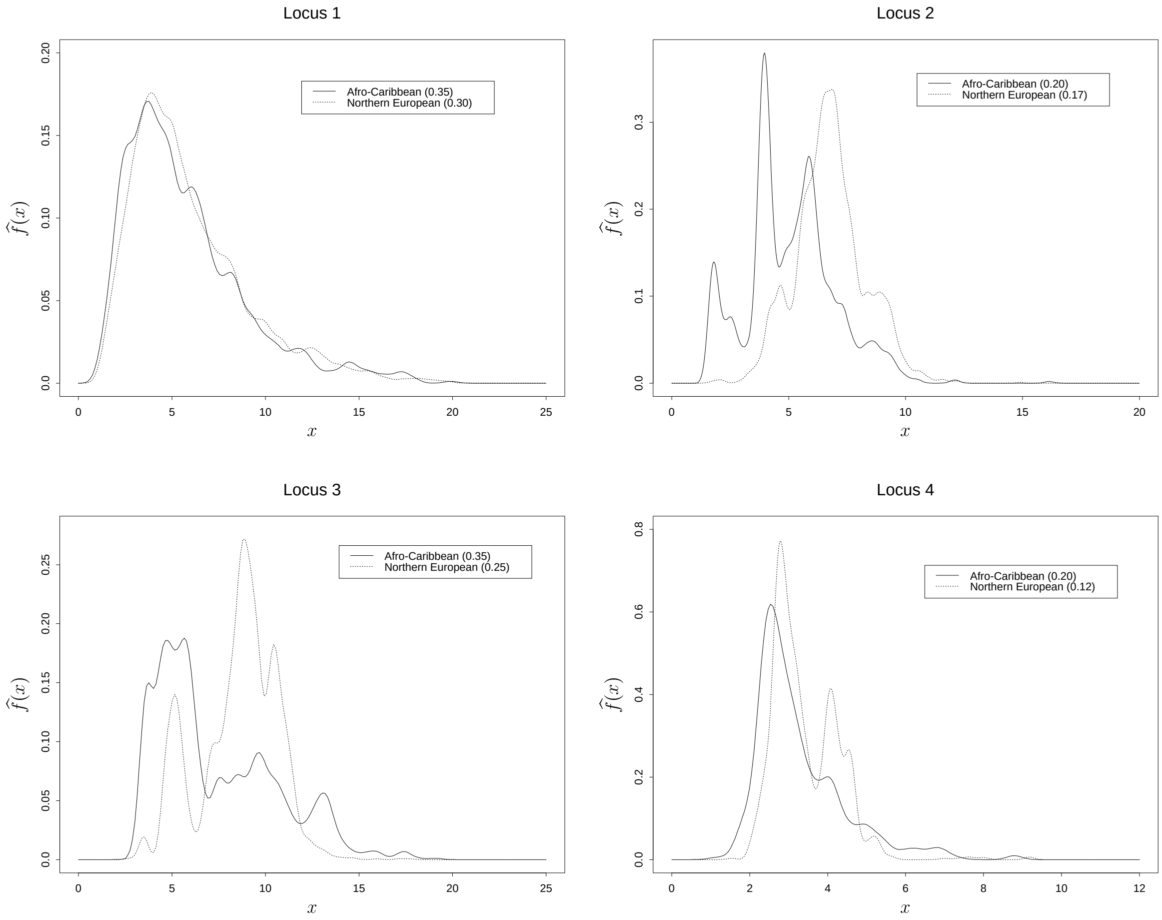


Figure 3.2: Kernel density estimates of the gene fragment data, using a normal kernel. The smoothing parameters, which were chosen by eye, are given in the legends.

written as:

$$\phi_{jk}(x) = p_{j\tau}^{1/2}\phi(p_{j\tau}x - k) \ , \tag{3.2}$$

$$\psi_{jk}(x) = p_{j\tau}^{1/2}\psi(p_{j\tau}x - k) \ , \tag{3.3}$$

where the *resolution* $p_{j\tau} = \tau\, 2^j$ for *tuning parameter* $\tau \in \left(\frac{1}{2}, 1\right]$ and *resolution level* j. Since τ simply rescales the problem, the set $\{\phi_{Lk}, \psi_{jl} : k, l \in \mathbb{Z},\, j \in \mathbb{J}_L\}$, will form an orthonormal basis for $L^2(\mathbb{R})$ if the corresponding set in the usual wavelet notation (i.e. with $\tau = 1$) forms such a basis.

Consider a density estimate which has the same form as (3.1). As noted in §2.6.3 for general orthogonal series estimators, the coefficients can be written as:

$$\begin{aligned} c_{jk} &= E\Big[\phi_{jk}(X)\Big] \\ d_{jk} &= E\Big[\psi_{jk}(X)\Big] \ , \end{aligned}$$

where the obvious estimators are:

$$\widetilde{c}_{jk} = \frac{1}{n}\sum_{i=1}^{n}\phi_{jk}(X_i) \tag{3.4}$$

$$\widetilde{d}_{jk} = \frac{1}{n}\sum_{i=1}^{n}\psi_{jk}(X_i) \ . \tag{3.5}$$

These empirical coefficients are calculated for resolution levels L up to some large value J, say, which is chosen so that $P_J f$ approximates f very well. J is called the *primary resolution level* and $p = p_{J\tau}$ the *primary resolution*. The empirical coefficients $\widetilde{d}_{jk}$ are then thresholded to give estimated coefficients $\widehat{d}_{jk}$. The resulting density estimate is then:

$$\widehat{f}(x) = \sum_{k}\widetilde{c}_{Lk}\phi_{Lk}(x) + \sum_{j=L}^{J-1}\sum_{k}\widehat{d}_{jk}\psi_{jk}(x) \ .$$

Note that this can be thought of as smoothing the empirical density function (2.49) by thresholding its wavelet expansion. To see this, consider the wavelet coefficients of the empirical density function:

$$\begin{aligned} \int \psi_{jk}(x)\widetilde{f}(x)\,dx &= \frac{1}{n}\sum_{i=1}^{n}\int \psi_{jk}(x)\delta(x - X_i)\,dx \\ &= \frac{1}{n}\sum_{i=1}^{n}\psi_{jk}(X_i) \\ &= \widetilde{d}_{jk} \ . \end{aligned}$$

So the true wavelet coefficients of the empirical density function are the empirical coefficients of f defined in (3.5).

The values of the parameters J, τ and L are, of course, unknown and must be chosen. This will be discussed later in this chapter. Note that n is not required to be a power of two and that this theory is also valid for higher dimensional data ($X_i \in \mathbb{R}^d$).

3.2 Efficient Computation of the Empirical Coefficients

Consider how to compute the coefficients (3.4) and (3.5). Rather than doing the projections directly for all the coefficients, it is more efficient to calculate the $\widetilde{c}_{Jk}$ directly and then use the DWT to find the $\widetilde{c}_{Lk}$ and $\widetilde{d}_{jk}$. Note that the coefficients found in this way will still have the form (3.4) and (3.5):

$$\begin{aligned}
\widetilde{d}_{J-1,k} &= \sum_l g(l-2k)\widetilde{c}_{Jl} \\
&= \frac{1}{n}\sum_l \sum_{i=1}^n g(l-2k)\phi_{Jl}(X_i) \\
&= \frac{1}{n}\sum_{i=1}^n \sum_l g(l-2k)\phi_{Jl}(X_i) \\
&= \frac{1}{n}\sum_{i=1}^n \psi_{J-1,k}(X_i) \ .
\end{aligned}$$

Similarly:

$$\widetilde{c}_{J-1,k} = \frac{1}{n}\sum_{i=1}^n \phi_{J-1,k}(X_i) \ ,$$

and by induction all the coefficients will have this form. This will be more efficient than direct computation of all coefficients because the direct projections are slow, particularly as the wavelets and scaling functions need to be evaluated. It now remains to find an efficient way to compute the $\widetilde{c}_{Jk}$.

3.2.1 Calculation of Limits on Various Indices

Assume compactly supported wavelets are being used, with integer bounds on the support, so for some $a_\phi, b_\phi \in \mathbb{Z}$, with $a_\phi < b_\phi$, supp $\phi = [a_\phi, b_\phi]$. The Daubechies wavelets (see §2.1.4)) satisfy this property, with $a_\phi = 0$ and $b_\phi = 2N-1$. Denote the width of this support by $w_\phi = b_\phi - a_\phi$. Note that

$$\text{supp}\ \phi_{Jk} = \left[\frac{a_\phi + k}{p_{J\tau}}, \frac{b_\phi + k}{p_{J\tau}}\right] \ , \tag{3.6}$$

so that the width of support of ϕ_{Jk} is $w_\phi / p_{J\tau}$.

The following results will be useful in calculating the $\widetilde{c}_{Jk}$. Note that $\lceil x \rceil$ denotes the smallest integer $\geqslant x$ and $\lfloor x \rfloor$ the largest integer $\leqslant x$. Firstly, for any i, the values of k for which $\phi_{Jk}(x_i)$ is non-zero will be found.

Values of k for which $\phi_{Jk}(x_i)$ is non-zero

Consider one particular observation x_i. Since compactly supported wavelets are being used, $\phi_{Jk}(x_i)$ will be non-zero only for the finite number of values of k for which the support of $\phi_{Jk}(x)$ covers x_i. From (3.2), these values of k satisfy:

$$\begin{aligned}
& & a_\phi \ &\leqslant\ p_{J\tau}x_i - k \ \leqslant\ b_\phi \\
&\Rightarrow & a_\phi - p_{J\tau}x_i \ &\leqslant\ -k \ \leqslant\ b_\phi - p_{J\tau}x_i \\
&\Rightarrow & p_{J\tau}x_i - b_\phi \ &\leqslant\ k \ \leqslant\ p_{J\tau}x_i - a_\phi \\
&\Rightarrow & \lceil p_{J\tau}x_i - b_\phi \rceil \ &\leqslant\ k \ \leqslant\ \lfloor p_{J\tau}x_i - a_\phi \rfloor \ ,
\end{aligned} \tag{3.7}$$

since k takes integer values.

Now the number of ϕ_{Jk} that contain x_i in their support will be found.

Number of ϕ_{Jk} covering each data point

The number (n_i) of k's for which $\phi_{Jk}(x_i) \neq 0$ for each data point will now be found. Note that since $a_\phi, b_\phi \in \mathbb{Z}$ and $\lfloor x \rfloor + 1 = \lceil x \rceil$ (for $x \notin \mathbb{Z}$):

$$\begin{aligned}
n_i &= \lfloor p_{J\tau}x_i - a_\phi \rfloor - \lceil p_{J\tau}x_i - b_\phi \rceil + 1 \\
&= \lfloor p_{J\tau}x_i \rfloor - \lceil p_{J\tau}x_i \rceil + 1 + b_\phi - a_\phi \\
&= b_\phi - a_\phi = w_\phi \ ,
\end{aligned} \tag{3.8}$$

So, for each point x_i, $\phi_{Jk}(x_i)$ will be non-zero for w_ϕ consecutive values of k, independently of the resolution. The total number of non-zero $\phi_{Jk}(x_i)$ is nw_ϕ, whatever the resolution. (If $x \in \mathbb{Z}$ then $\lfloor x \rfloor = \lceil x \rceil$, so for $p_{J\tau}x_i \in \mathbb{Z}$, $n_i = w_\phi + 1$. However, $\phi(a_\phi) = \phi(b_\phi) = 0$ so one of these values can be excluded so that (3.8) holds.)

Next, the range of values of k for which $\widetilde{c}_{Jk}$ is non-zero will be found.

Range of k

To implement the coefficient estimation global bounds on k will be needed, which can now easily be computed:

$$\begin{aligned} k_{\min} &= \min_{i=1}^{n} \lceil p_{J\tau} x_i - b_\phi \rceil \\ &= \lceil p_{J\tau} x_{\min} - b_\phi \rceil \ , \end{aligned} \tag{3.9}$$

$$\begin{aligned} k_{\max} &= \max_{i=1}^{n} \lfloor p_{J\tau} x_i - a_\phi \rfloor \\ &= \lfloor p_{J\tau} x_{\max} - a_\phi \rfloor \ . \end{aligned} \tag{3.10}$$

Maximum number of non-zero $\widetilde{c}_{Jk}$

Note that if, for a particular value k_1 of k, m data points lie in the support of ϕ_{Jk_1}, then there will be m non-zero terms $\phi_{Jk_1}(x_i)$ in the sum (3.4) for the coefficient $\widetilde{c}_{Jk_1}$. Now consider how the number of non-zero coefficients changes with the resolution. As J increases, the support width of ϕ_{Jk} will decrease and so each ϕ_{Jk} will cover fewer data points. Eventually, for large enough J, each ϕ_{Jk} will cover at most one of the x_i. When this happens, each non-zero coefficient will be of the form

$$\widetilde{c}_{Jk} = n^{-1} \phi_{Jk}(x_{i_k}), \tag{3.11}$$

where i_k is the index of the data point which lies in the support of ϕ_{Jk}. Since, from (3.8) there are w_ϕ ϕ_{Jk} covering each data point, there will be nw_ϕ non-zero coefficients for such a J, since there are nw_ϕ non-zero $\phi_{Jk}(x_i)$. This will be the maximum number of non-zero coefficients for any value of the primary resolution level J.

3.2.2 Computation

An `S-Plus` function, `denproj`, has been written to calculate the coefficients $\widetilde{c}_{Jk}$, given specified values of J and τ, for $k_{\min} \leqslant k \leqslant k_{\max}$. The computations are done by the `C` routine `SFDE`, which uses the Daubechies-Lagarias algorithm (see §2.1.4) to evaluate ϕ. This algorithm and the extension for calculating ψ have been implemented in `C`, with `S-Plus` interface functions. As noted on page 14, given the decimal part of $p_{J\tau} x_i$, the Daubechies-Lagarias algorithm in fact calculates all the non-zero $\phi_{Jk}(x_i)$. This is exploited in `SFDE`. For each data point, the Daubechies-Lagarias algorithm is used to evaluate the non-zero $\phi_{Jk}(x_i)$. Using (3.7) these contributions are then added onto the appropriate empirical coefficients. In other words, rather than calculating the coefficients $\widetilde{c}_{Jk}$ one at a time, they are gradually built up by taking each of the x_i in turn and adding on its contributions to all the $\widetilde{c}_{Jk}$.

The resulting high resolution density estimate can be plotted using the function `denplot`, which calls the `C` routine `PLDE`. Again, ϕ is evaluated using the Daubechies-Lagarias algorithm. Since the high resolution density estimate will be incredibly noisy, this function will be much more use after the transformed coefficients have been thresholded and reconstructed. If the reconstructed sequence after thresholding is denoted $\widehat{c}_{Jk}$ then the plotting function calculates:

$$\widehat{f}(x) = \sum_{k=k_{\min}}^{k_{\max}} \widehat{c}_{Jk}\phi_{Jk}(x) \ , \tag{3.12}$$

for specified values of x.

3.2.3 Applying the DWT

As mentioned before, having estimated the $\widetilde{c}_{Jk}$ the DWT is used to find the empirical coefficients at coarser levels. In the regression setting, with the requirement of 2^J data points, it is clear that there will be J levels in the wavelet decomposition. In addition there are a number of different methods of handling the boundaries that can be justified. There are some complications, however, in the density estimation setting. Boundary conditions will be discussed first.

Boundary Conditions

In the density estimation case neither periodic or symmetric boundary conditions are appropriate. Any coefficients not in the vector returned by `denproj` are zeros, so complete knowledge about the $\widetilde{c}_{Jk}$ is available. In this sense there are not really any boundaries. Thus when using the DWT, zero boundary conditions must be used; whenever the filter overlaps the ends of the scaling function coefficient vector, zeros are used to fill up the "missing" values. This option was not available with the `WaveThresh` software, so modifications were made to enable the use of zero boundary conditions.

Number of Coefficients at Each Level

Another consequence of having complete knowledge about the coefficients is that there is no need to have a dyadic number of coefficients at the highest level (J). At each level it is only necessary to calculate all the coefficients that are non-zero. In practice all the $\widetilde{c}_{jk}$ are calculated for $f_c^j \leqslant k \leqslant l_c^j$, where f_c^j and l_c^j are the smallest and largest values, respectively, of k such that $\widetilde{c}_{jk} \neq 0$. Similarly, the $\widetilde{d}_{jk}$ are calculated for $f_d^j \leqslant k \leqslant l_d^j$, where f_d^j and l_d^j are the smallest and largest values, of k such that $\widetilde{d}_{jk} \neq 0$.

The range of these coefficients can be calculated from the range at the previous level (above) by using the following formulae, which are derived on page 24 of Nason & Silverman (1994):

$$f_c^{j-1} = \left\lceil \tfrac{1}{2}(f_c^j - N_h + 1) \right\rceil \,, \tag{3.13}$$

$$l_c^{j-1} = \left\lfloor \tfrac{1}{2} l_c^j \right\rfloor \,, \tag{3.14}$$

$$f_d^{j-1} = \left\lceil \tfrac{1}{2}(f_c^j - 1) \right\rceil \,, \tag{3.15}$$

$$l_d^{j-1} = \left\lfloor \tfrac{1}{2}(l_c^j + N_h - 2) \right\rfloor \,, \tag{3.16}$$

where N_h is the length of the wavelet filter. For the Daubechies wavelet bases, $N_h = 2N$. The `WaveThresh` function `first.last` uses (3.13) – (3.16) to calculate a database of first and last values when symmetric boundary conditions are used. However, as will be discussed in the next section, the number of levels in the decomposition are not known a priori so this function had to be adapted for calculating the first/last database with zero boundary conditions.

Number of Levels

Since there are not a dyadic number of $\widetilde{c}_{Jk}$ being entered into the DWT, it is not immediately clear how many levels of coefficients there will be in the wavelet decomposition. The number of $\widetilde{c}_{jk}$ certainly decreases with decreasing j and must reach some minimum, so the simplest way to determine the number of levels is to carry on decomposing until this minimum is reached. To do this, consider whether f_c^j and l_c^j converge to some limits f_c and l_c, say. If f_c and l_c can be found then convergence is guaranteed. From

(3.13) f_c must satisfy:

$$\begin{aligned} f_c &= \left\lceil \tfrac{1}{2}(f_c - N_h + 1) \right\rceil \\ \Rightarrow \quad f_c - 1 &< \tfrac{1}{2}(f_c - N_h + 1) \leqslant f_c \\ \Rightarrow \quad f_c - 2 &< 1 - N_h \leqslant f_c \\ \Rightarrow \quad f_c &= 1 - N_h \quad \text{or} \quad 2 - N_h \ . \end{aligned}$$

Similarly, it can be shown that if l_c exists then

$$l_c = 0 \quad \text{or} \quad -1 \ .$$

Since the limits f_c and l_c have been found they must exist. Now define the parameter L to be the first level at which f_c^j and l_c^j reach the limits f_c and l_c. Note that there is no reason why L should be zero. The first/last database can therefore be found by applying (3.13) – (3.16) until f_c^j and l_c^j converge.

Number of Coefficients at Lowest Level

It is possible to show when $l_c^L = 0$ and when $l_c^L = -1$, and similarly for f_c^L. These conditions are:

$$l_c^J \geqslant 0 \quad \Rightarrow \quad l_c^L = 0 \ , \tag{3.17}$$

$$l_c^J < 0 \quad \Rightarrow \quad l_c^L = -1 \ , \tag{3.18}$$

$$f_c^J \geqslant 2 - N_h \quad \Rightarrow \quad f_c^L = 2 - N_h \ , \tag{3.19}$$

$$f_c^J < 2 - N_h \quad \Rightarrow \quad f_c^L = 1 - N_h \ . \tag{3.20}$$

The proof is by induction. Consider, for example, (3.17). Note from (3.14) that

$$l_c^j = k \quad \Rightarrow \quad l_c^{j+1} = 2k \text{ or } 2k+1 \ . \tag{3.21}$$

Assume that if $l_c^J \in \{0, \dots, k\}$, for some positive integer k, then

$$l_c^j = 0 \quad \forall j \leqslant L \ , \tag{3.22}$$

for some $L \in \mathbb{Z}$. Now suppose that $l_c^J \in \{0, \dots, 2k+1\}$, which can be written as $l_c^J = 2n$ or $2n+1$, for some $n \in \{0, \dots, k\}$. Then, from (3.21), $l_c^{J-1} = n$. So,

$l_c^{J-1} \in \{0, \ldots, k\}$ and from the assumption (3.22) it can be seen that $l_c^j = 0\ \forall\, j \leqslant L-1$. From (3.21) it is clear that if $l_c^J \in \{0,1\}$ then $l_c^j = 0 \quad \forall\, j \leqslant J-1$.

Now consider the number of coefficients at level L, n_c^L. Clearly $n_c^L = l_c^L - f_c^L + 1$. From (3.17)–(3.20), there are four cases to consider:

1.

$$\left.\begin{array}{lcl} f_c^J & \geqslant & 2 - N_h \\ l_c^J & \geqslant & 0 \end{array}\right\} \Rightarrow \left.\begin{array}{lcl} f_c^L & = & 2 - N_h \\ l_c^L & = & 0 \end{array}\right\} \Rightarrow n_c^L = N_h - 1$$

2.

$$\left.\begin{array}{lcl} f_c^J & \geqslant & 2 - N_h \\ l_c^J & < & 0 \end{array}\right\} \Rightarrow \left.\begin{array}{lcl} f_c^L & = & 2 - N_h \\ l_c^L & = & -1 \end{array}\right\} \Rightarrow n_c^L = N_h - 2$$

3.

$$\left.\begin{array}{lcl} f_c^J & < & 2 - N_h \\ l_c^J & \geqslant & 0 \end{array}\right\} \Rightarrow \left.\begin{array}{lcl} f_c^L & = & 1 - N_h \\ l_c^L & = & 0 \end{array}\right\} \Rightarrow n_c^L = N_h$$

4.

$$\left.\begin{array}{lcl} f_c^J & < & 2 - N_h \\ l_c^J & < & 0 \end{array}\right\} \Rightarrow \left.\begin{array}{lcl} f_c^L & = & 1 - N_h \\ l_c^L & = & -1 \end{array}\right\} \Rightarrow n_c^L = N_h - 1$$

Note that the conditions in 2 are, though theoretically possible, unlikely to occur in practice. Clearly $f_c^J < l_c^J$. This will not be possible unless $N_h \geqslant 3$. If this is satisfied, then $n_c^J < N_h - 2$, so at the highest resolution the length of the non-zero coefficient vector will be less than the length of the filter, which is never likely to happen.

3.3 Thresholding

Having computed all the empirical wavelet coefficients, the next stage is to threshold (or denoise) them before reconstruction. Most established methods of threshold selection assume that the noisy wavelet coefficients are independent and normally distributed with equal variance. The empirical coefficients (3.4) and (3.5) are, however, clearly not mutually independent, as each observation x_i contributes to many different coefficients.

3.3.1 Covariance Structure of the Empirical Coefficients

Consider the covariance of the empirical wavelet coefficients:

$$
\begin{aligned}
\mathrm{Cov}\left[\widetilde{d}_{j_1k_1},\ \widetilde{d}_{j_2k_2}\right] &= \mathrm{Cov}\left[\frac{1}{n}\sum_{i=1}^{n}\psi_{j_1k_1}(X_i),\ \frac{1}{n}\sum_{l=1}^{n}\psi_{j_2k_2}(X_l)\right] \\
&= \frac{1}{n^2}\sum_{i=1}^{n}\sum_{l=1}^{n}\mathrm{Cov}\left[\psi_{j_1k_1}(X_i),\ \psi_{j_2k_2}(X_l)\right] \\
&= \frac{1}{n^2}\sum_{i=1}^{n}\mathrm{Cov}\left[\psi_{j_1k_1}(X_i),\ \psi_{j_2k_2}(X_i)\right] \\
&= \frac{1}{n}\mathrm{Cov}\left[\psi_{j_1k_1}(X),\ \psi_{j_2k_2}(X)\right] \\
&= \frac{1}{n}\Big(E\left[\psi_{j_1k_1}(X)\psi_{j_2k_2}(X)\right] - E\left[\psi_{j_1k_1}(X)\right]E\left[\psi_{j_2k_2}(X)\right]\Big) \\
&= \frac{1}{n}\left(\int \psi_{j_1k_1}(x)\psi_{j_2k_2}(x)f(x)\,dx - d_{j_1k_1}d_{j_2k_2}\right)\ , \qquad (3.23)
\end{aligned}
$$

as the X_i are assumed to be i.i.d. from f. Similarly, it is easy to see that

$$
\mathrm{Cov}\left[\widetilde{c}_{j_1k_1},\ \widetilde{c}_{j_2k_2}\right] = \frac{1}{n}\left(\int \phi_{j_1k_1}(x)\phi_{j_2k_2}(x)f(x)\,dx - c_{j_1k_1}c_{j_2k_2}\right)\ . \qquad (3.24)
$$

Consider the integral in (3.23). Clearly this will be zero if the supports of $\psi_{j_1k_1}$ and $\psi_{j_2k_2}$ do not overlap. Without loss of generality assume $j_2 \geqslant j_1$, and let supp $\psi = [a_\psi, b_\psi]$, for some $a_\psi, b_\psi \in \mathbb{Z}$, with $a_\psi < b_\psi$. Note that for the Daubechies familes of wavelets, $a_\psi = 1-N$ and $b_\psi = N$. Then for the product $\psi_{j_1k_1}(x)\psi_{j_2k_2}(x)$ to be non-zero, k_1 and k_2 must satisfy:

$$
\begin{aligned}
& \frac{a_\psi + k_2}{2^{j_2}} < \frac{b_\psi + k_1}{2^{j_1}} \qquad \text{and} \qquad \frac{b_\psi + k_2}{2^{j_2}} > \frac{a_\psi + k_1}{2^{j_1}} \\
\Rightarrow\quad & a_\psi + k_2 < \frac{b_\psi + k_1}{2^{j_1-j_2}} \qquad \text{and} \qquad b_\psi + k_2 > \frac{a_\psi + k_1}{2^{j_1-j_2}} \\
\Rightarrow\quad & \left\lceil \frac{a_\psi + k_1}{2^{j_1-j_2}} - b_\psi \right\rceil \leqslant\ k_2\ \leqslant \left\lfloor \frac{b_\psi + k_1}{2^{j_1-j_2}} - a_\psi \right\rfloor\ .
\end{aligned}
$$

Note that if $j_1 = j_2$ then this simplifies to:

$$
|k_1 - k_2| \leqslant w_\psi\ ,
$$

where $w_\psi = b_\psi - a_\psi$ is the support width of ψ.

It might be expected that $\mathrm{Cov}\left[\widetilde{d}_{jk_1},\ \widetilde{d}_{jk_2}\right]$ would be zero for $|k_1 - k_2| > w_\psi$, as in this case $\widetilde{d}_{jk_1}$ and $\widetilde{d}_{jk_2}$ are guaranteed to be based on distinct subsets of the data and hence be independent. From (3.23) this is clearly not true, because in this situation

$$
\mathrm{Cov}\left[\widetilde{d}_{jk_1},\ \widetilde{d}_{jk_2}\right] = -\frac{1}{n}d_{jk_1}d_{jk_2}\ ,
$$

which does not necessarily equal zero.

Note that from (3.23) the variances of the coefficients are given by:

$$\operatorname{Var}\left[\widetilde{d}_{jk}\right] = \frac{1}{n}\left(\int \psi_{jk}^2(x) f(x)\,dx - d_{jk}^2\right) . \tag{3.25}$$

So the empirical coefficients will not necessarily have equal variances, even within levels.

The low resolution coefficients are linear combinations of a relatively large number of independent random variables, so by the central limit theorem, an assumption of normality might well be reasonable. However, these coefficients are still not independent and no such assumption can be made about the high resolution coefficients, as these are linear combinations of only a few random variables.

Because of these complications, thresholding empirical coefficients from the Haar basis will be considered first. The simplicity of the Haar basis should enable insights to be gained for the more general case.

3.3.2 Haar Wavelet Basis

Some results specific to the Haar basis, which will be of use later in this section, will now be derived.

Variance and Distribution of Empirical Haar Coefficients

Recall that the mother wavelet and scaling function for the Haar basis are given in (2.6) and (2.19), so that the wavelets are given by:

$$\psi_{jk}(x) = \begin{cases} 2^{j/2} & \text{if } 2^{-j}k \leqslant x < 2^{-j}(k+\frac{1}{2}) \\ -2^{j/2} & \text{if } 2^{-j}(k+\frac{1}{2}) \leqslant x < 2^{-j}(k+1) \\ 0 & \text{otherwise} \end{cases} \tag{3.26}$$

$$\phi_{jk}(x) = \begin{cases} 2^{j/2} & \text{if } 2^{-j}k \leqslant x < 2^{-j}(k+1) \\ 0 & \text{otherwise .} \end{cases} \tag{3.27}$$

Using (3.26) and (3.27) it is easy to see that the true coefficients of the density can be written as:

$$\begin{aligned} d_{jk} &= \int \psi_{jk}(x) f(x)\, dx \\ &= 2^{j/2} \left\{ \int_{2^{-j}k}^{2^{-j}(k+\frac{1}{2})} f(x)\, dx - \int_{2^{-j}(k+\frac{1}{2})}^{2^{-j}(k+1)} f(x)\, dx \right\} , \qquad (3.28) \\ c_{jk} &= \int \phi_{jk}(x) f(x)\, dx \\ &= 2^{j/2} \int_{2^{-j}k}^{2^{-j}(k+1)} f(x)\, dx \ . \qquad (3.29) \end{aligned}$$

In addition, notice that

$$\begin{aligned} \int \psi_{jk}^2(x) f(x)\, dx &= 2^{j} \left\{ \int_{2^{-j}k}^{2^{-j}(k+\frac{1}{2})} f(x)\, dx + \int_{2^{-j}(k+\frac{1}{2})}^{2^{-j}(k+1)} f(x)\, dx \right\} \\ &= 2^{j} \int_{2^{-j}k}^{2^{-j}(k+1)} f(x)\, dx \\ &= 2^{j/2} c_{jk} \ . \qquad (3.30) \end{aligned}$$

From (3.25) and (3.30) the variance of the empirical Haar wavelet coefficients can be written as:

$$\mathrm{Var}\left[\widetilde{d}_{jk}\right] = \frac{1}{n}\left(2^{j/2} c_{jk} - d_{jk}^2\right) \ . \qquad (3.31)$$

Now consider the distributions of the coefficients. Figure 3.3 illustrates the construction of the empirical Haar coefficients. First look at the $\widetilde{c}_{jk}$. Let N_{jk} be the number of data points which fall in the interval $\left[2^{-j}k, 2^{-j}(k+1)\right)$. Then from (3.4) and (3.27):

$$N_{jk} = n 2^{-j/2} \widetilde{c}_{jk} \ . \qquad (3.32)$$

The probability of a random observation falling in the interval $\left[2^{-j}k, 2^{-j}(k+1)\right)$ is:

$$\begin{aligned} P\Big(X \in \left[2^{-j}k, 2^{-j}(k+1)\right)\Big) &= \int_{2^{-j}k}^{2^{-j}(k+1)} f(x)\, dx \\ &= 2^{-j/2} c_{jk} \ . \qquad (3.33) \end{aligned}$$

If n, the total number of observations is known, then since the observations are i.i.d., N_{jk} has the binomial distribution:

$$N_{jk} \sim \mathrm{Bin}(n, 2^{-j/2} c_{jk}) \ .$$

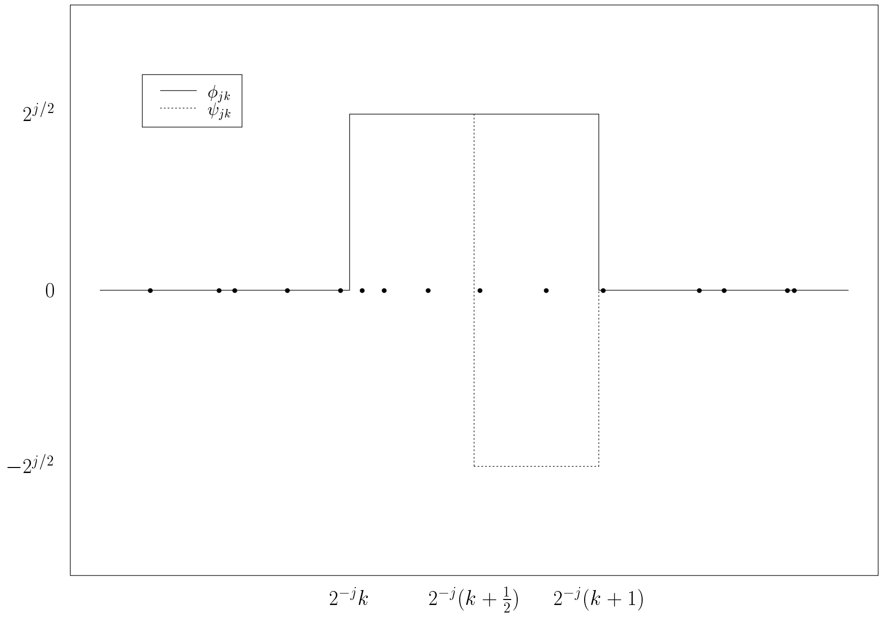


Figure 3.3: Illustration of the construction of the empirical Haar coefficients

Now consider the conditional distribution of $\widetilde{d}_{jk}$ given $\widetilde{c}_{jk}$. Let M_{jk} be the number of data points which fall in the interval $\big[2^{-j}k, 2^{-j}(k+\frac{1}{2})\big)$ and suppose it is known that $N_{jk}=n_{jk}$. Then from (3.5) and (3.26):

$$\begin{aligned} n2^{-j/2}\widetilde{d}_{jk} &= M_{jk} - (n_{jk} - M_{jk}) \\ \Rightarrow \quad M_{jk} &= \tfrac{1}{2}\big(n2^{-j/2}\widetilde{d}_{jk} + n_{jk}\big) \; . \end{aligned} \tag{3.34}$$

Now $M_{jk}\big|N_{jk} \sim \mathrm{Bin}(n_{jk}, p_{jk})$, where the probability of success, p_{jk}, is:

$$\begin{aligned} p_{jk} &= P\Big(X \in \big[2^{-j}k, 2^{-j}(k+\tfrac{1}{2})\big)\Big| X \in \big(2^{-j}k, 2^{-j}(k+1)\big)\Big) \\ &= \frac{P\Big(X \in \big[2^{-j}k, 2^{-j}(k+\tfrac{1}{2})\big)\Big)}{P\Big(X \in \big(2^{-j}k, 2^{-j}(k+1)\big)\Big)} \\ &= \frac{\int_{2^{-j}k}^{2^{-j}(k+\frac{1}{2})} f(x)\,dx}{2^{-j/2}c_{jk}} \qquad \text{from (3.33)} \\ &= \frac{2^{j/2}}{c_{jk}}\,\frac{1}{2}\left(\int_{2^{-j}k}^{2^{-j}(k+1)} f(x)\,dx + \left(\int_{2^{-j}k}^{2^{-j}(k+\frac{1}{2})} f(x)\,dx - \int_{2^{-j}(k+\frac{1}{2})}^{2^{-j}(k+1)} f(x)\,dx\right)\right) \\ &= \frac{2^{j/2}}{2\,c_{jk}}\big(2^{-j/2}c_{jk} + 2^{-j/2}d_{jk}\big) \qquad \text{from (3.28) and (3.29)} \\ &= \frac{c_{jk}+d_{jk}}{2\,c_{jk}} \; . \end{aligned} \tag{3.35}$$

Let m_{jk} be the sample value of M_{jk}.

Exact Binomial Thresholding

Thresholding the empirical (noisy) wavelet coefficients is an attempt to find which of the coefficients are really non-zero. This can be thought of as performing multiple hypothesis tests of:

$$H_0 : d_{jk} = 0 \quad v \quad H_1 : d_{jk} \neq 0 \; . \tag{3.36}$$

If $d_{jk}=0$ then, from (3.35), $p_{jk}=1/2$, so (3.36) is equivalent to testing:

$$H_0 : p_{jk} = 1/2 \quad v \quad H_1 : p_{jk} \neq 1/2 \; .$$

It should be noted that this procedure contains a large number of significance tests, which are not independent, as the $\widetilde{d}_{jk}$ are not independent.

This hypothesis testing thresholding routine is implemented in `S-Plus`. The n_{jk} are calculated from the $\widetilde{c}_{jk}$ and the m_{jk} from the $\widetilde{d}_{jk}$, as in equations (3.32) and (3.34). The p-value used in the individual tests must be supplied by the user. Alternatively, the user can supply the p-value required for the whole thresholding procedure. Bonferroni corrections are then used to calculate the p-value to be applied to the individual tests. However, this is rather misleading, because of the tests not being independent.

Normal Approximation

An alternative to the binomial testing procedure is to use the normal approximation to the binomial distribution. Using this, the coefficients can be directly thresholded, rather than having to transform them to integers first, as the exact binomial thresholding requires. However, a large number of hypothesis tests that are not independent are still being performed. From (3.31) the variance of $\widetilde{d}_{jk}$ under the null hypothesis is $n^{-1}2^{j/2}c_{jk}$, which can be estimated using the empirical scaling function coefficients. Note that since, for the Haar basis, ϕ is non-negative, the $\widetilde{c}_{jk}$ will also be non-negative.

Comparison

The exact binomial and normal approximation alternatives have been tried and give very similar results. They were tested on a sample of 1000 simulated observations from a density containing a discontinuity, which is shown in the top left of figure 3.4. It consists of two halves of normal distributions with different variances (10/3 and 32/3).

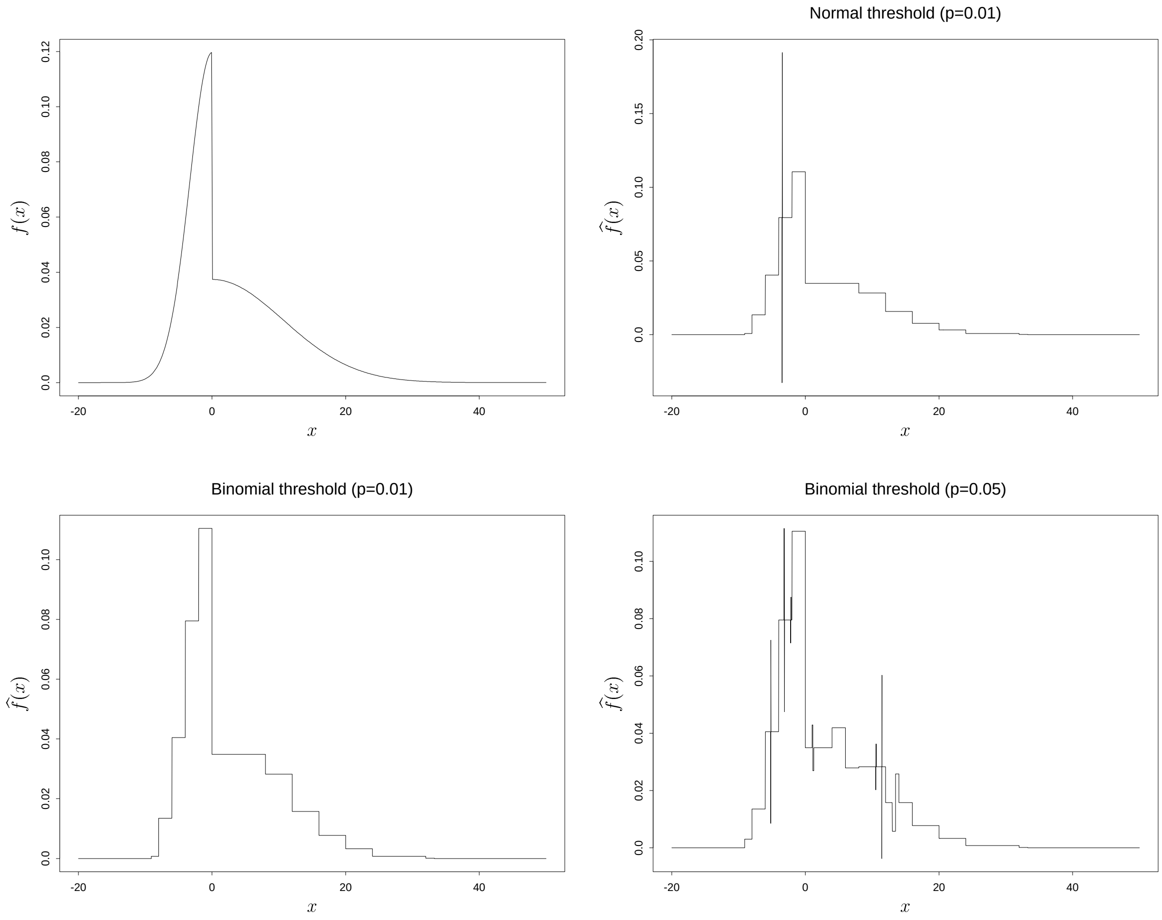


Figure 3.4: Haar wavelet density estimates of the discontinuous density (the left half is $N(0, 10/3)$ and the right half $N(0, 32/3)$) using a sample of size 1000. The top row shows the true density (left) and the estimate using the normal significance test for thresholding (right). The bottom row shows estimates using the binomial test at different p-values.

In the estimation procedure, the parameters were chosen as $J = 10$ and $\tau = 1$. This value of τ gives the standard wavelet representation. The choice of parameter J was based on the minimum interpoint distance procedure, which will be discussed in §3.5.1. The full version of this procedure (3.43) suggested $J = 20$, which is too high to be of practical use. The discounted version, however, suggested values of J around 10. The choice of this parameter will be discussed much more thoroughly in §3.5. The p-values at which the significance tests were performed are indicated in the plot titles. The results are shown in figure 3.4. Note that normal thresholding has let a fine scale coefficient through which the binomial thresholding kills. This sort of problem is to be expected as the normal approximation will be much worse at fine scales, as n_{jk} will be much smaller than at coarser scales.

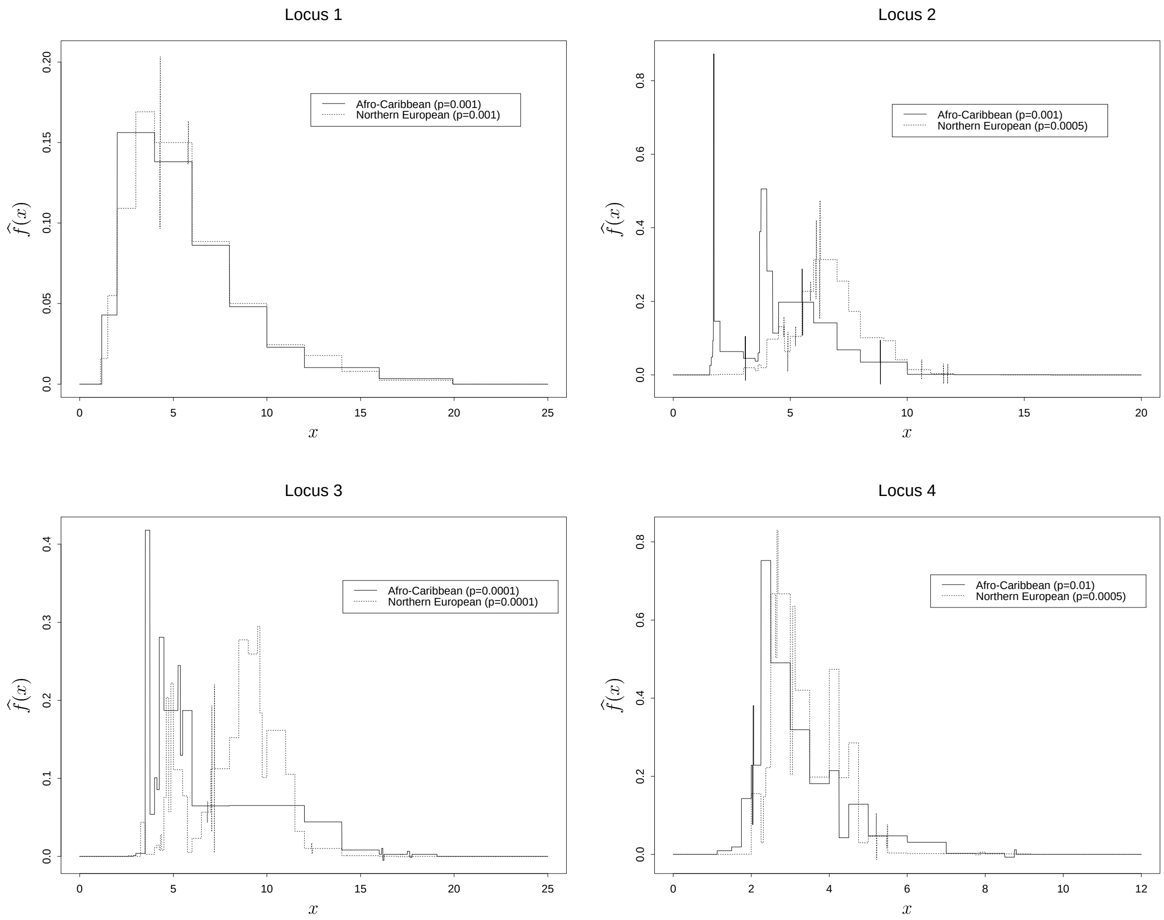


Figure 3.5: Haar wavelet density estimates of the gene fragment data. The p-values are given in the legends.

Gene Fragment Data

Figure 3.5 shows density estimates for the gene fragment data using Haar wavelets with the binomial thresholding. Again, $\tau = 1$ was used. $J = 7$ was chosen, as suggested by the discounted minimum interpoint distance scheme. The general behaviour exhibited is the same as that shown by the kernel density estimates, but the wavelets pick up some large spikes which are very resistant to thresholding (as well, unfortunately, as some smaller blips which would have been better smoothed out). A good example is the huge spike on the left of locus 2 for the Afro-Caribbean data. This is shown in close up in figure 3.6. The numbers plotted on the line $y = 1$ shown the number of coincident observations at their location. This shows good evidence in the data for the spike.

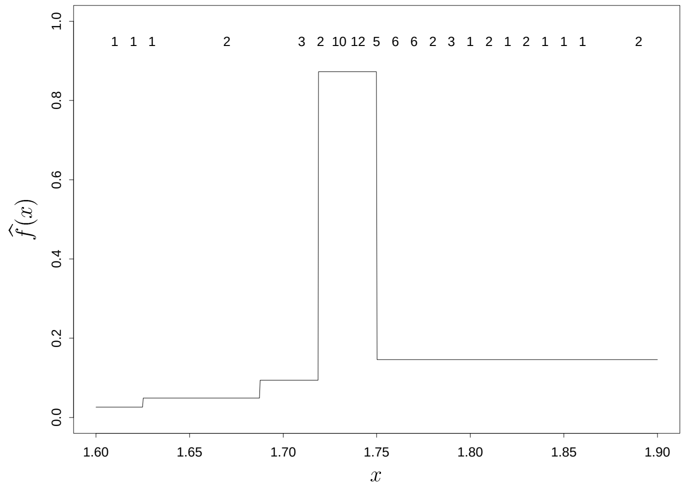


Figure 3.6: Close up of the spike in the Haar wavelet density estimate of the second locus of the Afro-Caribbean gene fragment data. The numbers indicate the number of coincident observations, plotted at their location on the line $y=1$.

3.3.3 Other Wavelet Bases - Using the Scaling Function Coefficients

Consider the normal approximation to the binomial distribution, as discussed above for the Haar basis. This is nice because of the simple form of the variance of the wavelet coefficients (3.31), which is expressed in terms of the scaling function coefficients at the same level. This works for the Haar basis because $\psi^2 = \phi$. This is obviously not true for any other of the Daubechies wavelets. In order to estimate the variance of the wavelet coefficients, it will be necessary to approximate ψ^2 in some way, so that the integral in (3.25) can be simplified. Following the Haar case, try expanding ψ^2 as a linear combination of scaling functions at some level. Recall that $\psi \in W_0 \subset V_1$, so it is likely that the ϕ_{jk} for some $j \geqslant 1$ will be needed.

Suppose that, approximately, $\psi^2 \in V_m$, for some m, so that:

$$\psi^2(x) \approx \sum_l e_l \phi_{ml}(x) \; , \tag{3.37}$$

$$\text{where} \qquad e_l = \int \psi^2(x)\phi_{ml}(x)\, dx \; .$$

The coefficients e_l can be found by applying the DWT, with zero boundary conditions, to carefully chosen values of ψ^2. This result can be used to approximate ψ_{jk}^2:

$$\begin{aligned}
\psi_{jk}^2(x) &= 2^j \psi^2(2^j x - k) \\
&\approx 2^j \sum_l e_l \phi_{ml}(2^j x - k) \\
&= 2^j \sum_l e_l 2^{m/2} \phi\big(2^m(2^j x - k) - l\big) \\
&= 2^{j/2} \sum_l e_l \phi_{j+m,l+2^m k}(x) \ ,
\end{aligned}$$

and so the following approximation can now be made:

$$\begin{aligned}
\int \psi_{jk}^2(x) f(x)\, dx &\approx 2^{j/2} \sum_l \left\{ e_l \int \phi_{j+m,l+2^m k}(x) f(x)\, dx \right\} \\
&= 2^{j/2} \sum_l e_l c_{j+m,l+2^m k} \ .
\end{aligned}$$

Using this result in (3.25):

$$\mathrm{Var}\left[\widetilde{d}_{jk}\right] \approx \frac{1}{n}\Big(2^{j/2} \sum_l e_l c_{j+m,l+2^m k} - d_{jk}^2 \Big) \ ,$$

so that under the null hypothesis (3.36):

$$\mathrm{Var}_{H_0}\left[\widetilde{d}_{jk}\right] \approx \frac{2^{j/2}}{n} \sum_l e_l c_{j+m,l+2^m k} \ .$$

This can be estimated using the empirical coefficients:

$$\widehat{\mathrm{Var}}_{H_0}\left[\widetilde{d}_{jk}\right] = \frac{2^{j/2}}{n} \sum_l e_l \widetilde{c}_{j+m,l+2^m k} \ . \tag{3.38}$$

If the empirical coefficients are then assumed to be normally distributed, they can be thresholded by performing multiple significance tests.

A nice feature of this idea is that the $\widetilde{c}_{jk}$ are computed by the DWT, so it exploits information that has already been computed. However, it will not be possible to use (3.38) on the finest $m-1$ levels of wavelet coefficients, as the necessary scaling function coefficients are not available. These coefficients will simply be set to zero. If m is small and J large then, under mild smoothness assumptions, this will not discard any appreciable signal.

Note that for compactly supported wavelets, there will only be a finite number of non-zero e_l, because there are only a finite number of values of l for which ψ and ϕ_{ml}

overlap. These values are:

$$p_{m\tau} a_\psi - b_\phi + 1 \leqslant l \leqslant p_{m\tau} b_\psi - a_\phi - 1 \ ,$$

assuming $m \geqslant 0$. For the Daubechies compactly supported wavelets, this simplifies to:

$$-(2 + p_{m\tau})(N-1) \leqslant l \leqslant p_{m\tau} N - 1 \ .$$

The result $\psi^2 \in V_m$ has not been proved in any way. It may be possible to prove this result by considering the Fourier series representation of ψ, which looks like, for the Daubechies wavelets:

$$\psi(x) \sim \sum_{n=-\infty}^{\infty} \Psi_n e^{i\frac{2n\pi x}{2N-1}}, \tag{3.39}$$

where

$$\Psi_n = \frac{1}{2N-1} \int_{-(N-1)}^{N} \psi(t) e^{-i\frac{2n\pi t}{2N-1}} \, dt \ .$$

Now, it is known that compactly supported wavelets are not compactly supported in the frequency domain, so the summation in (3.39) will be infinite. However, $\hat{\psi}(\omega)$ does decay rapidly, so a good approximation to ψ can be obtained by truncating (3.39) at some maximum frequency M, say. So

$$\psi(x) \approx \sum_{n=-M}^{M} \Psi_n e^{i\frac{2n\pi x}{2N-1}} \ .$$

Note that for the Daubechies wavelets, the approximation improves as the filter number N increases. This is because for higher N the wavelet is smoother and so is "more" compactly supported in the frequency domain. Using this, ψ^2 can be approximated as:

$$\psi^2(x) \approx \sum_{k=-2M}^{2M} \Psi'_k e^{i\frac{2k\pi x}{2N-1}} \ ,$$

where

$$\Psi'_k = \sum_{n=\max(-M,k-M)}^{\min(M,k+M)} \Psi_n \Psi_{k-n} \ .$$

Comparing this with the expansion of the ϕ_{jk} may yield more insight into a possible proof.

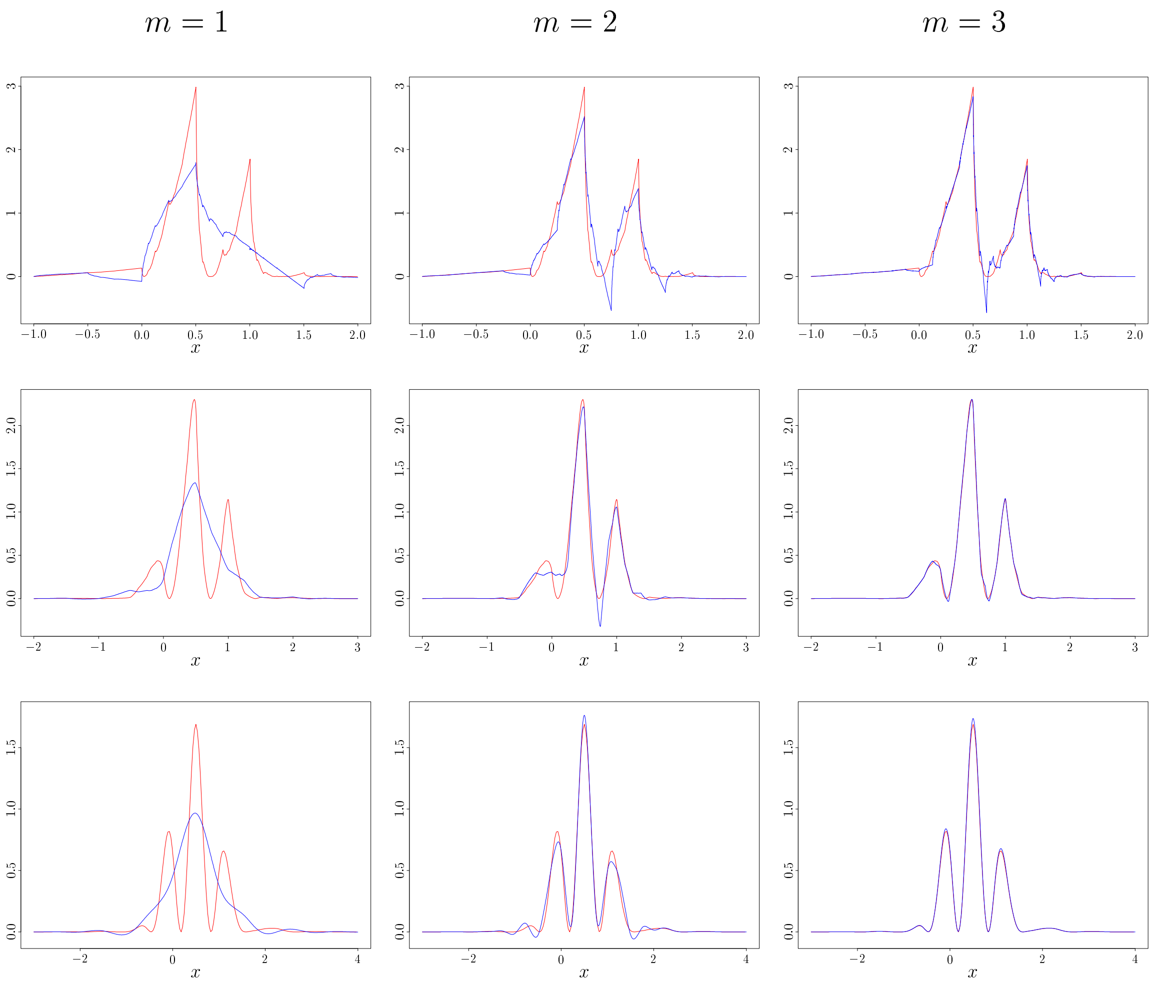


Figure 3.7: $\psi^2(x)$ (red) and the approximation $\sum_l e_l \phi_{ml}(x)$ (blue) for three different values of m. The top row is for the Daubechies extremal phase wavelet with $N=2$ and the other two rows for the Daubechies least asymmetric wavelets with $N=4$ (middle) and $N=10$ (bottom).

Although there is no formal proof, figure 3.7 shows the approximation (3.37) for the Daubechies extremal phase wavelet $(N = 2)$ and the least asymmetric wavelets $(N = 4, 10)$ for $m = 1, 2, 3$. As can be seen the approximation improves as m and N increase. The approximations at $m=1$ are clearly not good, but those at $m=2$ look as though they might be close enough for practical purposes.

This method was tried out on the data used in figure 3.4 for a variety of Daubechies wavelet bases for $m=1, 2$ and 3. Unfortunately the results were very poor in that the resulting estimates contained lots of high frequency noise, unless the thresholding was so severe as to remove most of the signal as well.

3.3.4 Other Wavelet Bases - Estimating the Variances Directly

Recall from section 3.1.2 on page 54 that the empirical wavelet coefficients are the true wavelet coefficients of the empirical density function (2.49). Using the same reasoning, another way of estimating the integral in (3.25) is:

$$\begin{aligned}\int \psi_{jk}^2(x) f(x)\, dx &\approx \int \psi_{jk}^2(x) \widetilde{f}(x)\, dx \\ &= \frac{1}{n}\sum_{i=1}^{n} \psi_{jk}^2(X_i)\ .\end{aligned}$$

The thresholding is still formulated as multiple hypothesis testing. Under the null hypothesis (3.36), the estimated variance is:

$$\widehat{\mathrm{Var}}_{H_0}\left[\widetilde{d}_{jk}\right] = \frac{1}{n^2}\sum_{i=1}^{n} \psi_{jk}^2(X_i)\ . \tag{3.40}$$

The most efficient way to calculate these is to estimate the covariance matrix of the $\widetilde{c}_{Jk}$ and use the DWT to find the estimated variances of the $\widetilde{d}_{jk}$ (3.40). From (3.24)

$$\mathrm{Cov}\left[\widetilde{c}_{Jk_1},\ \widetilde{c}_{Jk_2}\right] = \frac{1}{n}\left(\int \phi_{Jk_1}(x)\phi_{Jk_2}(x) f(x)\, dx - c_{Jk_1}c_{Jk_2}\right)\ . \tag{3.41}$$

Since the DWT is linear, $c_{Jk_1}c_{Jk_2}$ will transform to $d_{J-1,k}^2$. Under H_0 $d_{jk} = 0$, so it is only necessary to estimate the first term in (3.41). This is:

$$\widehat{E}\left[\widetilde{c}_{Jk_1}\widetilde{c}_{Jk_2}\right] = \frac{1}{n^2}\sum_{i=1}^{n} \phi_{Jk_1}(X_i)\phi_{Jk_2}(X_i)\ . \tag{3.42}$$

Notice that (3.42) defines a band limited matrix, where the bandwidth is $2N-1$. It is therefore possible to use the fast algorithm of Kovac & Silverman (2000) to find the variance, which improves the efficiency of the process. Note that again zero boundary conditions must be used. This meant that new code had to be written to perform this fast transform. The covariance matrix (3.42) is calculated by the `C` routine `SFDE` at the same time as the $\widetilde{c}_{Jk}$ are calculated. As with the coefficient computation, the Daubechies-Lagarias algorithm is exploited and the contributions of each data point to (3.42) added one at a time.

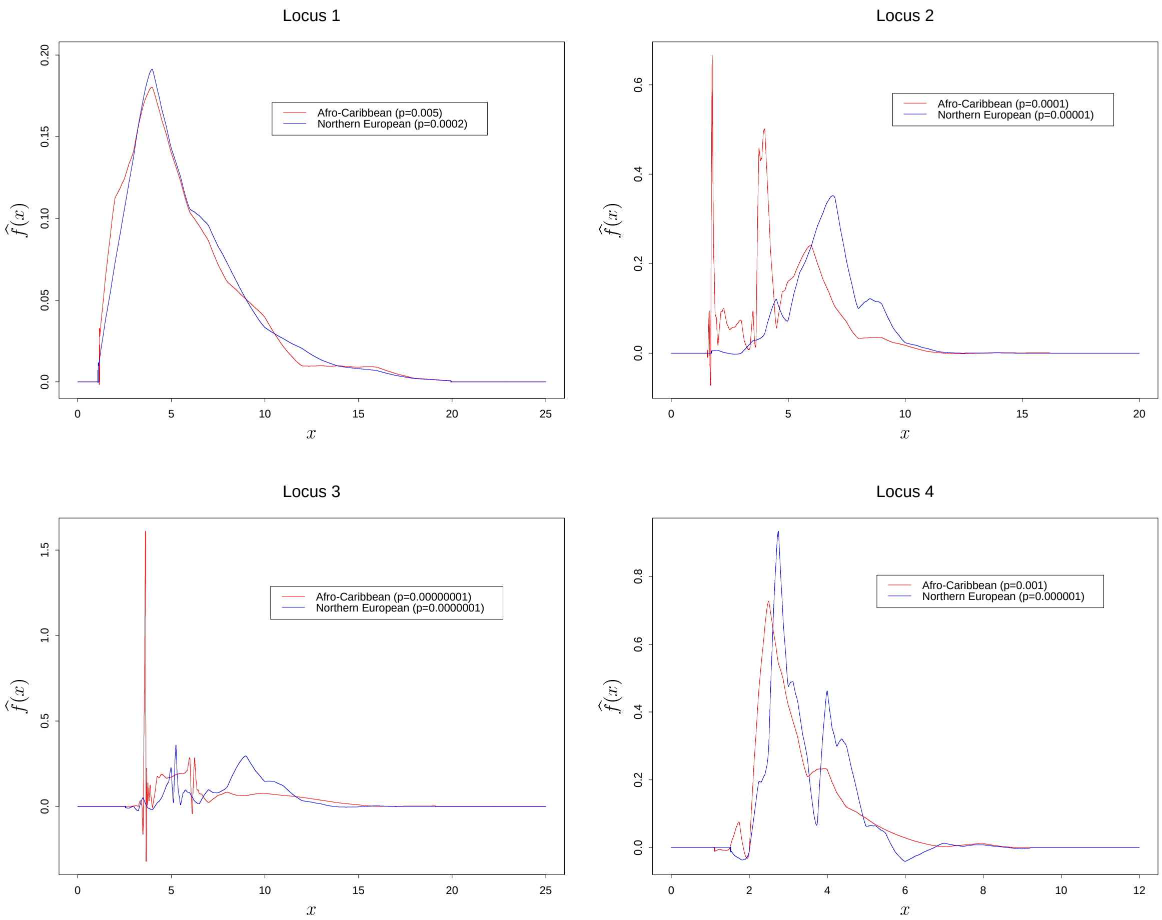


Figure 3.8: Daubechies least asymmetric wavelet $(N=4)$ density estimates of the gene fragment data. The p-values are given in the legends.

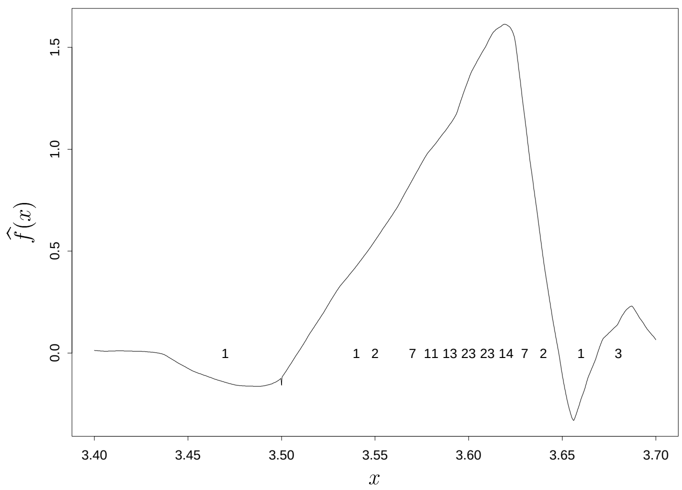


Figure 3.9: Close up of the spike in the Daubechies least asymmetric wavelet $(N=4)$ density estimate of the third locus of the Afro-Caribbean gene fragment data. The numbers indicate the number of coincident observations plotted at their location on the line $y=0$.

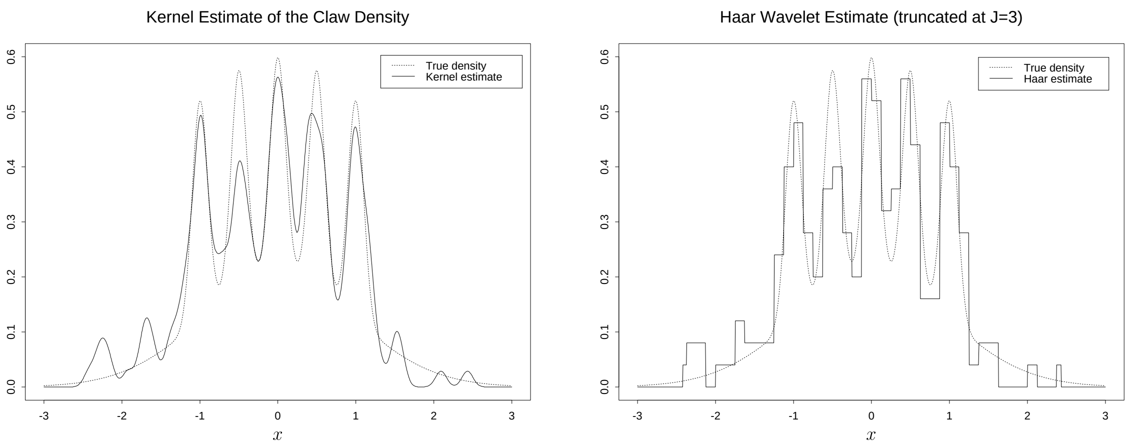


Figure 3.10: Kernel density estimate of the claw density (right) and Haar estimate truncated at $J=3$ (left).

Gene Fragment Data

Figure 3.8 shows density estimates for the gene fragment data using these directly estimated variances in the thresholding. The Daubechies least asymmetric wavelets (N=4) were used, with $\tau = 1$. As for the Haar estimation on page 3.3.2, $J = 7$ was used. These pictures are quite encouraging, the most striking point being the huge spike on the third locus of the Afro-Caribbean data. Figure 3.9 shows a close up of this spike. Also plotted are the number of coincident observations at their value, which shows that there is indeed a large cluster of observations at this point, in an area where observations are quite sparse. Notice that this large spike peaks at a density above 1 and that in several places these estimates go negative. This illustrates that these wavelet estimates are not bona fide densities.

3.4 Investigation using the Claw Density

The various thresholding methods proposed are evaluated using the claw density from Marron & Wand (1992). This is a normal mixture density and is illustrated in figure 3.10, together with a kernel density estimate using the S-Plus default bandwidth selection based on a sample of size 200.

First Daubechies least asymmetric $N = 4$ wavelets were used, with the variances estimated directly for the thresholding, as in (3.40). Unfortunately it was not possible to obtain a reasonable looking density estimate, however much the p-value was varied. The only way to get rid of the large number of high frequency spikes was to reduce p to such a small value that none of the claw detail remained either.

Consider instead the Haar wavelet analysis of the same simulated data. The parameters were chosen to be $J = 10$ and $\tau = 1$. Figure 3.10 shows, alongside the kernel density estimate, the Haar wavelet estimate truncated at level 3 (so all the $\widetilde{d}_{jk}$ for $j > 3$ are set to zero). The claws of the density are clearly visible in both of these plots, illustrating that there is evidence in the data for the shape of the density. However, the estimates found by thresholding using the binomial distribution and normal approximation, as described in §3.3.2, again fail to denoise without removing the signal.

True Haar Wavelet Coefficients

For the Haar wavelet the true wavelet coefficients can easily be calculated, since the density is known and the Haar wavelet and scaling function have simple closed form formulae. These coefficients are of the form:

$$d_{jk} = 2^{j/2}\Big(P\big[2^{-j}k \leqslant X < 2^{-j}(k + \tfrac{1}{2})\big] - P\big[2^{-j}(k + \tfrac{1}{2}) \leqslant X < 2^{-j}(k + 1)\big]\Big)\ .$$

The true Haar wavelet coefficients are plotted in the top of figure 3.11, with the empirical coefficients in the bottom. The true and empirical coefficients look quite similar for levels -2 to 3 inclusive. This plot shows that the information is in the empirical coefficients but the thresholding isn't doing well at extracting it.

Various modifications to the thresholding process were tried, in an attempt to "help" it locate the correct coefficients to keep. Firstly levels 7, 8 and 9 were truncated to zero, and the thresholding applied to the remaining levels. Secondly, only levels 2 to 9 inclusive were thresholded, with levels -2 to 1 remaining untouched. Neither modification produced any substantial improvement.

True Variances of the Empirical Haar Wavelet Coefficients

For the Haar basis, it is also possible to calculate the true variances (under H_0) of the empirical coefficients and compare them to the estimated variances $n^{-1}2^{j/2}\widetilde{c}_{jk}$. This

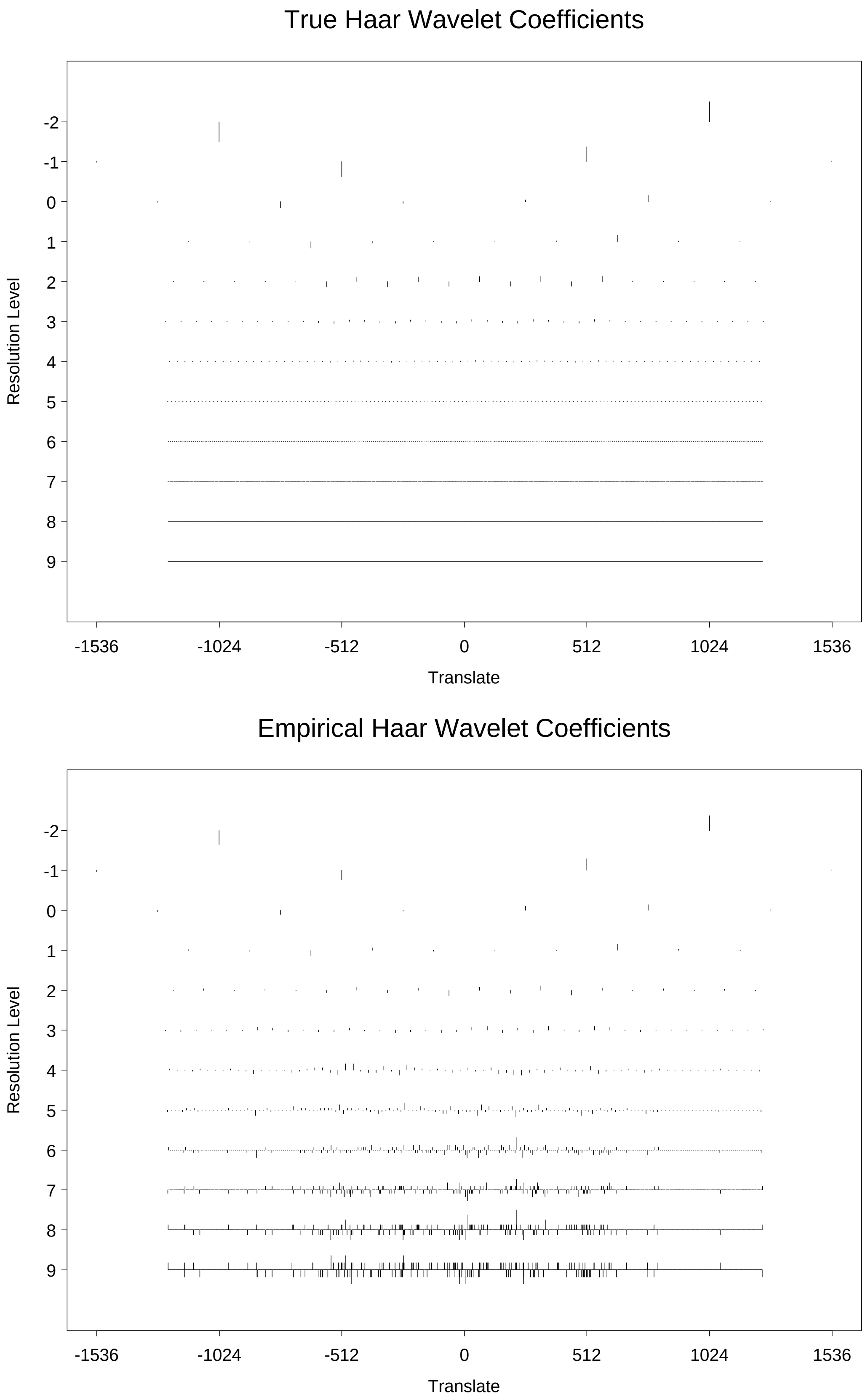


Figure 3.11: True and empirical Haar coefficients for the claw density. A global scale factor is applied to the coefficients in each plot. However, this scale factor is different in each plot. The largest empirical coefficient is 0.32, but the largest true coefficient is 0.24 (to 2 d.p.).

Table 3.4: Two sided p-values of extreme values of m_{jk} for different values of n_{jk}

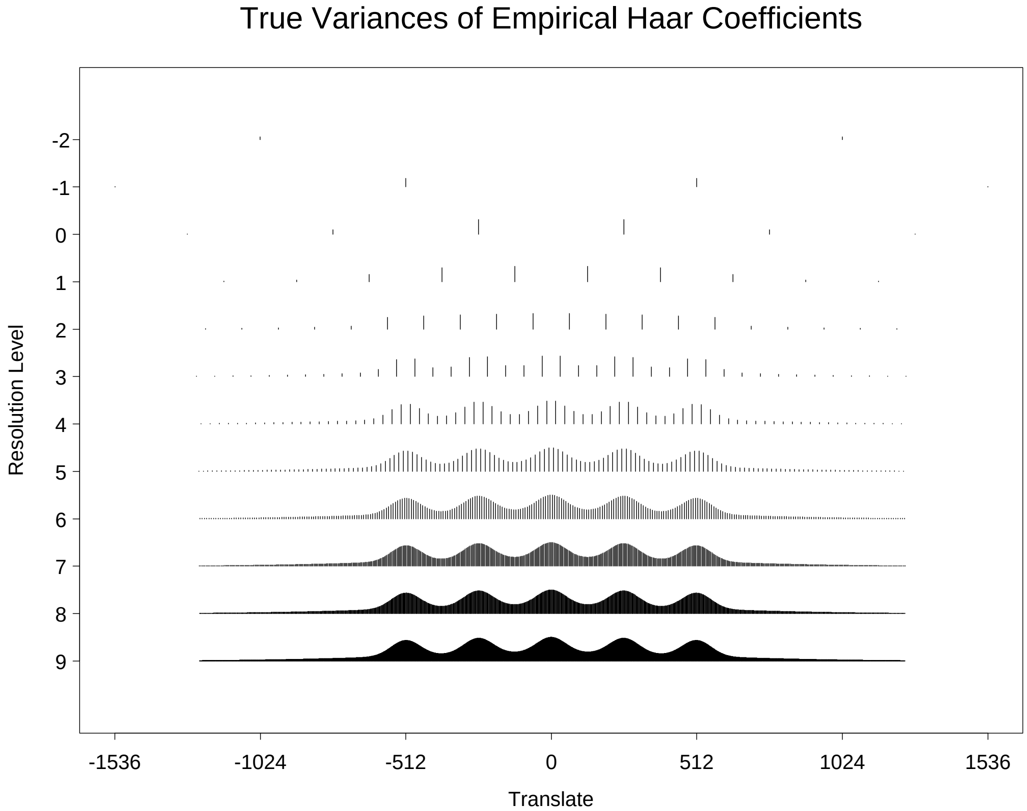
n_{jk}	p-value of $m_{jk}=0$
2	0.5000
3	0.2500
4	0.1250
5	0.0625
6	0.0313
7	0.0156
8	0.0078
9	0.0039
10	0.0020

is shown in figure 3.12. Even down to level 5 the estimates look similar to the true variances. Having calculated the true variances they can be used in the thresholding to give an indication of whether the idea of correcting using the variances is sound, but this does not give good results. It is particularly bad at the boundaries, because the true variance of the outlying coefficients is very small and so magnifies the importance of these coefficients.

Figure 3.13 (page 80) shows the p-values for each coefficient for the normal thresholding using both true and estimated variances. Note that the coefficients that have zero estimated variance will be zero themselves and so have been given a p-value of 1. It can be seen why using the true variance works so poorly as the non-zero high level coefficients have very small p-values. As mentioned previously, the normal approximation can be expected to be worse at the high levels than at the low levels.

Binomial Thresholding and Power

Now return to the binomial testing thresholding. Figure 3.14 (page 81) shows the values of n_{jk}, the number of observations on which each empirical wavelet coefficient is based. This is interesting because if n_{jk} is small, there will be a large probability of observing $m_{jk} = 0$ or n_{jk} which lead to the largest possible wavelet coefficient. Table 3.4 shows the p-values of observing the extreme values of m_{jk} for n_{jk} between 2 and 10. For example, if the thresholding is carried out at the 5% level, n_{jk} must be at least 6 before any values of m_{jk} imply that p_{jk} can be considered significantly different from 0.5. So, considering figure 3.14, no coefficients in levels 6 to 9 inclusive



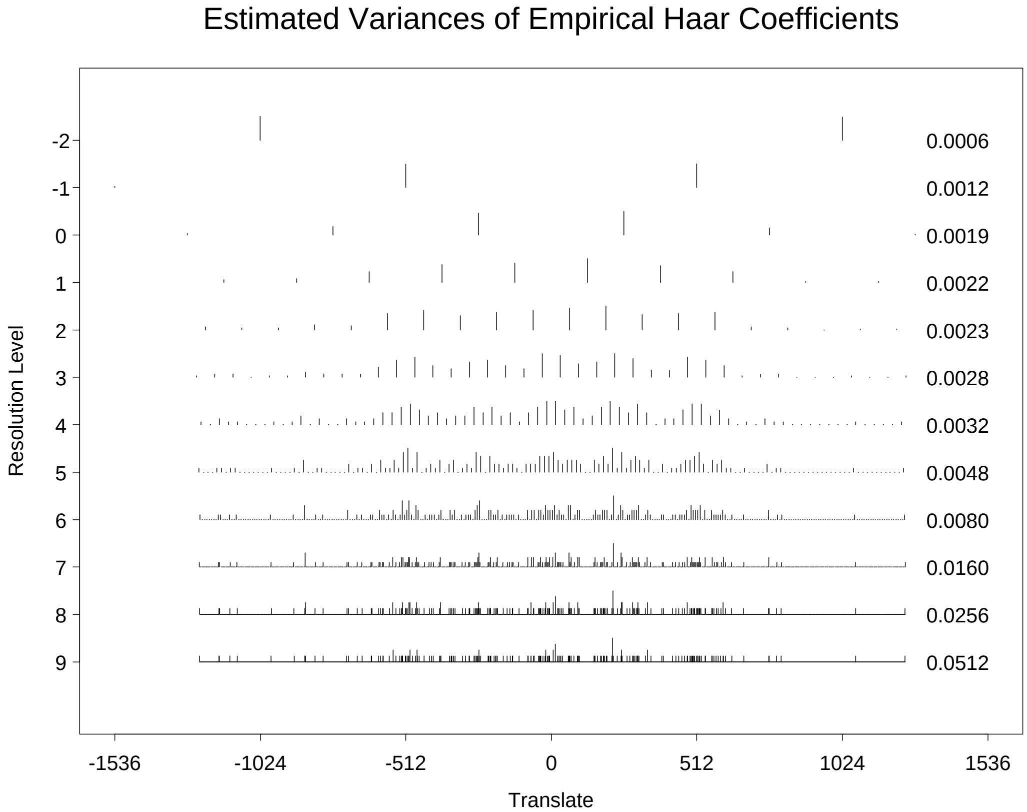


Figure 3.12: True and estimated variances of the empirical Haar coefficients of the claw density. The estimated variances are scaled by level, with the size of the largest coefficient given on the right. The equivalent (global) scale for the true variances is 0.0030 (to 4 d.p.).

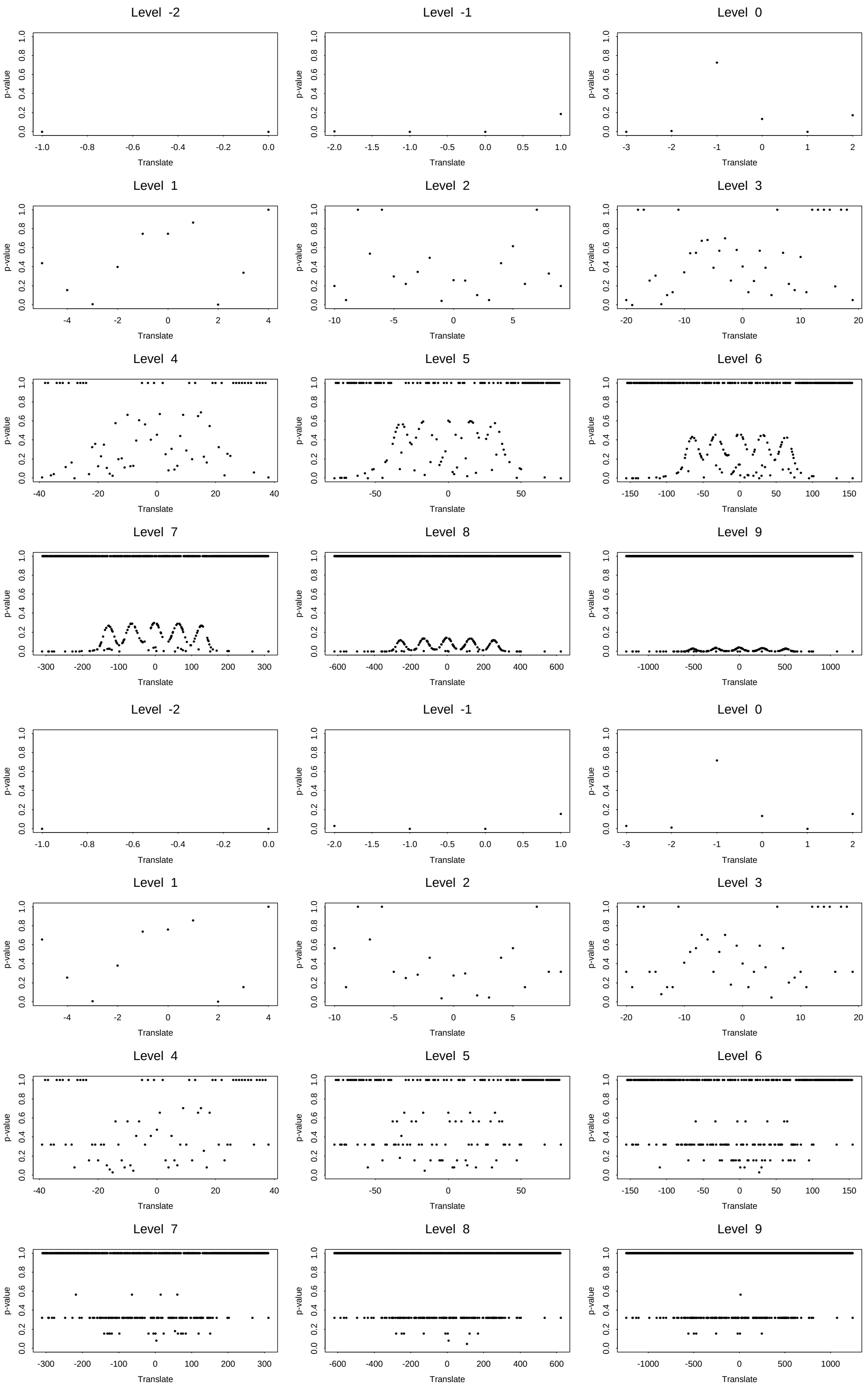


Figure 3.13: p-values for each empirical coefficient under H_0, using the true variances (top four rows) and the estimated variances (bottom four rows) in the normal approximation thresholding.

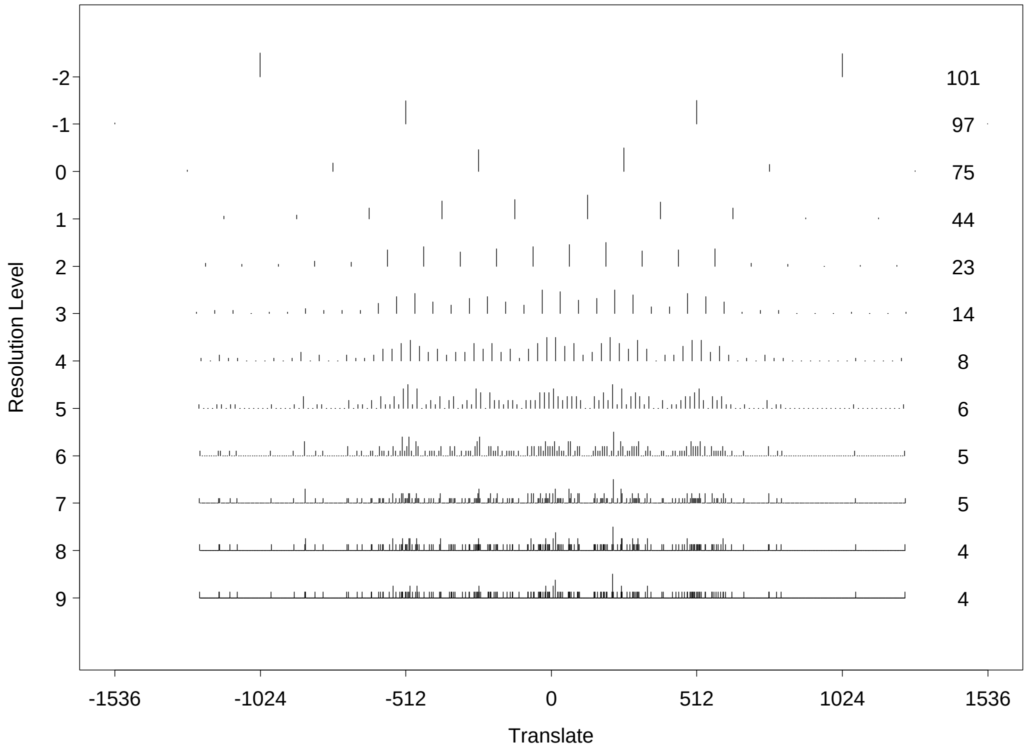


Figure 3.14: n_{jk} for the sample from the claw density. This is the number of observations in the support of ϕ_{jk}. The scaling is different on each level. The column of numbers on the right gives the largest value of n_{jk} on each level.

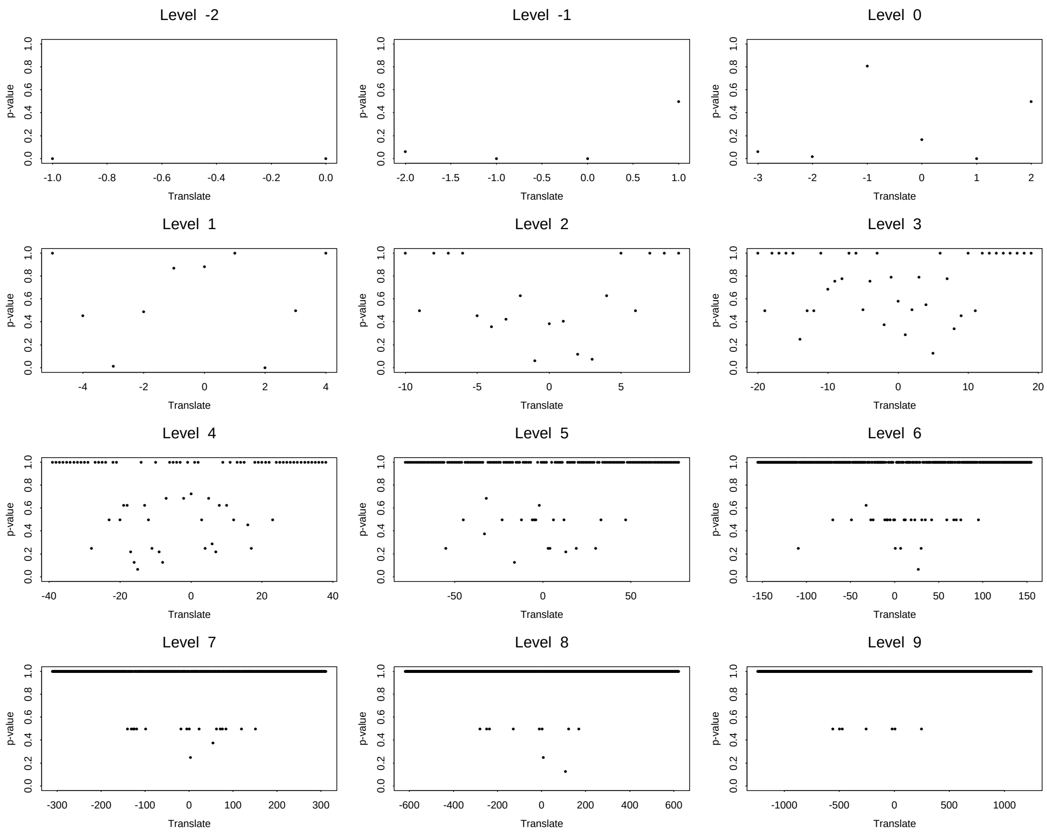


Figure 3.15: p-values for each empirical coefficient under H_0, using the binomial testing.

Table 3.5: Probabilities of type I (α) and type II (β) errors for different values of n_{jk}, where it is required that $\alpha \leqslant 0.05$.

n_{jk}	α	β
10	0.0215	0.7559
20	0.0414	0.3828
30	0.0428	0.1966
40	0.0385	0.1032
50	0.0328	0.0551
60	0.0273	0.0298

can be significantly different from zero at the 5% level, so they will all be removed by the thresholding procedure. This will help ensure that very fine level coefficients, which tend to be based on relatively few observation, are set to zero. However, the binomial thresholding doesn't produce good results either. The problem seems to be in discriminating between signal and noise at the intermediate levels. The p-values for the binomial testing are shown in figure 3.15 (page 81).

It is possible get an idea about the power of all these tests by considering the simple alternative hypothesis $H_1 : p_{jk} = 1/4$. If the probability of a type I error, α, is required to be no greater than 0.05, then table 3.5 shows the values of α and β, the probability of a type II error, for different values of n_{jk}. Note that, while β does decrease as n_{jk} increases, it does not do so monotonically, because for discrete random variables it is not possible to hold α constant. To get β down to about 0.05, n_{jk} must be around 50. For the example from the claw density, none of the coefficients in levels 1 to 9 achieve this. Large values of β mean that there is a large probability of setting to zero empirical coefficients that are really non-zero. The problem is that too many coefficients (and variances) are being estimated from too little data.

3.5 Some Ideas for Choosing the Primary Resolution Level

In this section several possible approaches to choosing the parameter J will be investigated.

3.5.1 Minimum Inter-Point Distance

Recall the discussion in §3.2.1 about the maximum number of non-zero high resolution scaling function coefficients. Suppose J is chosen to be the smallest j for which the $\phi_{jk}(x)$ cover at most one of the data points. This is the smallest j for which there are nw_ϕ coefficients. Then J must be large enough so that the support width of $\phi_{J0}(x)$ is less than the smallest gap between data points. Assume that data are ordered, and let

$$\Delta x_i = x_{i+1} - x_i$$

In real applications there may well be coincident data. Here, only unique points will be considered, so any of the Δx_i that are zero are ignored. Then for each $\phi_{Jk}(x)$ to cover at most one of the data points, J must satisfy:

$$\begin{aligned} w_\phi/p &\leqslant \min\{\Delta x_i : \Delta x_i > 0\} \\ \Rightarrow \qquad p &\geqslant w_\phi/\min\{\Delta x_i : \Delta x_i > 0\} \\ \Rightarrow \qquad J &\geqslant \log_2\Big(w_\phi/\big(\tau \ \min\{\Delta x_i : \Delta x_i > 0\}\big)\Big)\ , \end{aligned}$$

so the *minimum inter-point distance* choice of the primary resolution level is

$$J = \left\lceil \log_2\Big(w_\phi/\big(\tau \ \min\{\Delta x_i : \Delta x_i > 0\}\big)\Big)\right\rceil . \qquad (3.43)$$

Properties of this choice

For J chosen by (3.43), the possibility of recovering the data from the resulting high resolution scaling function coefficients will be investigated.

Suppose vectors containing the identifiably non-zero $\widetilde{c}_{Jk}$ and the corresponding values of k are given, and it is also known which father wavelet was used to construct them. Assume that there were no coincident data points and that the coefficient vector is ordered with increasing k. The non-zero coefficients have the form (3.11). If the vector of coefficients is divided into blocks of length w_ϕ, then the coefficients in each block will have been calculated from the same data point, with each block based on a different point.

Taking each block one at a time and multiplying by n would give $\phi_{Jk}(x_i)$ for w_ϕ consecutive values of k, for one of the data points x_i. Consider the Daubechies wavelet bases. If $w_\phi = 1$ it will not be possible to determine x_i uniquely, because the scaling function for the Haar basis is piecewise constant. It would only be possible to say that $k/p_{J\tau} \leqslant x_i < (k+1)/p_{J\tau}$. However, if $w_\phi \neq 1$, ϕ_{Jk} will not be constant on any subinterval of its support and $\phi_{Jk}(x_i)$ will be available for $w_\phi \geqslant 3$ consecutive values of k. As x varies the values of the $\phi_{Jk}(x)$ will trace out a path in a space of at least three dimensions. Such a path is highly unlikely to cross itself, so it is reasonable to suppose that x_i could be determined uniquely, although this is conjecture. This could be done for each block of coefficients so that the whole data vector could be recovered from the $\widetilde{c}_{Jk}$.

In this sense, for the minimum inter-point distance choice of primary resolution level and using a scaling function with support width at least two, the empirical coefficients contain complete information about the data, if there are no coincident data points. It may be possible to recover the data from the coefficients for smaller J, particularly when using scaling functions with $w \gg 2$, even though some of the coefficients will depend on two or more data points. This requires further investigation.

Practicalities

As noted in §3.2.1, the number of non-zero coefficients increases with j until the limit of nw_ϕ is reached. Since the number of evaluations of ϕ in (3.4) is constant with respect to j, the computational effort required to calculate just the non-zero coefficients shouldn't increase with j. However, the range of k values will increase with j, and so the length of the vector needed to store the coefficients will increase, and for large J this storage requirement will be prohibitive. For example, consider the simulated data set of size 1000 from the discontinuous density used in figure 3.4. With the Haar basis, and $\tau = 1$, the value of J given by (3.43) is 20. This would require a vector with over 44 million elements to store the $\widetilde{c}_{jk}$, although only 1000 of these would be non-zero.

Another version of the `denproj/SFDE` code has been written which only stores the non-zero coefficients and the corresponding values of k, making the minimum inter-point distance choice of J feasible. Since the coefficients come in blocks of size w, the minimum amount of information needed is the first k in each block (the lower bound in (3.7) for each of the data points). This code assumes that the primary resolution is at least that given by (3.43), and so does not perform any summation in the coefficient estimation. This means that it will not work for smaller values of J. However, for the next stage when the DWT is applied the full vector of coefficients will be needed, including all the zeros. Creating and storing this will be impractical whenever it is done. It should, however, be possible to code a sparse version of the DWT which can transform the compactly stored $\widetilde{c}_{Jk}$ and store the resulting $\widetilde{c}_{jk}$ and $\widetilde{d}_{jk}$ efficiently. Without such a sparse transform, (3.43) is of no practical value.

Discounted Minimum Inter-point Distance

One way to make use of the minimum inter-point distance idea is to discard the smallest $\nu\%$ of the inter-point distances and choose J so that the support width of ϕ_{J0} is less than the smallest of the remaining inter-point distances. For the same example used above, $\nu = 1$ gives $J = 14$, $\nu = 5$ gives $J = 11$ and $\nu = 10$ gives $J = 9$. Note that $J = 9$ still requires a vector with 21691 elements. Of course, it is not possible to determine what value of ν will give a practical value of J.

3.5.2 Maximum Observable Frequency

The idea that will be discussed in this section is to choose J so that the wavelets can capture the finest detail that can be determined from the data.

Frequencies in the Wavelets

Suppose that the maximum frequency contained in the data, f_h, is known. How does this help us choose the primary resolution level? To investigate the frequency properties of *mother* wavelets consider the Fourier transform:

$$\hat{\psi}(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \psi(t) e^{-i\omega t} dt \ .$$

The commonly used compactly supported wavelets are not compactly supported in the frequency domain, but the Fourier transform will decay reasonably fast if the wavelets

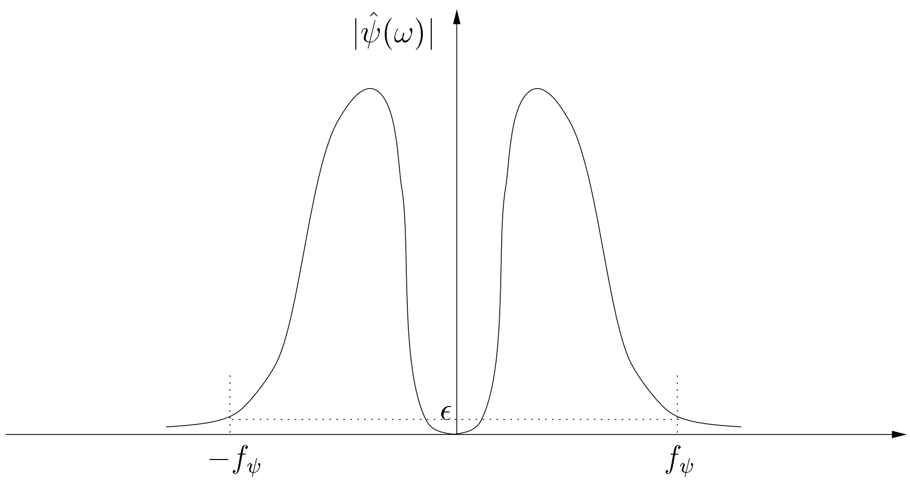


Figure 3.16: Illustration of bounds on the frequencies in a mother wavelet.

are regular. Define a bound on the frequencies in the wavelet by:

$$f_\psi = \sup\{|\omega| : |\widetilde{\psi}(\omega)| > \epsilon\}$$

for some small ϵ. This is illustrated in Figure 3.16. $\hat{\psi}$ will have this sort of shape as it is representative of a band-pass filter. The value of ϵ can be chosen by insisting that, say, only 5% of the power is for frequencies beyond f_ψ. The choice of ϵ is an efficiency issue, since if f_ψ is too small (ϵ is too large) then we will end up choosing a larger primary resolution than is really necessary. However, it is important to ensure that ϵ isn't too *small*, otherwise there will be little power in the wavelet at high frequencies and the high frequencies in the data will not be well modelled.

Now consider the Fourier transform of ψ_{jk}, noting that frequency properties are independent of the translation factor k:

$$\begin{aligned}
\hat{\psi}_{j0}(\omega) &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \psi_{j0}(t) e^{-i\omega t} dt \\
&= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} p_{j\tau}^{1/2} \psi(p_{j\tau} t) e^{-i\omega t} dt \\
&= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} p_{j\tau}^{-1/2} \psi(t) e^{-i(\omega/p_{j\tau})t} dt \\
&= p_{j\tau}^{-1/2} \hat{\psi}(\omega/p_{j\tau})
\end{aligned}$$

So $|\hat{\psi}_{j0}|$ can be represented by Figure 3.17.

So, j should be large enough so that $p_{j\tau} f_\psi > f_h$ ($\Rightarrow j > \log_2(f_h/\tau f_\psi)$). Note that $\forall \omega > p_{j\tau} f_\psi$ we have $\hat{\psi}_{j0}(\omega) < p_{j\tau}^{-1/2} \epsilon$, not $\hat{\psi}_{j0}(\omega) < \epsilon$. Of course, choosing the primary

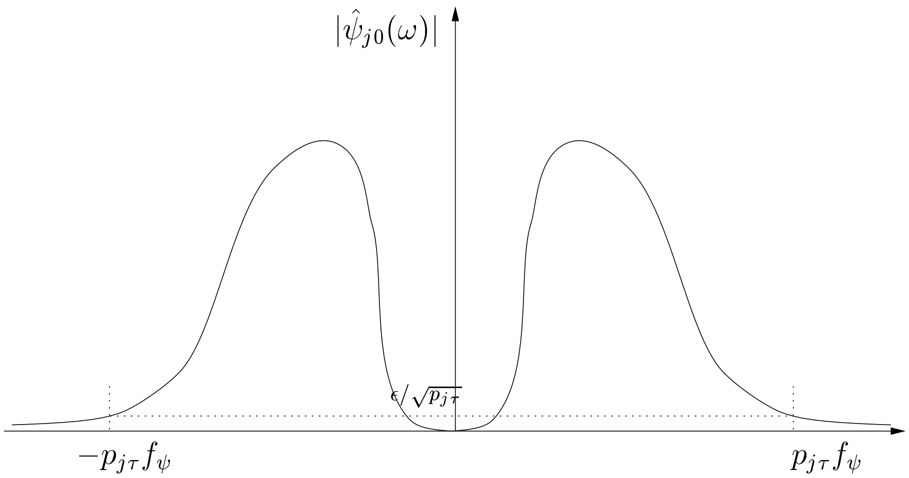


Figure 3.17: Bounds at level j corresponding to those in figure 3.16.

resolution involves choosing τ as well as j. To ensure that the primary resolution is big enough for the wavelets to contain all possible frequencies in the signal, the easiest thing to do is set $\tau = 0.5$ and choose $j > \log_2(2f_h/f_{\mathrm{w}})$.

To choose the resolution J at which to compute the empirical scaling function coefficients, note that the choice of j above ensures that the space W_j will contain any possible reconstruction of the density function, and that $V_{m+1} = V_m \oplus W_m$, so $W_m \subset V_{m+1}$. Hence the space V_{m+1} will also contain all possible reconstructions of the density estimate and $J = j + 1$ should be taken as the primary resolution level.

Frequencies in the Data

It remains now to try and determine from the data what the highest frequency that can be observed is. First note that when data in a regression context (x_i, y_i) are sampled, the sampling method imposes a limit (the Nyquist rate) on the frequencies that can be observed in the signal. See, for example, Priestley (1981). For equally spaced data, with the observations a distance Δx apart (i.e. $x_{i+1} - x_i = \Delta x$, $\forall i$) the Nyquist rate is

$$f_h = \pi/\Delta x.$$

Although this can be generalised for unequally spaced data, there is no obvious extension to the one dimensional samples of the density estimation problem. Unless f_h can be determined, this idea for choosing the primary resolution is of no use.

3.5.3 Energy

The possibility of choosing J so that the $\widetilde{c}_{Jk}$ achieve maximum energy is investigated in this section.

The energy of the estimated coefficients is

$$\sum_k \widetilde{c}_{jk}^2 \ . \tag{3.44}$$

However, recall that the estimated coefficients of the density function are the true father wavelet coefficients of the empirical density function. The empirical density function is a weighted sum of delta functions, which have infinite energy. This means that the empirical density function is a signal with infinite energy, so the sum of squared wavelet coefficients, and hence (3.44) will be unbounded.

Thinking about this heuristically, recall from p57 that for the minimum inter-point distance choice of primary resolution there will be a constant number of non-zero $\widetilde{c}_{Jk}$, each of which will look like $n^{-1}\phi_{Jk}(X_{i_k})$. So each non-zero

$$\begin{aligned} \widetilde{c}_{Jk}^2 &= n^{-2}\phi_{Jk}^2(X_{i_k}) \\ &= n^{-2}2^J\phi^2(2^J X_{i_k} - k) \ . \end{aligned} \tag{3.45}$$

Since there are a constant number of these, it seems reasonable to conclude that $\sum_k \widetilde{c}_{Jk}^2$ will grow like 2^J, which is what seems to happen in practice.

3.5.4 p-values

Recall the discussion about p-values on page 78. Another possibility for choosing the primary resolution level is to choose the highest level at which a significant coefficient can be found in the Haar analysis for some sensible p-value. Using the example of the simulated claw data, for a p-value of 0.05, J would be chosen to be 6.

3.6 Conclusions and Further Work

The calculation of the empirical wavelet coefficients from the data has been efficiently programmed and is available for use in `S-Plus`. However, the thresholding of these coefficients to smooth the density estimate has not been satisfactorily worked out, although several ideas have been programmed and tested.

The main area of development is, therefore, determining a suitable thresholding scheme. Further investigation into those proposed in §3.3 may lead to a thresholding scheme with better performance. It is possible that the problems experienced with the thresholding are due to trying to estimate too much information (wavelet coefficients and variances) from too little data. This suggests that a Bayesian approach is worth investigating, as a way of adding some extra (prior) information.

If the thresholding is successfully sorted out, the choice of the parameters J and τ could then be considered in more detail. Several ideas for the choice of J have been discussed already, but perhaps the simple idea of choosing J to be the smallest j such that there are more $\widetilde{c}_{jk}$ than there data points may be worth investigating. The parameter τ has not been discussed in much detail, and the choice of this needs investigating. Although not computationally efficient, J, τ and the threshold could be chosen in tandem using a cross-validation approach.

Chapter 4

Wavelet Surface Estimation with Unequally Spaced Data

This chapter is concerned with the extension of two dimensional wavelet regression (or wavelet surface estimation) to allow data that is not observed on a regularly spaced grid to be analysed. This will be done by developing some of the ideas in Kovac & Silverman (2000), which is discussed in §2.4. Firstly, §4.1 will introduce the dataset that will be used as a running example, and then discuss some properties of the Voronoi diagram and the Delaunay triangulation, as these constructs will be used in the remainder of the chapter. Spatial interpolation will then be the subject of §4.2 and the extension of wavelet smoothing to irregularly spaced data will be thoroughly investigated in §4.3. The possibility of selecting the threshold by cross validation will be considered in §4.4, which will be followed by some concluding remarks in §4.5.

4.1 Introduction

4.1.1 Preliminaries and Notation

Suppose f is a function of two dimensions, and that irregularly spaced data points are observed over a finite region, or *window*, S of $\mathbb{R}^2$:

$$z_i = f(x_i, y_i) + \varepsilon_i \ , \quad i = 1, \dots, n \ .$$

The set of three dimensional data points is labelled $P = \{p_1, \dots, p_n\}$, where the points are of the form $p_i = (x_i, y_i, z_i)$. The locations $\boldsymbol{x}_i = (x_i, y_i)$ are called *data sites* and the

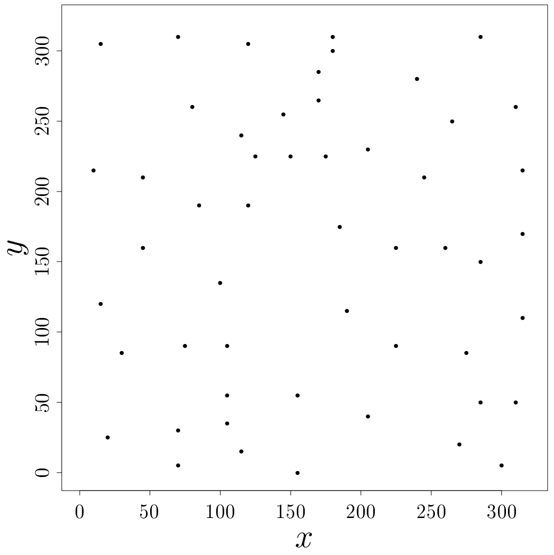


Figure 4.1: Data sites for the points in the `topo` dataset.

observations z_i *data values.* The set of data sites is labelled $X = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$.

Example 4.1 *The dataset* `topo`*, presented on page 312 of Davis (1973) will be used as a running example. It contains 52 measurements of topographic height (in feet) within a square of side 310 feet (labelled in 50 feet units). The range of x (east-west) coordinates is* $[0.2, 6.3]$ *(corresponding to* $[10, 315]$ *in feet) and the range of y (north-south) coordinates is* $[0.0, 6.2]$ *(*$[0, 310]$ *in feet), measured from an origin in the southwest corner of the region. The measurements were recorded by a surveying class, at an unspecified location. The observations are not taken on any sort of regular grid; the data sites are plotted in figure 4.1. This dataset is also discussed in Chapter 15 of Venables & Ripley (1994), and is contained in the accompanying* `S-Plus` *library.*

To use the fast 2D DWT, points with data sites on a regular lattice are required. Following Kovac & Silverman (2000), it is proposed that such regularly spaced data is obtained by interpolating the observed data points onto a suitable lattice, which can be defined on the unit square as follows:

$$\breve{x}_i = (i + \tfrac{1}{2})2^{-J} , \quad \text{for } 0 \leqslant i \leqslant 2^J - 1 , \tag{4.1}$$

$$\breve{y}_j = (j + \tfrac{1}{2})2^{-J} , \quad \text{for } 0 \leqslant j \leqslant 2^J - 1 . \tag{4.2}$$

For data not on the unit square a suitably scaled and shifted version of this can be used. See figure 4.11 for an example.

Many methods of spatial interpolation are based on the Voronoi diagram, and the associated Delaunay triangulation, so it is worthwhile reviewing some basic definitions and properties of these constructs here. Some other triangulations will be discussed later, in §4.2.

4.1.2 Voronoi Diagrams

A detailed discussion of Voronoi diagrams, their properties and applications can be found in Okabe *et al.* (1992). Only two dimensional diagrams will be discussed here, although the ideas extend to higher dimensions. Voronoi diagrams are, perhaps, more commonly called Dirichlet tessellations. The basic geometrical idea is that given a set of points in the real plane $\Pi = \{\pi_1, \ldots, \pi_n\}$, where $\pi_i = (x_i, y_i)$, the real plane (or the subset $S \subset \mathbb{R}^2$) is divided into a number of Voronoi polygons. Each polygon is based on one of the points in Π, and contains all locations in $\mathbb{R}^2$ (or S) that are closer to that point than to any other member of Π. The totality of Voronoi polygons forms the Voronoi diagram. More formally, the region

$$V_S(\pi_i) = \{\boldsymbol{x} \in S : \| \boldsymbol{x} - \pi_i \| \leqslant \| \boldsymbol{x} - \pi_j \| \text{ for } j \neq i\}$$

is called the *Voronoi polygon* (or *cell*) associated with π_i. The set

$$\mathcal{V}_S(\Pi) = \{V_S(\pi_1), \ldots, V_S(\pi_n)\}$$

is called the *Voronoi diagram* generated by Π and is a partition of S.

The edges of the Voronoi polygons are called *Voronoi edges*. The end point of a Voronoi edge is called a *Voronoi vertex*. When there exists at least one Voronoi vertex at which four or more Voronoi edges meet $\mathcal{V}_S(\Pi)$ is *degenerate*; otherwise $\mathcal{V}_S(\Pi)$ is *non-degenerate*. Degenerate Voronoi diagrams often appear when the generator points are regularly spaced. Two points π_i, π_j are described as *contiguous* if $V_S(\pi_i)$ and $V_S(\pi_j)$ share a Voronoi edge.

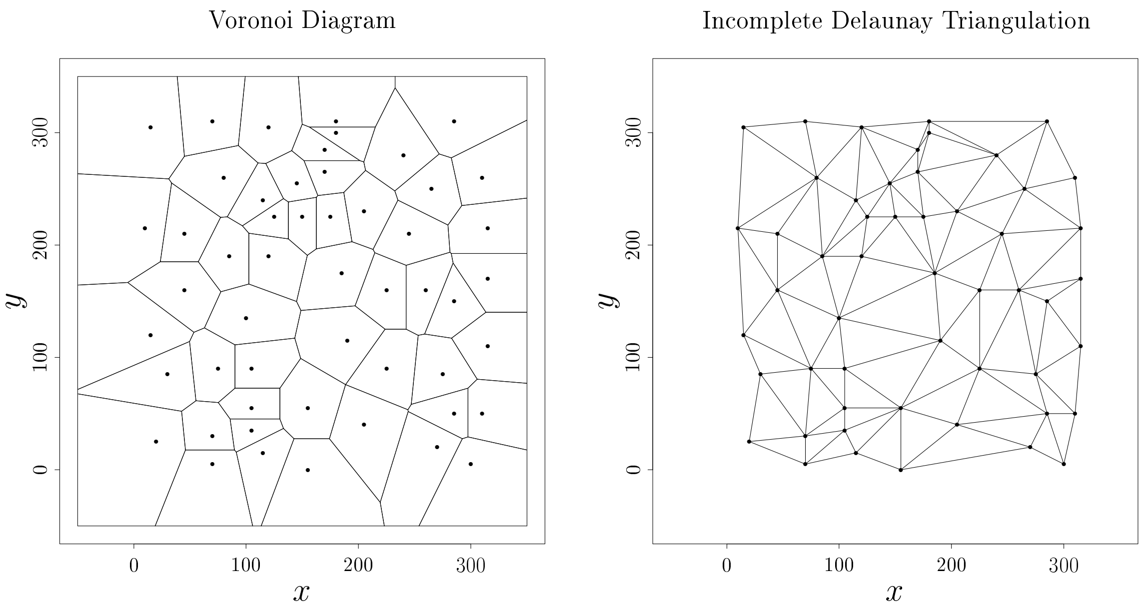


Figure 4.2: Voronoi diagram $\mathcal{V}_S(X)$ generated by the data sites of the `topo` dataset (left) and the incomplete Delaunay triangulation generated by joining the data sites of points that are contiguous in $\mathcal{V}_S(X)$ (right).

Example 4.2 *The Voronoi diagram $\mathcal{V}_S(X)$ generated by the data sites of the points in the* `topo` *dataset, with*

$$S = \left\{(x, y) \in \mathbb{R}^2 : -1 \leqslant x \leqslant 7, -1 \leqslant y \leqslant 7\right\} ,$$

is shown in figure 4.2. The restrictions on x and y are $[-50, 350]$ in feet.

4.1.3 Delaunay Triangulation

If the points in Π are not collinear, then the *Delaunay triangulation* $\mathcal{D}(\Pi)$ can be generated by joining all points that are contiguous in $\mathcal{V}_{\mathbb{R}^2}(\Pi)$. If $\mathcal{V}_{\mathbb{R}^2}(\Pi)$ is degenerate then the above procedure will contain at least one polygon with four or more vertices. In this case the Delaunay triangulation is obtained by inserting non-intersecting line segments joining the vertices to partition the polygon into triangles. It should be noted that the Delaunay triangulation is not unique when $\mathcal{V}_{\mathbb{R}^2}(\Pi)$ is degenerate.

Example 4.3 *The Delaunay triangulation $\mathcal{D}(X)$ generated by the data sites of the points in the* `topo` *dataset is shown in the left hand plot in figure 4.3. To illustrate the*

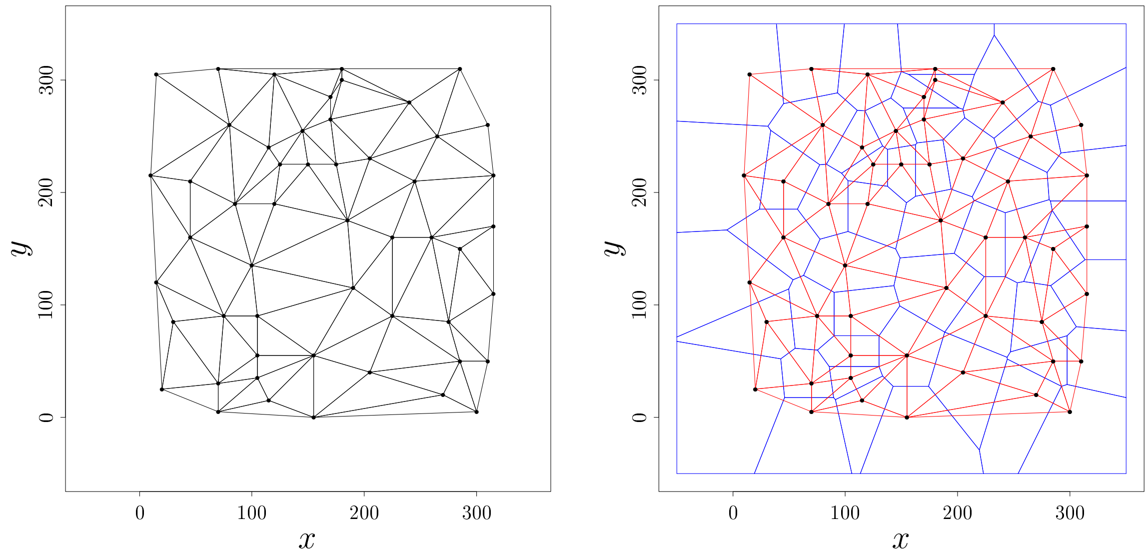


Figure 4.3: The Delaunay triangulation generated by the `topo` dataset (left) and the Voronoi diagram (blue) and Delaunay triangulation (red) generated by the `topo` dataset overlaid (right).

construction, the Voronoi diagram and Delaunay triangulation are shown overlaid in the right hand plot in figure 4.3.

The Delaunay triangulation has a number of interesting properties that are worth mentioning. An important property in the construction of algorithms for finding $\mathcal{D}(\Pi)$ is the *max-min angle* criterion, which Ripley (1981) refers to as the *Lawson* criterion, after Lawson (1977). This says that the diagonal of every strictly convex quadrilateral Q formed by merging two adjacent triangles in $\mathcal{D}(\Pi)$ is chosen so that it maximises the minimum interior angle of the resulting two triangles (see figure 4.4). Sibson (1978) shows that the Delaunay triangulation uniquely satisfies this criterion. Lawson (1977) shows that this criterion is equivalent to the *circle* criterion. If k is a circle passing through any three vertices of Q and the fourth vertex is interior to k the diagonal linking this vertex to the opposite one is selected. If the fourth vertex is exterior to k, the other diagonal is selected and if it lies on k either diagonal is selected (see figure 4.4). Note that the fourth vertex lying on k is equivalent to the minimum interior angle of two triangles being the same wherever the diagonal is placed in the Lawson criterion and is equivalent to Q resulting from a degenerate part of $\mathcal{V}_{\mathbb{R}^2}(\Pi)$. Note that $\mathcal{D}(\Pi)$ is unique for non-degenerate $\mathcal{V}_{\mathbb{R}^2}(\Pi)$.

Although not of huge significance here, it is interesting to summarise the debate over the *Pitteway* criterion. This says that any point within any triangle should have as its closest data point one of the vertices of that triangle. McLain (1976) attributes this criterion to Pitteway and describes it as an *optimal partition*. He then describes an algorithm which he claims finds such a triangulation. Sabin (1976) points out that there exist configurations of data points for which no triangulations are Pitteway and that since McLain's algorithm produces triangulations for such configurations it cannot have the claimed property of Pitteway optimality. Lawson (1977) shows that McLain's algorithm in fact finds the Delaunay triangulation. Okabe *et al.* (1992) give, in Chapter 2, a proof that $\mathcal{D}(\Pi)$ is a Pitteway triangulation if and only if every internal Delaunay edge crosses the associated Voronoi edge of $\mathcal{V}_{\mathbb{R}^2}(\Pi)$. They also give a counter example to show that if Π admits a Pitteway triangulation, it is not necessarily a Delaunay triangulation. Their counter example is a set of four points at the vertices of a rhombus. Both possible triangulations are Pitteway, but only one is Delaunay. They conclude that if a Pitteway triangulation of Π exists, it is not necessarily unique and does not necessarily coincide with the Delaunay triangulation. However, this still leaves the question as to whether or not if Π admits at least one Pitteway triangulation, the Delaunay triangulation is Pitteway.

Okabe *et al.* (1992) consider on page 354 the relationship between the Delaunay triangulation and other triangulations, particularly the *minimum length triangulation* $\mathrm{MLT}(\Pi)$ and the *greedy triangulation* $\mathrm{GT}(\Pi)$. Let $|\mathcal{T}(\Pi)|$ denote the sum of the lengths of the edges in a general triangulation $\mathcal{T}(\Pi)$. Then $\mathrm{MLT}(\Pi)$, originally proposed by Bengtsson & Nordbeck (1964), is the triangulation that minimises $|\mathcal{T}(\Pi)|$. Okabe *et al.* (1992) assert, without reference or discussion, that $\mathrm{MLT}(\Pi)$ is unique for general sites. Watson & Philip (1984) note that this point did not, at that time, appear to have been investigated. $\mathrm{GT}(\Pi)$ is defined by demanding that no edge should be included in $\mathcal{T}(\Pi)$ if there is a shorter edge that which would properly intersect it. It can be found by considering all possible edges in order of increasing length and inserting an edge if no edge already exists that intersects it. If no pair of edges has the same length, $\mathrm{GT}(\Pi)$ is unique. Lloyd (1977) provides counter examples to show that, although algorithms based on the criteria for $\mathcal{D}(\Pi)$ and $\mathrm{GT}(\Pi)$ were claimed to find $\mathrm{MLT}(\Pi)$, these three

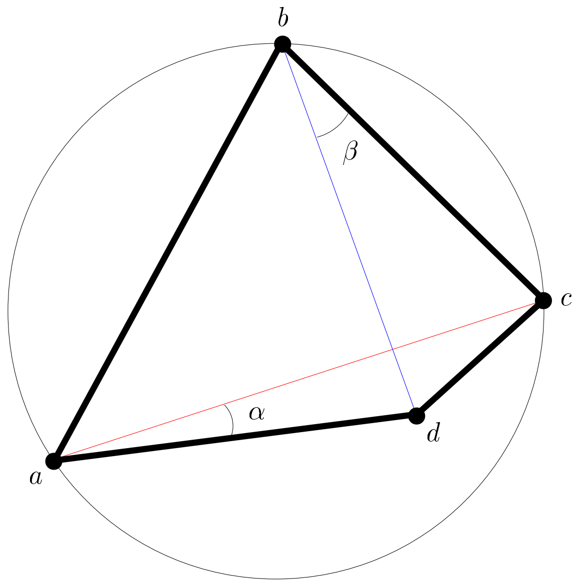


Figure 4.4: Illustration of the max-min angle and the circle criteria. The quadrilateral $abcd$ is split into triangles by the blue line because angle β (the smallest of the six angles in $\triangle abd$ and $\triangle bcd$) is larger than α (the smallest of the six angles in $\triangle abc$ and $\triangle acd$). Also, notice that point d is interior to the circle passing through a, b and c.

triangulations do not necessarily coincide.

4.1.4 Computation of Voronoi Diagrams and Delaunay Triangulations

Green & Sibson (1978) present an algorithm for computing Voronoi diagrams. Their algorithm is a recursive method for computing contiguities. (The *contiguities* of a point π_i are the points in $\Pi \setminus \{\pi_i\}$ with which π_i is contiguous.) All other properties, such as the position of the Voronoi vertices are easily calculated from the list of contiguities and data sites. Their Fortran code was used to compute and plot the Voronoi diagram in figure 4.2. It is not possible to compute $\mathcal{V}_{\mathbb{R}^2}(\Pi)$, so the user must supply the boundaries of the window S. Consider for a moment the difference between $\mathcal{V}_{\mathbb{R}^2}(\Pi)$ and $\mathcal{V}_S(\Pi)$. If the assumption is made that the window contains all the data sites, then the windowing merely obscures everything outside of S. This may affect the list of contiguities as two points that are contiguous in $\mathcal{V}_{\mathbb{R}^2}(\Pi)$ will not be contiguous in $\mathcal{V}_S(\Pi)$ if their shared Voronoi edge in $\mathcal{V}_{\mathbb{R}^2}(\Pi)$ lies completely outside S. The authors note that since data are usually collected within a finite window, the windowing reflects and avoids introducing contiguities that would probably be eliminated if the data had been collected over a larger area. However, since $\mathcal{D}(\Pi)$ can be defined by joining together contiguous points in $\mathcal{V}_{\mathbb{R}^2}(\Pi)$, if Green and Sibson's algorithm is used to construct $\mathcal{D}(\Pi)$ by joining contiguous points in $\mathcal{V}_S(\Pi)$ some lines may be omitted so that $\mathcal{D}(\Pi)$ no longer fills $\mathrm{CH}(\Pi)$, the convex hull of Π. This is illustrated in figure 4.2. Two segments of $\mathrm{CH}(\Pi)$ are missing on the bottom, and one each on the right and top. Since it is impossible to guarantee that all Voronoi edges of $\mathcal{V}_{\mathbb{R}^2}(\Pi)$ lie partially in S before computing the diagram, computing the Delaunay triangulation using Green and Sibson's algorithm is a matter of trial and error, which is not very satisfactory.

Akima (1978b) presents an algorithm for directly computing the Delaunay triangulation which exploits the max-min angle criterion. To triangulate the x-y plane he first connects the closest pair of data sites. Points are then added, one at a time, in ascending order of the distance of the new data site from the midpoint of the closest pair of data sites. Each time a new point is added, triangles are constructed by connecting the new data site to the old sites that are visible from the new one and exchanging triangles whenever required by the max-min angle criterion. (Two data sites are visible from each other if they can be joined without the resulting line crossing any lines that already exist.)

The ordering in adding new points ensures that a new data site always lies outside the polygon constructed with the old sites. To see this note that the new data site lies outside the circle that is centred at the midpoint of the closest two data sites and passes through the most recently added data site and that this circle contains the current polygon. This helps minimise the number of exchanges that are necessary.

4.2 Spatial Interpolation

Recall that in order to extend the Kovac-Silverman method to 2D, the data must first be interpolated to a regular grid. Useful discussions of interpolation of three dimensional data appear in Chapter 4 of Ripley (1981) and Chapter 6 of Okabe *et al.* (1992). Spatial interpolation involves finding a function $\check{f}(\boldsymbol{x})$ that best represents the entire surface and predicts values of z for locations other than P in S.

Of particular interest is the evaluation of $\check{f}$ at the vertices of the lattice (4.1) & (4.2), to give interpolated values $\check{z}_{ij} = \check{f}(\check{x}_i, \check{y}_j)$, for $0 \leqslant i, j \leqslant 2^J - 1$. For this application to wavelet methods, the interpolant should be *exact* (i.e. $\check{f}(x_i, y_i) = z_i$) and preferably *local* (i.e. $\check{f}(\boldsymbol{x})$ depends only on "nearby" points in P), in order to preserve as much as possible of the information in the original data. Local interpolants represent $f(x, y)$ at an arbitrary site $\boldsymbol{x} = (x, y)$, as a weighted (usually linear) function of the data values of a set of points $D(\boldsymbol{x})$ (where $D(\boldsymbol{x}) \subset P$) with data sites "near" to $\boldsymbol{x}$, so that

$$\check{z} = \check{f}(x, y) = \sum_{i: p_i \in D(\boldsymbol{x})} w_i z_i \; . \tag{4.3}$$

Linear interpolants, in which the w_i are a linear function of x and y, will mostly be considered because of their relative simplicity. Note also that the aim is to smooth the data by wavelet shrinkage, not by applying a complicated interpolation procedure.

Linear interpolation of 3D data is more complicated than that of 2D data. For 2D data, the data sites can be ordered and so provide an obvious partition of the real line into segments, over which linear pieces can then be fitted. For three dimensional points, the data sites lie in $\mathbb{R}^2$ and there are many ways in which they can be joined to divide up the real plane. However, there is no one obvious, easy to compute, partition of $\mathbb{R}^2$.

The majority of this section will be concerned with triangular methods of interpolation, but first a brief description of Natural Neighbour interpolation will be given.

4.2.1 Natural Neighbour Interpolation

Natural neighbour interpolation was introduced in Sibson (1980a) and Sibson (1980b). In $\mathcal{V}_S(X \cup \boldsymbol{x})$, $\boldsymbol{x}$ has associated Voronoi polygon $V_S(\boldsymbol{x})$. The points of P whose data sites are contiguous to $\boldsymbol{x}$ in $\mathcal{V}_S(X \cup \boldsymbol{x})$ are the *natural neighbours* of $\boldsymbol{x}$. Natural neighbour interpolation defines $D(\boldsymbol{x})$ in (4.3) to be the set of natural neighbours of $\boldsymbol{x}$.

Sibson (1980a) also uses natural neighbours to define the weights w_i. In $\mathcal{V}_S(X \cup \boldsymbol{x})$, $V_S(\boldsymbol{x})$ can be exhaustively subdivided into regions $V_S((\boldsymbol{x}, \boldsymbol{x}_j))$ such that the first and second nearest points of $(X \cup \boldsymbol{x})$ from any location in $V_S((\boldsymbol{x}, \boldsymbol{x}_j))$ are $\boldsymbol{x}$ and $\boldsymbol{x}_j$ respectively. $V_S((\boldsymbol{x}, \boldsymbol{x}_j))$ can also be thought of as the intersection of $V_S(\boldsymbol{x})$ in $\mathcal{V}_S(X \cup \boldsymbol{x})$ with $V_S(\boldsymbol{x}_j)$ in $\mathcal{V}_S(X)$.

Now define $\kappa(\boldsymbol{x}, \boldsymbol{x}_j)$ to be the area of $V_S((\boldsymbol{x}, \boldsymbol{x}_j))$. (In the special case $\boldsymbol{x} = \boldsymbol{x}_j$ set $\kappa(\boldsymbol{x}, \boldsymbol{x}_j) > 0$ and $\kappa(\boldsymbol{x}, \boldsymbol{x}_i) = 0$ for $i \neq j$.) In fact Sibson goes on to use the normalised

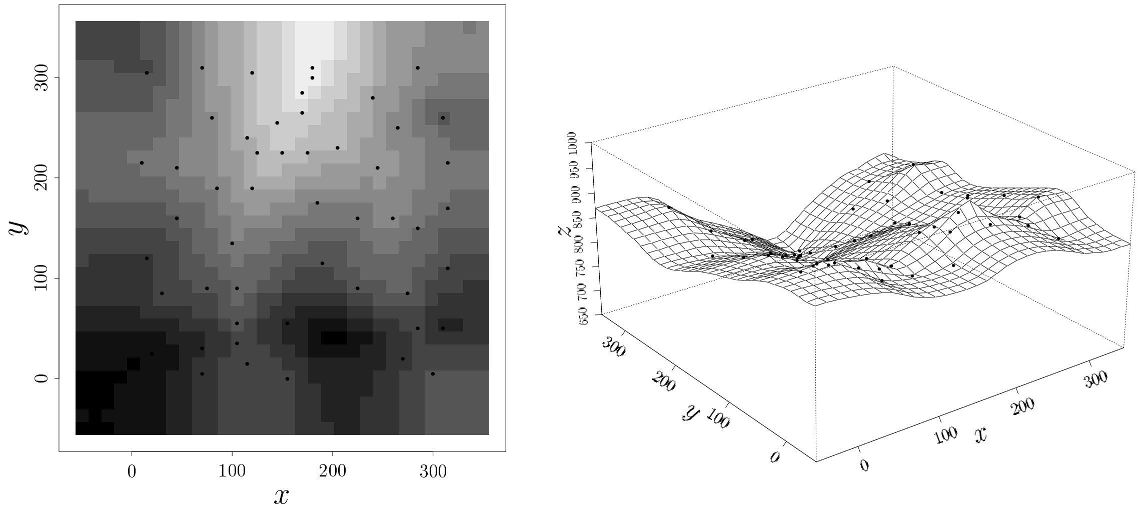


Figure 4.5: Grey scale plot (left) and perspective plot (right) of the natural neighbour interpolant of the `topo` dataset.

values:

$$\lambda(\boldsymbol{x}, \boldsymbol{x}_j) = \frac{\kappa(\boldsymbol{x}, \boldsymbol{x}_j)}{\sum_i \kappa(\boldsymbol{x}, \boldsymbol{x}_i)} \ .$$

These λ's are called the *local coordinates* of S.

$\lambda(\boldsymbol{x}, \boldsymbol{x}_j)$ is a continuous function of $\boldsymbol{x}$ which is continuously differentiable everywhere except for the data sites. A C^0 interpolant can therefore be defined as:

$$\check{f}(\boldsymbol{x}) = \sum_j \lambda(\boldsymbol{x}, \boldsymbol{x}_j) z_j \ . \tag{4.4}$$

This interpolant is local and exact, but not linear in x and y. If the data values come from a linear function $z = \alpha + \beta^T \boldsymbol{x}$ then (4.4) correctly reproduces it. Sibson (1980a) goes on to define a more complicated C^1 interpolant, that need not concern us here.

Example 4.4 *The natural neighbour interpolant of the* `topo` *dataset is shown in figure 4.5. The* `S-Plus` *functions for plotting the grey-scale and perspective plots require data points over an evenly spaced grid, as the wavelet methods discussed later will do. The grid points are as in (4.1) and (4.2) with $J = 5$ and scaled and translated so that the range of values in both the x and y directions is $[-1, 7]$.*

4.2.2 Triangular Methods of Interpolation

Triangular based methods of interpolation consist of three basic steps. First a triangulation $\mathcal{T}(X)$ must be constructed using the data sites as vertices. Then the triangle $T(\boldsymbol{x})$ of $\mathcal{T}(X)$ which contains $\boldsymbol{x}$ must be identified, and $D(\boldsymbol{x})$ defined to be the vertices of $T(\boldsymbol{x})$. Finally, the interpolated value $\check{z}$ at $\boldsymbol{x}$ is computed using (4.3).

Triangulation

It is desirable that $\mathcal{T}(X)$ should avoid triangles with two highly acute interior angles and that triangles be as equiangular as possible. This is because elongated triangles involve greater separation of data sites and reduce the ability of the triangulation to reflect local variation. Also, it is clearly desirable that the triangulation is Pitteway (see p.96) if possible, as it makes sense for the interpolated value at any location to depend on its nearest neighbour in P.

One problem is that $\mathcal{T}(X)$ is only defined within $\mathrm{CH}(X)$ and so usually does not cover S. Another problem is that $\mathcal{T}(X)$ invariably contains some undesirable elongated triangles associated with the boundary of $\mathrm{CH}(X)$, however $\mathcal{T}(X)$ is defined. One way to extend $\mathcal{T}(X)$ to cover S is to divide $S\backslash\mathrm{CH}(X)$ into trapezoids and triangles by inserting perpendiculars to the exterior edges of $\mathrm{CH}(X)$ at each of the vertices. Figure 4.6 illustrates this with the `topo` data set. Note that this extension preserves the original triangulation $\mathcal{D}(X)$. The value of the interpolant is then obtained from the data values at one (in triangular regions) or two (in trapezoidal regions) nearest data sites.

Another method for extending $\mathcal{T}(X)$ to cover S, which also reduces the incidence of elongated triangles, is to introduce imaginary data sites on the boundary of S. The impact this has on $\mathcal{T}(X)$ depends on the number and spacing of the imaginary sites. This approach is taken by McCullagh & Ross (1980). They only discuss the choice of the imaginary points in terms of the particular triangulation algorithm they use, so no general strategies can be inferred. In addition, they do not mention how data values should be assigned to the imaginary sites. One possibility is to extrapolate these values using the perpendicular extension, as above. However, this could result in the interpolated value at a site lying in a triangle with one or two vertices at imaginary

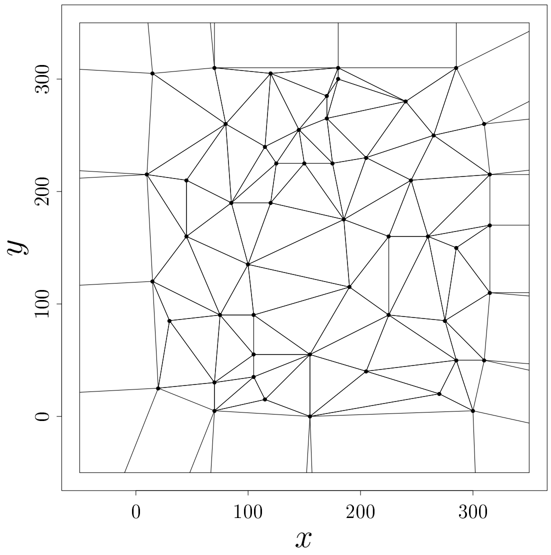


Figure 4.6: Extension of the Delaunay triangulation by inserting perpendiculars.

sites depending on more than three of the observed data values. Figure 4.7 illustrates this extension, again for the Delaunay triangulation of the `topo` data. Notice that the original triangulation has not been preserved, in that some of the edges in the original do not appear in the extension.

Many triangulations have been proposed in the literature, but they may be prone to lack of uniqueness and the creation of unnecessarily elongated triangles. Another problem is that there may be configurations of data sites for which no triangulation can be found. The Delaunay triangulation $\mathcal{D}(X)$ is usually proposed, because of its uniqueness and its direct construction. In addition, $\mathcal{D}(X)$ satisfies the max-min angle criterion and so, locally at least, is as equiangular as possible.

Having constructed $\mathcal{D}(X)$, the next task is to locate the site for interpolation $\boldsymbol{x}$ in the triangulation. Once that has been accomplished, the interpolated surface $\check{f}$ can be evaluated at $\boldsymbol{x}$.

Location

Akima (1978b) notes that locating a point in the triangulation is a time consuming procedure. The location subroutine he presents divides the x-y plane into nine rectangles and lists the triangles associated with each rectangle. This procedure helps to speed up the location. The subroutine extends $\mathcal{D}(X)$ by inserting perpendiculars, so

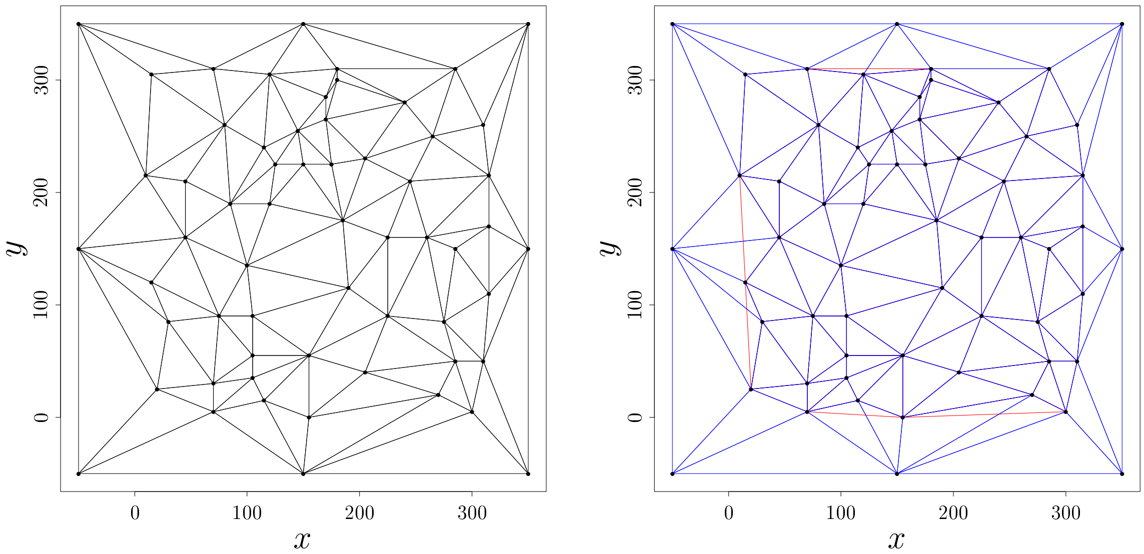


Figure 4.7: Extension of the Delaunay triangulation by inserting imaginary data sites on the boundary of S. The right hand plot shows the edges of the original triangulation (red) that do not appear in the extended version (blue).

that if the point lies outside the convex hull of the data it identifies the nearest border line segment when the point lies in a trapezoidal region and the nearest two border line segments when the point lies in an external triangular region.

Evaluation

Consider first the area inside the convex hull of the data (the true Delaunay triangulation), so that $\boldsymbol{x}$ lies in a triangle and $D(\boldsymbol{x}) = \{p_i, p_j, p_k\}$, say. The linear interpolant fits a plane over this triangle to interpolate the data. As in 4.3:

$$\check{z} = \check{f}(x, y) = w_i z_i + w_j z_j + w_k z_k \ ,$$

$$\text{where} \qquad w_i + w_j + w_k = 1 \ ,$$

and the w_i depend only on $\boldsymbol{x}_i$, $\boldsymbol{x}_j$, $\boldsymbol{x}_k$ and $\boldsymbol{x}$. To find these weights, note that the plane that is fitted over the triangle can be expressed as:

$$z = ax + by + c \ . \tag{4.5}$$

Since the points at the vertices of the triangle are known, a set of three linear equations in three unknowns can be derived. These can be expressed in matrix form as:

$$\begin{aligned} A(a,b,c)^{\mathrm{T}} &= (z_i, z_j, z_k)^{\mathrm{T}} \\ \Rightarrow \qquad (a,b,c)^{\mathrm{T}} &= A^{-1}(z_i, z_j, z_k)^{\mathrm{T}} \ , \end{aligned} \tag{4.6}$$

where

$$A = \begin{pmatrix} x_i & y_i & 1 \\ x_j & y_j & 1 \\ x_k & y_k & 1 \end{pmatrix} \quad \Rightarrow \quad A^{-1} = \frac{1}{\beta} \begin{pmatrix} y_j - y_k & y_k - y_i & y_i - y_j \\ -(x_j - x_k) & -(x_k - x_i) & -(x_i - x_j) \\ \alpha_{jk} & \alpha_{ki} & \alpha_{ij} \end{pmatrix} ,$$

with α_{ij}, α_{jk} and α_{ki} defined by:

$$\alpha_{pq} = x_p y_q - x_q y_p$$

and β by:

$$\beta = \alpha_{ij} + \alpha_{jk} + \alpha_{ki} \ .$$

If (4.6) is solved to find a, b and c, then the results can be put back into (4.5) and the equation rearranged to give the weights w_i, w_j and w_k, which are most conveniently expressed as:

$$w_i = \gamma_{jk} \ , \quad w_j = \gamma_{ki} \ , \quad w_k = \gamma_{ij} \ , \tag{4.7}$$

where γ_{ij}, γ_{jk} and γ_{ki} are defined by:

$$\gamma_{pq} = \big(x(y_p - y_q) - y(x_p - x_q) + \alpha_{pq}\big)/\beta \ . \tag{4.8}$$

A more geometrical derivation is as follows. Consider figure 4.8. First the point $\boldsymbol{x}'$ where the line through $\boldsymbol{x}_i$ and $\boldsymbol{x}$ meets the border of $\triangle \boldsymbol{x}_i \boldsymbol{x}_j \boldsymbol{x}_k$ opposite $\boldsymbol{x}_i$ must be found. The coordinates of this point are:

$$\begin{aligned} x' &= \big((x_j - x_k)(x_i y - x y_i) - \alpha_{jk}(x_i - x)\big)/\zeta \ , \\ y' &= \big((y_j - y_k)(x_i y - x y_i) - \alpha_{jk}(y_i - y)\big)/\zeta \ , \end{aligned}$$

where the denominator ζ is given by:

$$\zeta = \alpha_{ij} + \alpha_{ki} - x(y_j - y_k) + y(x_j - x_k) \ .$$

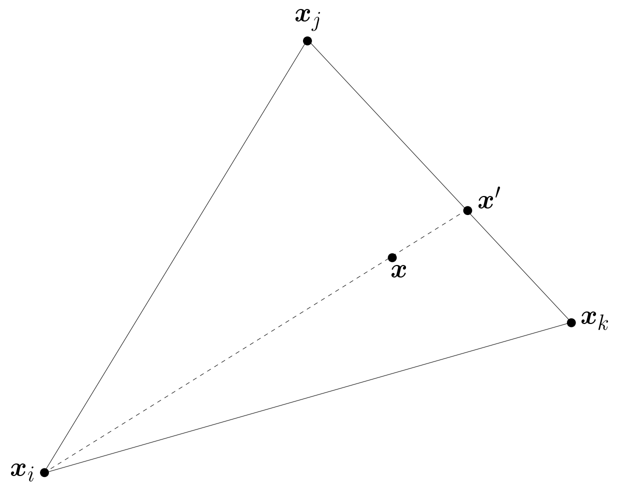


Figure 4.8: Illustration of linear interpolation over a triangle.

Interpolating over a line in $\mathbb{R}^2$ is simple, as it is similar to interpolation of 2D data, so the interpolated value $\check{z}'$ at $\boldsymbol{x}'$ can now be found easily as a linear combination of z_j and z_k:

$$\check{z}' = w'_j z_j + w'_k z_k \ ,$$

where w'_j and w'_k sum to one and

$$\begin{aligned} w'_j &= (x' - x_k)/(x_j - x_k) \\ &= \big(x(y_k - y_i) - y(x_k - x_i) + \alpha_{ki}\big)/\zeta \ . \end{aligned}$$

Now interpolate over the line joining $\boldsymbol{x}_i$ and $\boldsymbol{x}'$ to find the interpolated value $\check{z}$ at $\boldsymbol{x}$:

$$\begin{aligned} \check{z} &= w_i z_i + w' \check{z}' \\ &= w_i z_i + w' w'_j z_j + w' w'_k z_k \ , \end{aligned}$$

where w_i and w' sum to one and

$$\begin{aligned} w_i &= (x' - x)/(x' - x_i) \\ &= \big(x(y_j - y_k) - y(x_j - x_k) + \alpha_{jk}\big)/\beta \\ &= \gamma_{jk} \ , \end{aligned}$$

as in (4.8). Similarly $w' w'_j = w_j$ and $w' w'_k = w_k$ as presented in (4.7) and (4.8).

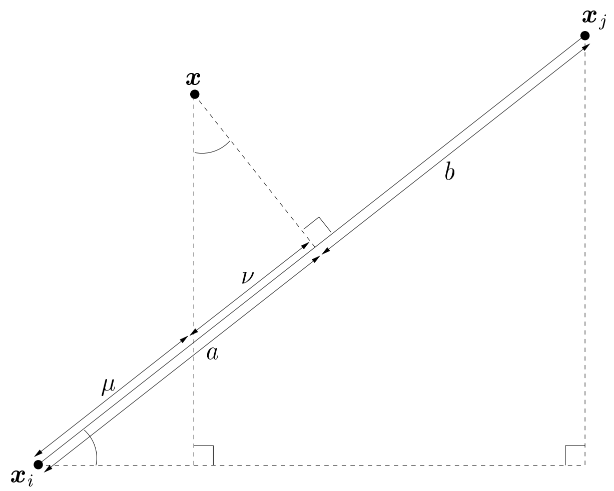


Figure 4.9: Illustration of linear interpolation over a trapezoidal region outside the convex hull of the data.

Now consider the area outside the convex hull in $\mathcal{T}(X)$ extended by inserting perpendiculars. In the "triangular" regions the interpolated value is just the data value of the point, p_i say, whose data site is the point where the "triangle" meets the convex hull. So $D(\boldsymbol{x})=\{p_i\}$, say, and

$$w_i = 1 \ . \tag{4.9}$$

The interpolated surface over this region is a plane with zero gradient.

In the "trapezoidal" regions the plane that ensures the whole surface is continuous is fitted. Thus points on lines perpendicular to the convex hull will take the same value - the interpolated value of the location on the convex hull where the perpendicular line meets the hull. Consider figure (4.9). In this case $D(\boldsymbol{x}) = \{p_i, p_j\}$, say, so that the interpolated value $\check{z}$ at $\boldsymbol{x}$ is given by:

$$\check{z} = w_i z_i + w_j z_j \ ,$$

where $w_i + w_j = 1$ and $w_j = a/(a+b)$. Since the three triangles are similar,

$$\mu/(a+b) = (x - x_i)/(x_j - x_i)$$

and $$\nu/(y_j - y_i) = \left((y - y_i) - (y_j - y_i)\frac{x - x_i}{x_j - x_i} \right) /(a+b) \ .$$

Rearrange to find $\nu/(a+b)$ and proceed:

$$\begin{aligned} w_j &= a/(a+b) \\ &= (\mu+\nu)/(a+b) \\ &= \frac{(x-x_i)(x_j-x_i)+(y-y_i)(y_j-y_i)}{(x_j-x_i)^2+(y_j-y_i)^2} \, . \end{aligned} \tag{4.10}$$

Note that a slightly different diagram is needed if $x < x_i$, but the same formula for w_j applies.

Computation

The `S-Plus` function `interp` performs linear interpolation based on the Delaunay triangulation, using the Fortran code from Akima (1978b). However, for two reasons this function is not sufficient for our needs. Firstly, for locations outside the convex hull of the data sites it returns only missing values. Secondly, as will be discussed in §4.3.1, the identities of the points in $D(\boldsymbol{x})$ and the values of the corresponding weights w_i are required for the extension of the Kovac-Silverman method, as well as the interpolated values, but `interp` returns only the interpolated values. Some new code has, therefore, been written. This consists of an `S-Plus` function `rarray`, which calls the `C` subroutine `RARRAY`. It is assumed that the data is to be interpolated onto a grid, so as well as the coordinates of the data points P the user must supply vectors containing the x and y coordinates of the grid points, e.g. the values of $\check{x}_i$ and $\check{y}_j$ in (4.1) and (4.2). The `C` subroutine first finds the Delaunay triangulation, using the Akima's `Fortran` subroutine `idtang`. This code is published in Akima (1978a) and is available from the Netlib archive. Then, for each grid point in turn, the new data site is located in the triangulation using Akima's subroutine `idlctn`. The weights w_i are then calculated, as in (4.7) & (4.8), (4.9) or (4.10) as appropriate. These values are stored in a three dimensional array `r`, where $\mathtt{r}_{ij1}$, $\mathtt{r}_{ij2}$ and $\mathtt{r}_{ij3}$ store the weights w_i, w_j and w_k and $\mathtt{r}_{ij4}$, $\mathtt{r}_{ij5}$ and $\mathtt{r}_{ij6}$ store the identities of the points in $D((\check{x}_i, \check{y}_j))$. Finally, the `C` code computes the interpolated values as in (4.3).

Example 4.5 *The linear interpolant for the* `topo` *data is shown in figure 4.10. The grid is the same as in example 4.4. Note that this grid extends a long way outside the convex hull of the data, so that a lot of extrapolation is done. In general, this is a*

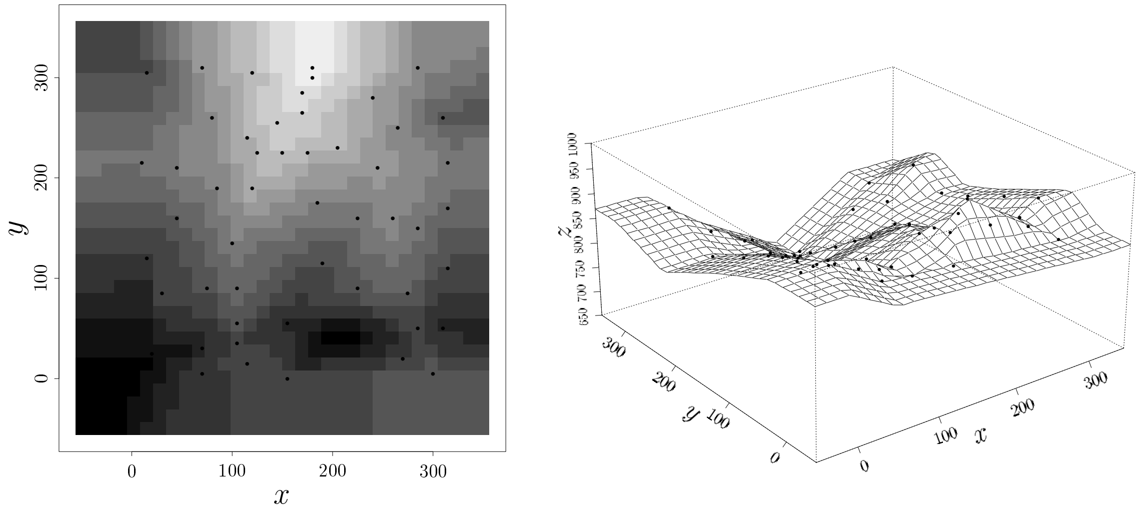


Figure 4.10: Grey scale plot (left) and perspective plot (right) of the linear interpolant based on the Delaunay triangulation of the `topo` dataset.

bad idea, but has been done here to illustrate again how the triangulation is extended beyond the convex hull.

4.3 Wavelet Smoothing

Recall that to use Mallat's efficient 2D DWT on irregularly spaced data it is necessary to first obtain data on an evenly spaced grid by, for example, interpolation. This interpolated data can most usefully be thought of as a matrix $\check{Z}$, with dimensions $2^J \times 2^J$, where $(\check{Z})_{ij} = \check{z}_{ij} = \check{f}(\check{x}_i, \check{y}_j)$. This can be decomposed using the 2D DWT to give wavelet coefficients which are hopefully close to the wavelet coefficients of the real data.

Example 4.6 *When interpolating data for use in wavelet methods, it is sensible to avoid much extrapolation. A suitable grid is shown in figure 4.11. The number of grid points in each direction is* 16. *This choice is somewhat arbitrary, but it does mean there are* 256 *interpolated values; many more than the number of observed points. In addition, the memory requirement for the programs increases a lot with the size of grid, so it is advisable to avoid making the grid larger than is necessary. On the other hand,*

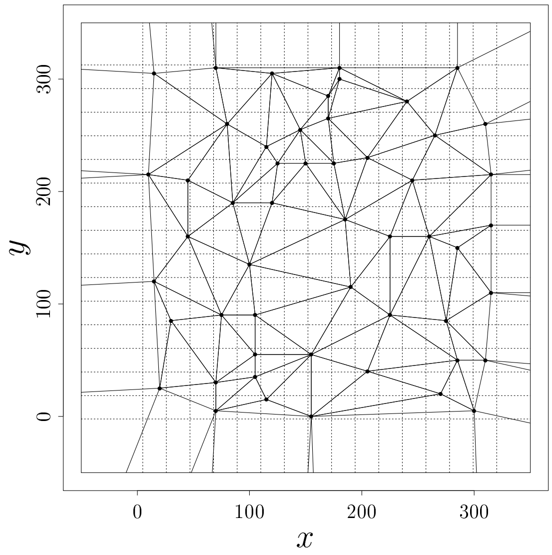


Figure 4.11: Extended Delaunay triangulation and grid for interpolation for the `topo` dataset.

the resulting interpolated data will yield only four levels of wavelet coefficients, and it is important that there are enough fine scale coefficients to represent the detail in the data. The x and y locations of the grid points are scaled and translated versions of (4.1), with $J=4$. The range of x values is $[0.1, 6.4]$ ($[5, 320]$ in feet) and the range of y values is $[-0.05, 6.25]$ ($[-2.5, 312.5]$ in feet), with the spacing between grid lines in both directions being 0.42 (21 feet).

Figure 4.12 shows a grey scale image and a perspective plot of the `topo` *dataset interpolated onto the above grid. The wavelet decomposition coefficients are displayed in figure 4.14.*

As in the 1D case, the interpolated data should not be directly thresholded using standard methods as the data are correlated and heteroscedastic. Following Kovac & Silverman (2000), coordinate wise thresholding is proposed, with thresholds proportional to the variance of each coefficient. Assume that the original data values $\boldsymbol{z}$ are uncorrelated and homoscedastic, so that:

$$\Sigma = \mathrm{Var}(\boldsymbol{z}) = \sigma^2 I \ .$$

The rest of this section will consider how the variances of the interpolated data can be computed and hence the variances of the wavelet coefficients.

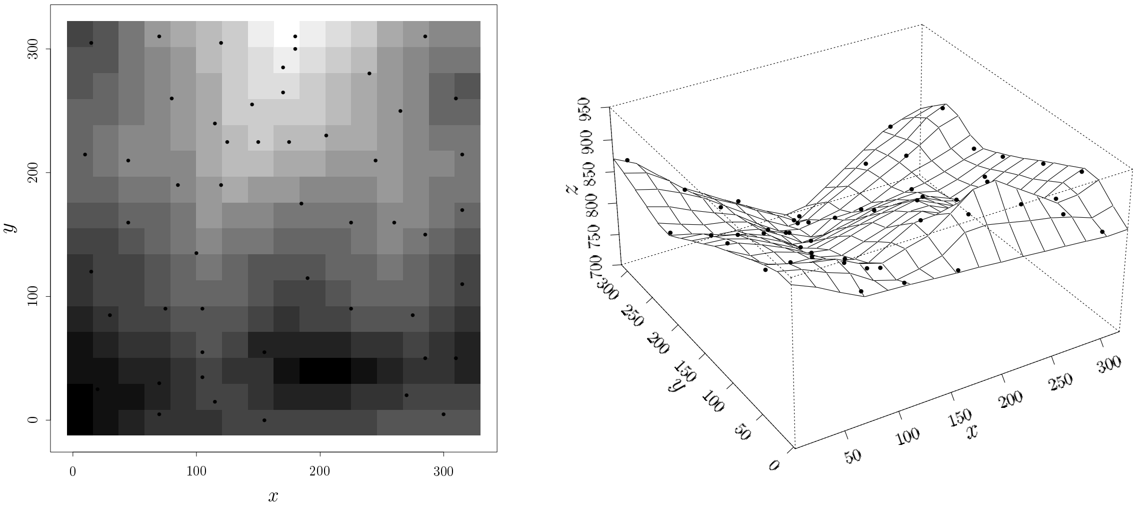


Figure 4.12: Grey scale plot (left) and perspective plot (right) of the interpolated `topo` dataset.

4.3.1 Variance of the Interpolated Data

If $\check{\Sigma} = \mathrm{Var}(\check{Z})$, then $\check{\Sigma}$ will be a 4D array, with dimensions $2^J \times 2^J \times 2^J \times 2^J$:

$$\check{\Sigma}_{ijkl} = \mathrm{Cov}(\check{z}_{ij}, \check{z}_{kl}) \ .$$

To investigate the structure of $\check{\Sigma}$, note that the linear interpolation can be written as a map involving a 3D array R, with dimensions $2^J \times 2^J \times n$:

$$\check{z}_{ij} = R_{ijk} z_k \ . \tag{4.11}$$

Note that R and $\check{\Sigma}$ can be thought of as tensors. Equation (4.11) and others to follow use the summation convention of tensor algebra: repeated subscripts in an expression are summed over all possible values. In (4.11) the subscript k is repeated on the right hand side, so $R_{ijk} z_k$ is summed over all possible values of k ($k = 1, \ldots, n$). $\check{\Sigma}$ can now be written as:

$$\begin{aligned}
\check{\Sigma}_{ijkl} &= \mathrm{Cov}(R_{ijn} z_n, R_{klm} z_m) \\
&= R_{ijn} R_{klm} \Sigma_{nm} \\
&= \sigma^2 R_{ijn} R_{kln} \\
&= \sigma^2 \Gamma_{ijkl} \ ,
\end{aligned}$$

say, where Γ is a 4-d array of *variance factors* which is determined by the interpolation scheme. The "rows" $R_{ij\cdot}$ of R contain the weights w_i in (4.3), with $w_i = 0$ for $p_i \in P \setminus D(\boldsymbol{x})$. Γ can be easily calculated from R.

Computation

The function `rarray`, discussed on p.107, calculates Γ, as well as R and the interpolated data.

The arrays R and Γ, as defined above, require a lot of storage space, even for moderate values of J, so it is important to consider how space can be saved. First consider R. Since linear interpolation is being used, there are at most three non zero entries in each "row" $R_{ij\cdot}$ of R, so only these entries and their locations are stored. In the notation of (4.3), the w_k and $\Big\{k : p_k \in D\big((\check{x}_i, \check{y}_j)\big)\Big\}$ are stored, so that the size of the stored object is $2^J \times 2^J \times 6$.

There are two ways in which storage for Γ can be reduced. Firstly, there is a symmetry in Γ:

$$\begin{aligned}
\Gamma_{ijkl} &= \frac{1}{\sigma^2}\,\mathrm{Cov}(\check{z}_{ij}, \check{z}_{kl}) \\
&= \frac{1}{\sigma^2}\,\mathrm{Cov}(\check{z}_{kl}, \check{z}_{ij}) \\
&= \Gamma_{klij}\ .
\end{aligned}$$

Clearly it is only necessary to store Γ_{ijkl} for $i \leqslant k$. However, a further restriction can be made when $i = k$ by only storing those elements with $j \leqslant l$, as $\Gamma_{ijil} = \Gamma_{ilij}$. These inequalities could, of course, be written the other way round, but this is the way the restrictions are presented in the programming. Table 4.1 shows in detail the ranges of i, j, k, l that need to be saved, for Γ based on an $n_r \times n_c$ image. This is useful in the programming, but also in the calculation of the values in Table 4.2, which shows a breakdown of the number of elements of Γ that need to be saved. Note that the total agrees with the more obvious calculation that all the "diagonal" elements (Γ_{ijij} – $n_r n_c$ of them) and half the "non-diagonal" elements ($(n_r^2 n_c^2 - n_r n_c)$ of them) need to be saved. However, it is worth bearing the long winded approach in mind, as a model for when things get more complicated. This saves considerably in storage space, although care must be taken to make sure that the elements are correctly stored and accessed.

Table 4.1: Interpreting restrictions on the indices of Γ based on an $n_r \times n_c$ data matrix in terms of the ranges of values they take. This is useful for calculating the contributions in Table 4.2 and for programming.

Restrictions on i and k	Range of i	Range of k
$i = k$	$1 \leqslant i \leqslant n_r$	–
$i < k$	$1 \leqslant i < k$	$1 < k \leqslant n_r$
Restrictions on j and l	Range of j	Range of l
$j = l$	$1 \leqslant j \leqslant n_c$	–
$j < l$	$1 \leqslant j < l$	$1 < l \leqslant n_c$
$j > l$	$1 < j \leqslant n_c$	$1 \leqslant l < j$

Table 4.2: Breakdown of the calculation of the number of distinct elements of Γ that need to be stored when symmetry is taken into consideration, for Γ based on an $n_r \times n_c$ data matrix.

Restrictions on indices	Number of entries in Γ
$i = k$ and $j = l$	$n_r n_c$
$i = k$ and $j < l$	$\frac{1}{2} n_r n_c (n_c - 1)$
$i < k$ and $j = l$	$\frac{1}{2} n_r n_c (n_r - 1)$
$i < k$ and $j < l$	$\frac{1}{4} n_r n_c (n_r - 1)(n_c - 1)$
$i < k$ and $j > l$	$\frac{1}{4} n_r n_c (n_r - 1)(n_c - 1)$
TOTAL	$\frac{1}{2} n_r n_c (n_r n_c + 1)$

`S-Plus` functions are available to access the stored version of Γ and also to decode it into a full 4D array, should this be required.

The second way in which storage can be reduced is if some sort of band structure can be found in Γ. This is not as easy to determine as in the 1D case. Recall that $R_{ij.}$ is a vector of length n with three (or one/two if $(\check{x}_i, \check{y}_j)$ lies outside the convex hull of the data) non zero entries. However, since there is no obvious ordering of the data z_k, the non zero entries will not necessarily be consecutive, and so it is not possible to assume anything about the relative positions of the non-zero values for different i and j. Since R has no particular band structure, R can not reveal anything about the

Table 4.3: Interpreting restrictions on the indices of Γ based on an $n_r \times n_c$ data matrix in terms of the ranges of values they take, when bandwidths are used. This is useful for calculating the contributions in Table 4.4 and for programming.

Restrictions on i and k	Range of i	Range of k
$i = k$	$1 \leqslant i \leqslant n_r$	–
$i < k$	$\max(1, k - b_1) \leqslant i < k$	$1 < k \leqslant n_r$
Restrictions on j and l	Range of j	Range of l
$j = l$	$1 \leqslant j \leqslant n_c$	–
$j < l$	$\max(1, l - b_2) \leqslant j < l$	$1 < l \leqslant n_c$
$j > l$	$1 < j \leqslant n_c$	$\max(1, j - b_2) \leqslant l < j$

Table 4.4: Breakdown of the calculation of the number of distinct, non-zero elements of Γ that need to be stored when symmetry and bandwidths are taken into consideration, for Γ based on an $n_r \times n_c$ data matrix.

Restrictions on indices	Number of entries in Γ
$i = k$ and $j = l$	$n_r n_c$
$i = k$ and $j < l$	$\frac{1}{2} n_r b_2 (2n_c - b_2 - 1)$
$i < k$ and $j = l$	$\frac{1}{2} n_c b_1 (2n_r - b_1 - 1)$
$i < k$ and $j < l$	$\frac{1}{4} b_1 b_2 (2n_r - b_1 - 1)(2n_c - b_2 - 1)$
$i < k$ and $j > l$	$\frac{1}{4} b_1 b_2 (2n_r - b_1 - 1)(2n_c - b_2 - 1)$
TOTAL	$\frac{1}{2}\Big\{ n_r n_c + \big(n_r + b_1(2n_r - b_1 - 1)\big) * \big(n_c + b_2(2n_c - b_2 - 1)\big)\Big\}$

structure of Γ. Heuristically, one might think that, because of the local nature of the linear interpolation scheme, two grid points that are far apart will not depend on the same data values, and so will be uncorrelated. It would then be possible to discover some sort of band structure. However, the presence of elongated triangles will reduce any effect as they can induce correlations between well separated grid points.

Recall that a bandwidth for a matrix A is the smallest integer b such that $A_{ij} = 0$ for all i, j such that $|i - j| > b$. One approach with a 4D array B is to define two bandwidths b_1, b_2 to be the smallest integers such that $B_{ijkl} = 0$ for all i, j, k, l such that

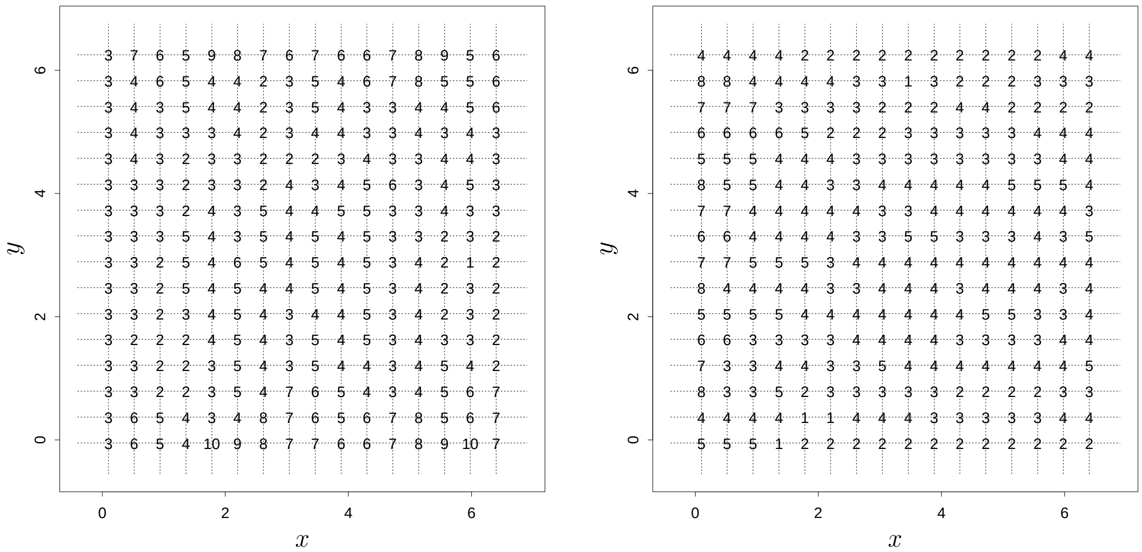


Figure 4.13: Correlation distances in the x direction (left) and the y direction (right) for the interpolated `topo` dataset.

$|i-k| > b_1$ or $|j-l| > b_2$. In terms of the 4D covariance factor array Γ, this says that $\Gamma_{ijkl} \propto \mathrm{Cov}(\check{z}_{ij}, \check{z}_{kl}) = 0$ if $|i-k| > b_1$ or $|j-l| > b_2$. Tables 4.3 and 4.4 show the same as tables 4.1 and 4.2, but also taking into account the fact that elements outside of the bandwidths need not be saved.

Example 4.7 *Figure 4.13 shows, for the* `topo` *dataset the maximum lag with non-zero correlation for each grid point, in both the x and y directions. For example, the top left grid point $(\check{x}_0, \check{y}_{15})$ is correlated with points up to three grid points away in the x direction, e.g. $(\check{x}_3, \check{y}_{15})$, and up to four grid points away in the y direction e.g. $(\check{x}_0, \check{y}_{11})$. Note that, for example, $(\check{x}_0, \check{y}_0)$ is not correlated with $(\check{x}_3, \check{y}_0)$ but is correlated with $(\check{x}_3, \check{y}_1)$ so its correlation distance in the x direction is* 3. *The bandwidths can now be easily identified, since b_1 is the largest value in the left hand plot and b_2 is the largest value in the right hand plot. So for this data and grid:*

$$b_1 = 10 \qquad b_2 = 8$$

Note that the larger values tend to cluster along the top and bottom edges of the grid in the left hand plot, and on the left and right hand edges in the right hand plot - i.e. when correlations parallel to the edges are considered for grid points close to the edges. This

Table 4.5: The effect of using symmetry and bandwidths to reduce the storage space for Γ based on a 16×16 grid for the `topo` dataset.

Factor	Size	% of whole
–	65536	100
symmetry	32896	50.2
symmetry and bandwidths $(10, 8)$	22728	34.7

is because elongated triangles reduce the local properties of the interpolant and such triangles tend to be found near the edge of the triangulation. It may possible to reduce such boundary effects by extending the Delaunay triangulation by inserting imaginary data points, as in figure 4.7.

To see the effect of using symmetry and bandwidths, look at table 4.5. Note that when taking symmetry into account only 4092 *(6.2%) of the* 32896 *distinct elements of* Γ *are in fact non-zero. So, in this example at least, there is considerable potential for savings in terms of the storage space required for* Γ.

At present only symmetry is used in the software; bandwidths have not yet been programmed. One important point to consider is the identification of the bandwidths before calculating Γ. If this can be done then clearly much time will be saved by not calculating zeros, as well as saving storage space. In the example the bandwidths were calculated directly from Γ, but it is possible to get this information from knowledge of the triangulation and the grid. Although a complete search of all possible correlations would be very time consuming, it is possible to do something quicker. Consider a general point in the grid $(\check{x}_i, \check{y}_j)$. To find the maximum x lag, λ^x_{ij} consider the point $(\check{x}_k, \check{y}_j)$ and, starting from $i+1$ increase k until $D(\check{x}_i, \check{y}_j)$ and $D(\check{x}_k, \check{y}_j)$ contain no points in common. Then for this value k' of k check to see if the same holds for all points $(\check{x}_{k'}, \check{y}_l)$. If this holds then set $k^+_{ij} = k'$. If not, then choose $l' \neq j$ such that $D(\check{x}_i, \check{y}_j)$ and $D(\check{x}_{k'}, \check{y}_{l'})$ have at least one point in common and continue in a similar fashion by increasing k beyond k'. Once k^+_{ij} has been found then k^-_{ij} can similarly be found by considering again $(\check{x}_k, \check{y}_j)$ but decreasing k, starting from $i-1$. Then:

$$\lambda^x_{ij} = \max(k^+_{ij} - i, i - k^-_{ij}) \ .$$

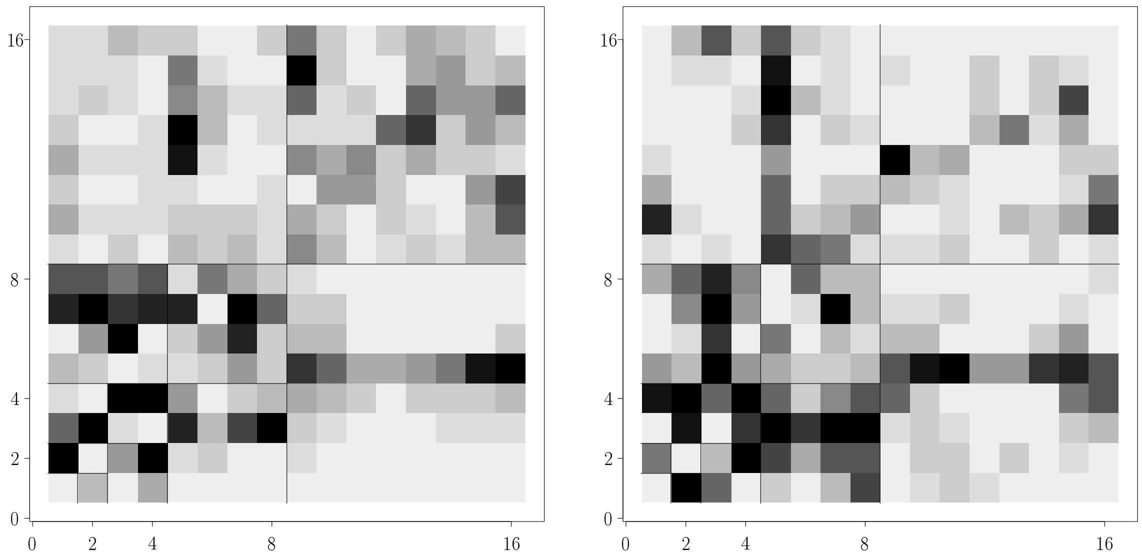


Figure 4.14: Wavelet coefficients (left) and variance factors of the wavelet coefficients (right) of the interpolated `topo` dataset.

Note that once λ^x_{ij} has been found, λ^x_{kl} for points lying in the same triangle follows easily. For points $(\check{x}_i, \check{y}_l)$ the lag will be the same and for points $(\check{x}_{i+1}, \check{y}_j)$, $k^+_{i+1,j} = k^+_{ij} - 1$ and $k^-_{i+1,j} = k^-_{ij} + 1$.

λ^y_{ij} can be found in a similar fashion. Note that once R has been calculated, $D(\check{x}_i, \check{y}_j)$ is known $\forall i, j$ so if this procedure is used after calculating R and before calculating Γ it should be quite effective.

4.3.2 Variance Factors of Wavelet Coefficients

It is now necessary to decompose Γ to find the variance factors of the wavelet coefficients. The formulae for the decomposition of 2D data, which will be applied to the interpolated data, are given in equations (2.43) – (2.46). Following Kovac & Silverman (2000), the correlations in $\check{Z}$ will be ignored and correction for heteroscedasticity concentrated on. Thus at each level it is necessary to compute the full covariance structure of the scaling function coefficients (denoted by Γ^j), but only the variances of the wavelet coefficients (denoted by $\Gamma^{i,j}$, for $i = 1, 2, 3$). The decompositions are defined as follows:

$$\begin{aligned}
\Gamma^{j-1}_{k_1 l_1 k_2 l_2} &= \mathrm{Cov}(c^{j-1}_{k_1 l_1}, c^{j-1}_{k_2 l_2})/\sigma^2 \\
&= h_{m_1-2k_1} h_{m_2-2k_2} h_{n_1-2l_1} h_{n_2-2l_2} \Gamma^{j}_{m_1 n_1 m_2 n_2} \ , \\
\Gamma^{1,j-1}_{kl} &= \mathrm{Var}(d^{1,j-1}_{kl})/\sigma^2 \\
&= g_{m_1-2k} g_{m_2-2k} h_{n_1-2l} h_{n_2-2l} \Gamma^{j}_{m_1 n_1 m_2 n_2} \ , \\
\Gamma^{2,j-1}_{kl} &= \mathrm{Var}(d^{2,j-1}_{kl})/\sigma^2 \\
&= h_{m_1-2k} h_{m_2-2k} g_{n_1-2l} g_{n_2-2l} \Gamma^{j}_{m_1 n_1 m_2 n_2} \ , \\
\Gamma^{3,j-1}_{kl} &= \mathrm{Var}(d^{3,j-1}_{kl})/\sigma^2 \\
&= g_{m_1-2k} g_{m_2-2k} g_{n_1-2l} g_{n_2-2l} \Gamma^{j}_{m_1 n_1 m_2 n_2} \ .
\end{aligned}$$

Example 4.8 *The variance factors for the* `topo` *dataset are shown in Figure 4.14 together with the coefficients themselves. Note that the coefficients are scaled by level and that the absolute values are displayed (obviously this won't effect the variance plot.)*

Thresholding

Having calculated the variance factors for all the wavelet coefficients, the coefficients can now be thresholded. As Kovac and Silverman do in the 1D case, coordinate wise thresholds are used, of the form:

$$\tau^{ij}_{kl} = \upsilon \Gamma^{ij}_{kl} \quad \text{for } i = 1, 2, 3 \ ,$$

for some *threshold constant* υ, where j is the resolution level. The simplest method of selecting a suitable value of υ is to use the adaptation of the universal threshold as in (2.40):

$$\upsilon^u = \sqrt{2 \log n} \ \sigma \ , \tag{4.12}$$

where σ is estimated from the data using a 2D analogue of (2.41).

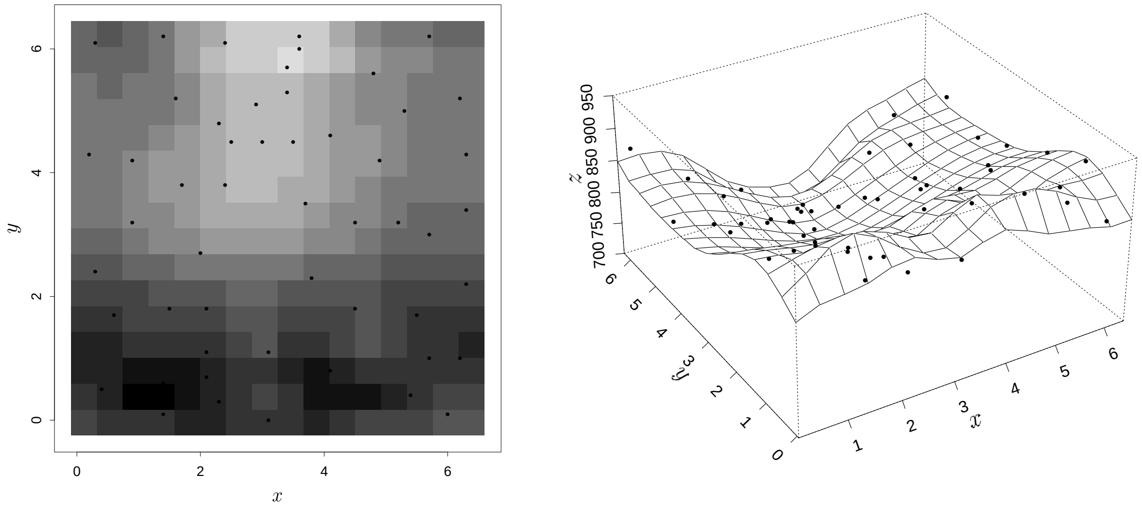


Figure 4.15: Grey scale plot (left) and perspective plot (right) of the smoothed `topo` dataset, using the universal threshold.

Example 4.9 *The universal threshold constant chosen for the* `topo` *data is* 67.09*, to* 2 *d.p.. The smoothed surface resulting from thresholding using this value is shown in figure 4.15.*

4.4 Threshold Selection via Cross Validation

Threshold selection by full leave-one-out cross validation is not possible for standard wavelet methods, because omitting one data point would violate the regular design. However, since a method is now available that can be used on any configuration of data points, such a cross validation approach is possible.

This full cross validation involves omitting each point (p_i) in turn and calculating the wavelet shrinkage estimate $\hat{f}_{\upsilon,-i}$ of f from the remaining points using a threshold constant υ. The predicted value at (x_i, y_i) is then compared with the data value z_i and the cross validation sum of squares is given by:

$$\hat{M}(\upsilon) = \sum_{i=1}^{n} (\hat{f}_{\upsilon,-i}(x_i, y_i) - z_i)^2$$

Note that since $\hat{f}_{\upsilon,-i}$ is evaluated on a grid $\widehat{f_{\upsilon,-i}}(x_i, y_i)$ is found by linear interpolation.

The squares of the grid are divided into triangles in a consistent way, and planes then fitted over these triangles. The threshold constant is chosen to minimise $\hat{M}(v)$.

Once the cross validation was implemented, problems became apparent. It was noticed that the answer depended on the starting value (initial threshold constant) given to the minimisation routine. Figure 4.16 shows the cross validation sum of squares plotted as a function of the threshold constant. The different lines correspond to a different number of coarse resolution levels being protected against thresholding. It is immediately clear that the reason for the answer depending on the starting value is that the function being minimised is far from smooth. There are several local minima to be found. Also, for the top two lines, the minimum is actually at zero. This corresponds to doing no thresholding at all. However, this does not mean that nothing was done. The data are first interpolated to a grid and then the value at the omitted location is found by interpolating between the values on the grid, so a significant amount of smoothing is actually done.

Another explanation for this is perhaps to do with the grid size. Looking at Figure 4.11 it can be seen that many triangles cover only one or two grid points, so that the gridded data do not really reflect the roughness of the interpolated surface. Perhaps more grid points need to be used so that it becomes clear that thresholded versions really are smoother and better fitting. By taking relatively few grid points, more smoothing is being done in the interpolation stage than would be the case if more grid points were taken.

However, there is a noticeable dip in all four cases for a constant between 70 and 80. This was investigated further by zooming in to finer scales. The results are shown in Figure 4.17. The minimum value in this region is achieved with a threshold constant of 77.6. The plots for the surface resulting from this choice are shown in Figure 4.18.

Nason (1994) investigates the form of the cross-validation sum of squares for the twofold cross validation discussed in §2.3. A heuristic justification is provided of the near convexity of the cross validation score (2.38), with a negative first derivative at $\tau = 0$. Unfortunately it is not possible to easily extend these arguments, because the interpolation stage complicates things considerably. The relationship between the wavelet coefficients and the data values depends on the data sites and this is what complicates things, since, for example, Parseval's relation can no longer be applied.

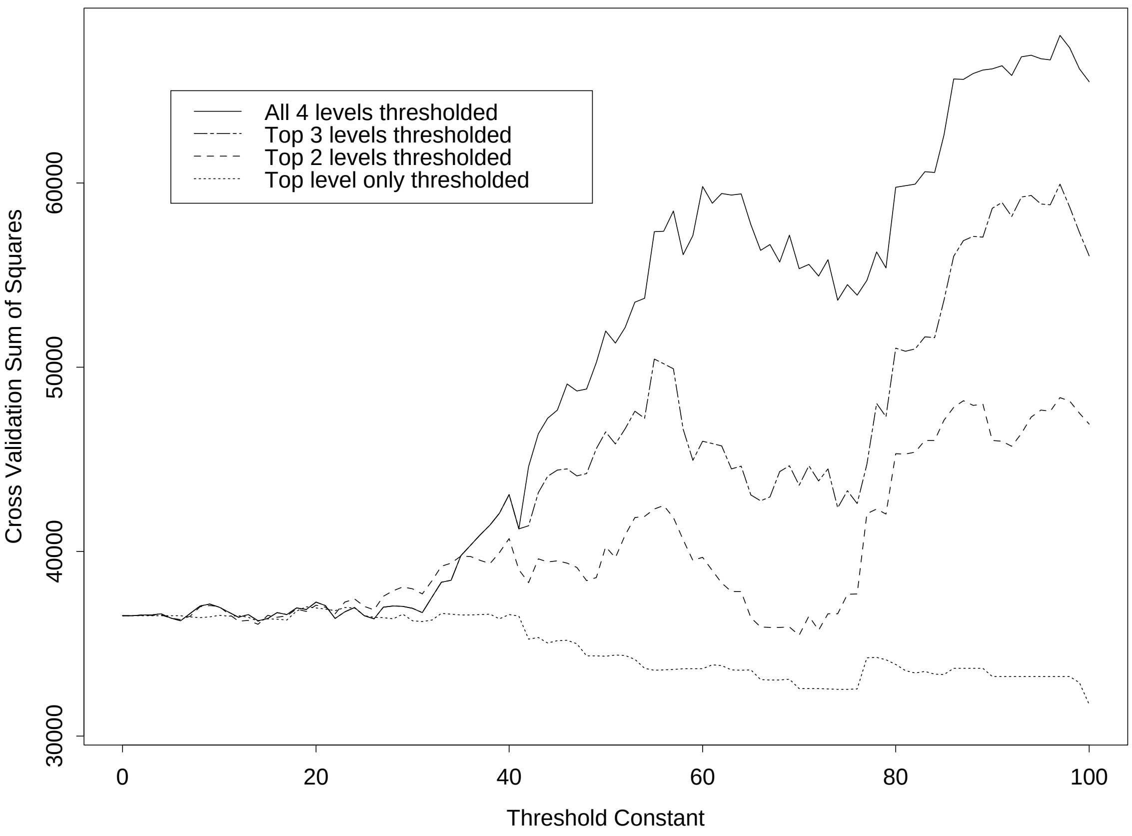


Figure 4.16: Cross validation sum of squares for the `topo` dataset. The different lines correspond to a different number of levels being thresholded.

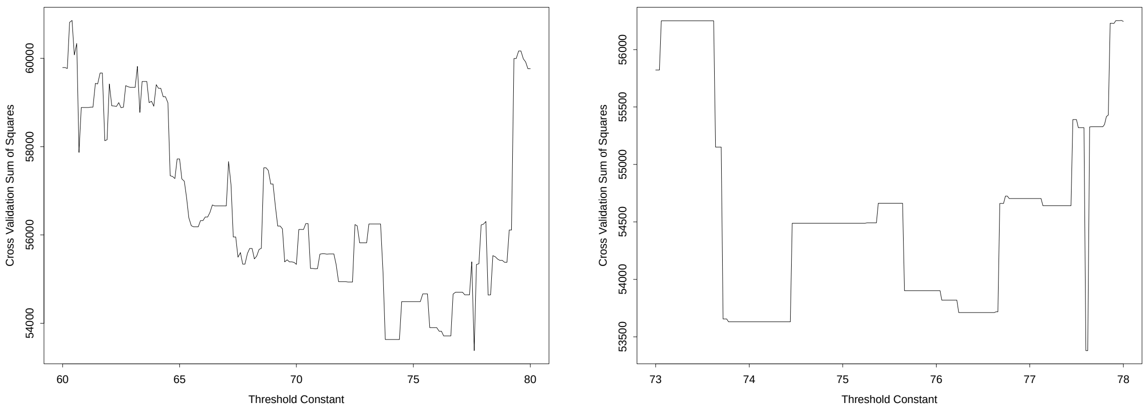


Figure 4.17: Higher resolution evaluation of the cross validation sum of squares in figure 4.16 for the case with all levels being thresholded. Note that the right hand plot is a close-up of the left hand plot.

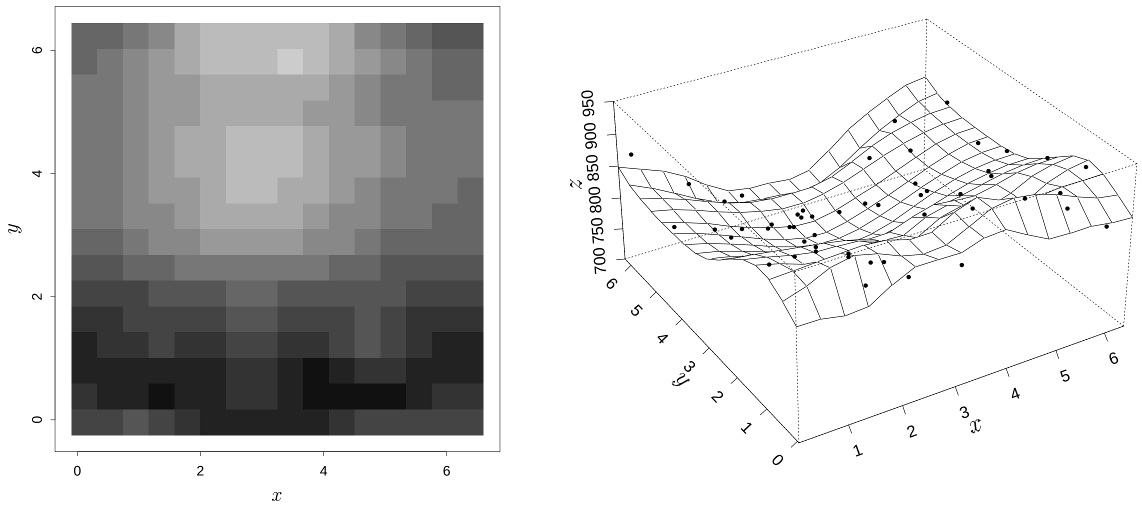


Figure 4.18: Grey scale plot (left) and perspective plot (right) of the smoothed `topo` dataset, with a threshold selected using cross validation.

4.5 Conclusions and Further Work

The Kovac-Silverman method for extending wavelet shrinkage in one dimension to handle unequally spaced data has been successfully extended to two dimensional problems. Although many steps have been taken in the implementation of this methodology to address questions of efficiency, more could be done, especially in the storage of the covariance array Γ. Although the possibility of defining a bandwidth for Γ has been discussed, none of the ideas have been implemented yet. It would also be worth investigating the effect of truncating correlations beyond a certain lag in order to obtain a tighter bandwidth. If the correlations at large lags are generally small, this should not have too much effect on the final computed variances of the wavelet coefficients.

Having said that, the 2D version of the Kovac-Silverman is inevitably computationally heavy, so extensions of this method to higher dimensional problems may not be computationally feasible. It is possible that a different approach, such as lifting, may prove to be much more efficient. The methodology developed here will be evaluated by means of a simulation study in chapter 5.

The leave-one-out cross validation approach has not been successfully implemented. However, such an approach should be feasible, but it would need to take account of the interpolation stage of the smoothing algorithm in some way.

Chapter 5

Evaluation of the Surface Estimation Procedure

In this chapter the methodology developed in Chapter 4, for using wavelet methods for surface estimation from irregularly spaced data, will be evaluated in a simulation study. The performance of the wavelet methodology will be compared to that of kriging. The results of the study are summarised in tables 5.11 – 5.17. Summaries of these tables, which are useful aids in the interpretation of the results, can be found in tables 5.3, 5.4 and 5.5.

The main finding in this chapter is that the wavelet methodology performs better than kriging for small sample sizes and high noise levels. In addition, a simple adaptation to the wavelet algorithm allows a known feature to be included in the model with minimal loss of performance.

The surface from which the data will be simulated will be discussed first, in §5.1. §5.2 will then outline the methods which will be used to find surface estimates from the simulated data, and the way in which the data will be simulated will be detailed in §5.3. Finally, the results will be discussed in §5.4.

5.1 The True Surface

It was planned to use real data as an example of the new methodology in action. This data was to have been provided by the British Antarctic Survey, and would have consisted of measurements of the density of krill in the ocean. Some sets of these

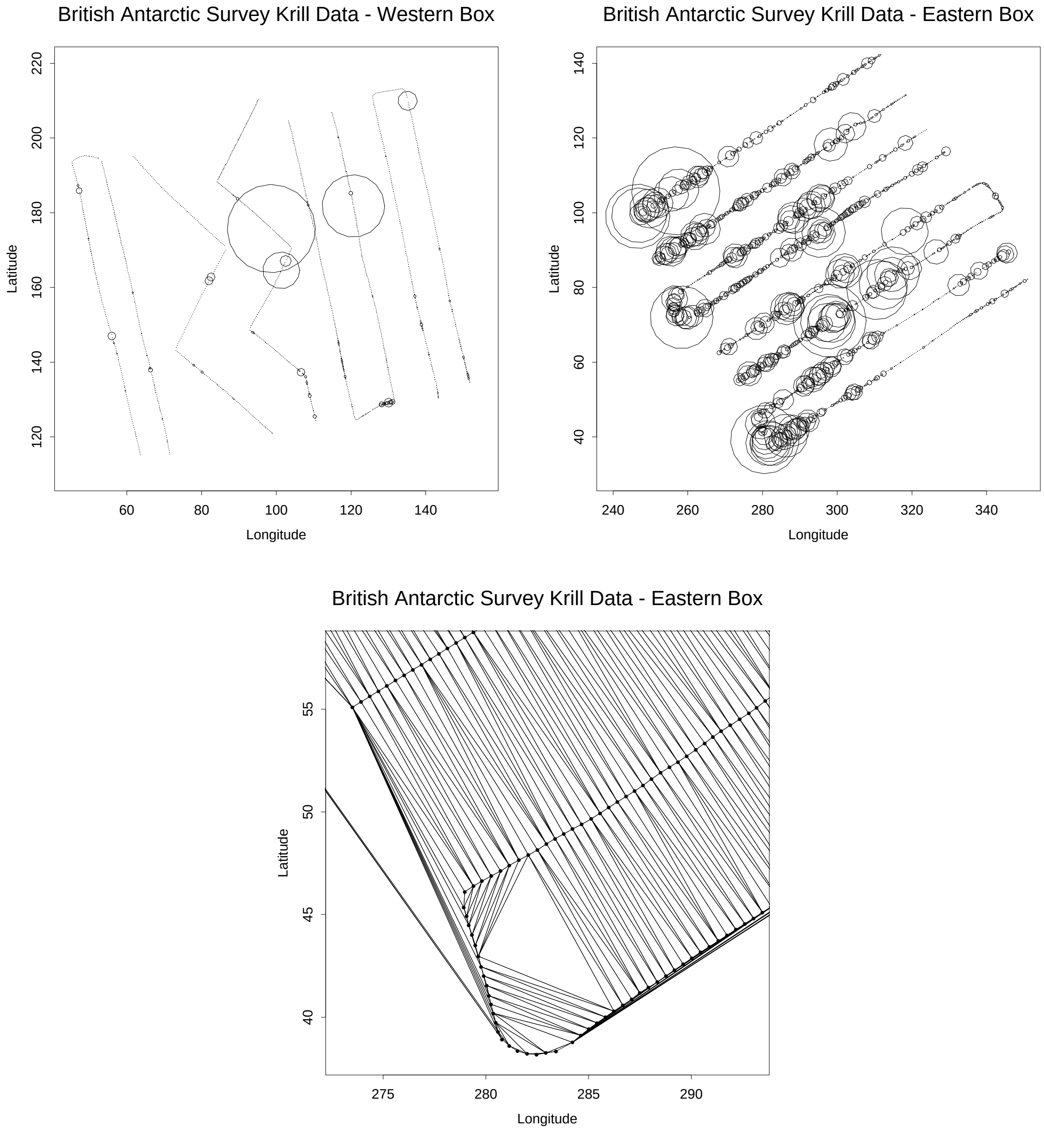


Figure 5.1: Two sets of observations (top row), collected by the British Antarctic Survey, of krill density in the Antarctic ocean. The radius of the circle centred at the location of each observation is proportional to the recorded density there. The largest recorded density in the western box data is 7726 and in the eastern box data 638 (to the nearest integer). The bottom row shows a close-up of the bottom corner of the Delaunay triangulation of the eastern box krill data.

measurements are available, and are displayed in figure 5.1. These data were collected by J.L.Watkins, A.S.Brierly, C.Goss and A.W.A.Murray. These measurements were taken from a ship, and the particular path taken by the ship can be clearly seen. Notice that the western box data are dominated by a couple of very large values. The British Antarctic Survey team have commented that these are probably errors. Although these observations are not taken on a regular grid, they are clearly not taken at random and will be unsuitable for use with the wavelet methodology. This is because the interpolation stage will produce a triangulation with numerous long thin triangles (see the bottom row of figure 5.1). Although measurements taken at random locations have been promised, such data are, unfortunately, not yet available for analysis.

Instead, simulated data will be used. The model will be motivated by some of the features that the real data were expected to exhibit. The most striking feature was that there was an island in the middle of the data collection area. This means that there were no measurements collected in this zone and that the density of krill is known to be zero there. To model this situation, a smooth, polynomial, surface over the unit square will be used, and then an island will be "cut" into it. A quartic surface in the unit square was chosen, with the parameters selected to ensure that the surface is periodic ($f(x,0)=f(x,1)$, $f(0,y)=f(1,y)$). Periodicity is desirable as it will help avoid artificially large wavelet coefficients near the boundaries of the unit square (if periodic boundary conditions are used in the DWT). If the quartic surface is written as:

$$\begin{aligned} f(x,y) = a_1 + b_1x + b_2y + c_1x^2 + c_2xy + c_3y^2 + d_1x^3 + d_2x^2y + d_3xy^2 + d_4y^3 \\ +e_1x^4 + e_2x^3y + e_3x^2y^2 + e_4xy^3 + e_5y^4 \ , \end{aligned}$$

then to ensure periodicity, the following restrictions on the parameters should be applied:

$$\begin{gathered} d_2 = -c_2 \ , \qquad d_3 = -c_2 \ , \qquad e_2 = 0 \ , \qquad e_3 = c_2 \ , \qquad e_4 = 0 \ , \\ e_1 = -(b_1 + c_1 + d_1) \ , \qquad e_5 = -(b_2 + c_3 + d_4) \ . \end{gathered}$$

In fact the coefficients were chosen as:

$$\begin{gathered} a_1 = 1 \ , \quad b_1 = 0 \ , \quad b_2 = 0 \ , \quad c_1 = 1 \ , \quad c_2 = -1 \ , \quad c_3 = 0.5 \ , \\ d_1 = 2 \ , \quad d_2 = 1 \ , \quad d_3 = 1 \ , \quad d_4 = 1 \ , \\ e_1 = -3 \ , \quad e_2 = 0 \ , \quad e_3 = -1 \ , \quad e_4 = 0 \ , \quad e_5 = -1.5 \ . \end{gathered}$$

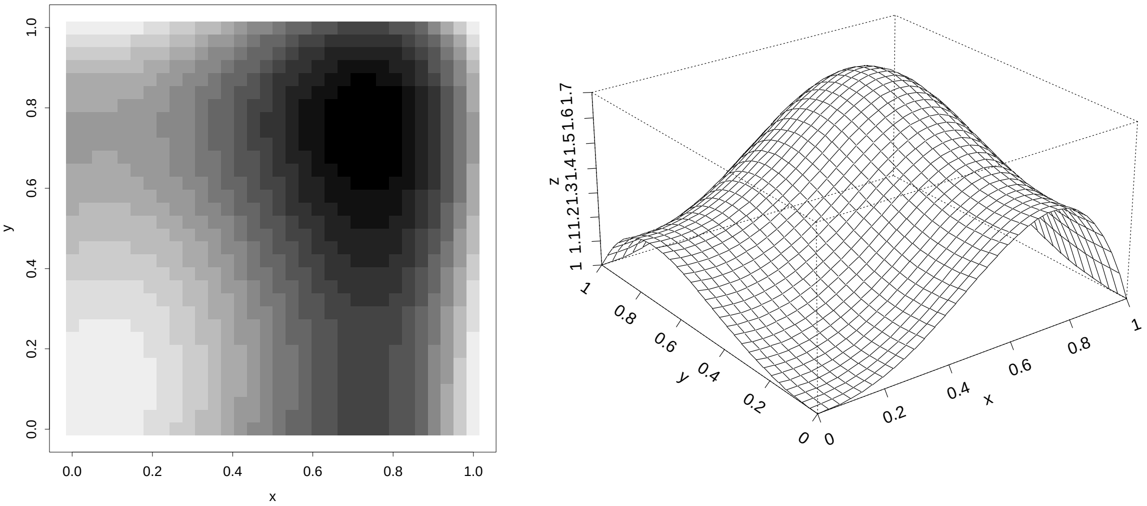


Figure 5.2: Grey scale plot (left) and perspective plot (right) of the smooth surface model for the simulation study.

The chosen surface is displayed in figure 5.2.

The parameters were chosen so that the surface had a peak in the unit square and that the position of this peak left enough space for the island to be placed in front of it. The motivation for this was that it seems reasonable to assume that the tidal flow around the island will affect the distribution of krill and result in a higher concentration on one side of the island.

Three islands will be considered. The first will be a circle of radius 0.15 centred at $(0.3, 0.3)$. The other two islands will be square, both with sides of length 0.3, centred at $(0.3, 0.3)$ and rotated by an angle θ about $(0.3, 0.3)$. One will be aligned with the x and y axes (i.e. $\theta = 0$), the other will be rotated anti-clockwise by $\frac{\pi}{8}$ about its centre $(\theta = -\frac{\pi}{8})$. The surfaces containing these three islands are illustrated in figure 5.3. Note that these surfaces, as with those in figure 5.2, have only been evaluated over a 32×32 grid, which accounts for the jagged appearance of the grey scale plots. This is the resolution that will be used in the simulation study, but for illustrative purposes finer resolution grey scale plots can be seen in figure 5.4. For these plots the surfaces are evaluated over a 256×256 grid.

The reason for the rotation of the square island is that the 2D wavelet transform coefficients represent detail in the horizontal, vertical and diagonal directions (see §2.5.2).

It will, therefore, be interesting to see if the performance of the wavelet smoothing is significantly reduced when the sides of the square (edges of the discontinuity) are not aligned with any of these directions. In principal the wavelet methodology could be adapted so that, at the interpolation stage, the orientation of the grid relative to the axes could be chosen as another parameter. This would mean that any important features could be aligned as closely as possible with the horizontal, vertical or diagonal and therefore be most efficiently estimated. It would be possible, if time consuming, for this parameter to be chosen in conjunction with the thresholding parameters by some cross validation routine.

The methods of surface estimation that will be used in the simulation study will now be discussed.

5.2 Estimation Methods

The simulated noisy data will be smoothed using the wavelet methodology for irregularly spaced 2D data. The standard method, as introduced in chapter 4, will also be adapted for the specific situation described above, of the presence of a known feature in the data collection area. To provide a comparison with an established technique of spatial smoothing, kriging will also be used.

5.2.1 Kriging

Kriging is discussed in some detail in Chapter 4 of Ripley (1981). It explicitly takes into consideration the correlation between values of the surface at short distances. If the covariance function is unknown, then it is estimated as part of the kriging algorithm, although it is necessary to specify a model for it. Since in the simulation study the covariance function will be unknown, the description below assumes this to be the case.

Kriging involves splitting the surface model into trend and fluctuation components:

$$f(x, y) = m(x, y) + \varepsilon(x, y) \ .$$

The trend m is generally regarded as fixed large-scale variation and the fluctuation ε as a small-scale random process. These two components are predicted separately, using

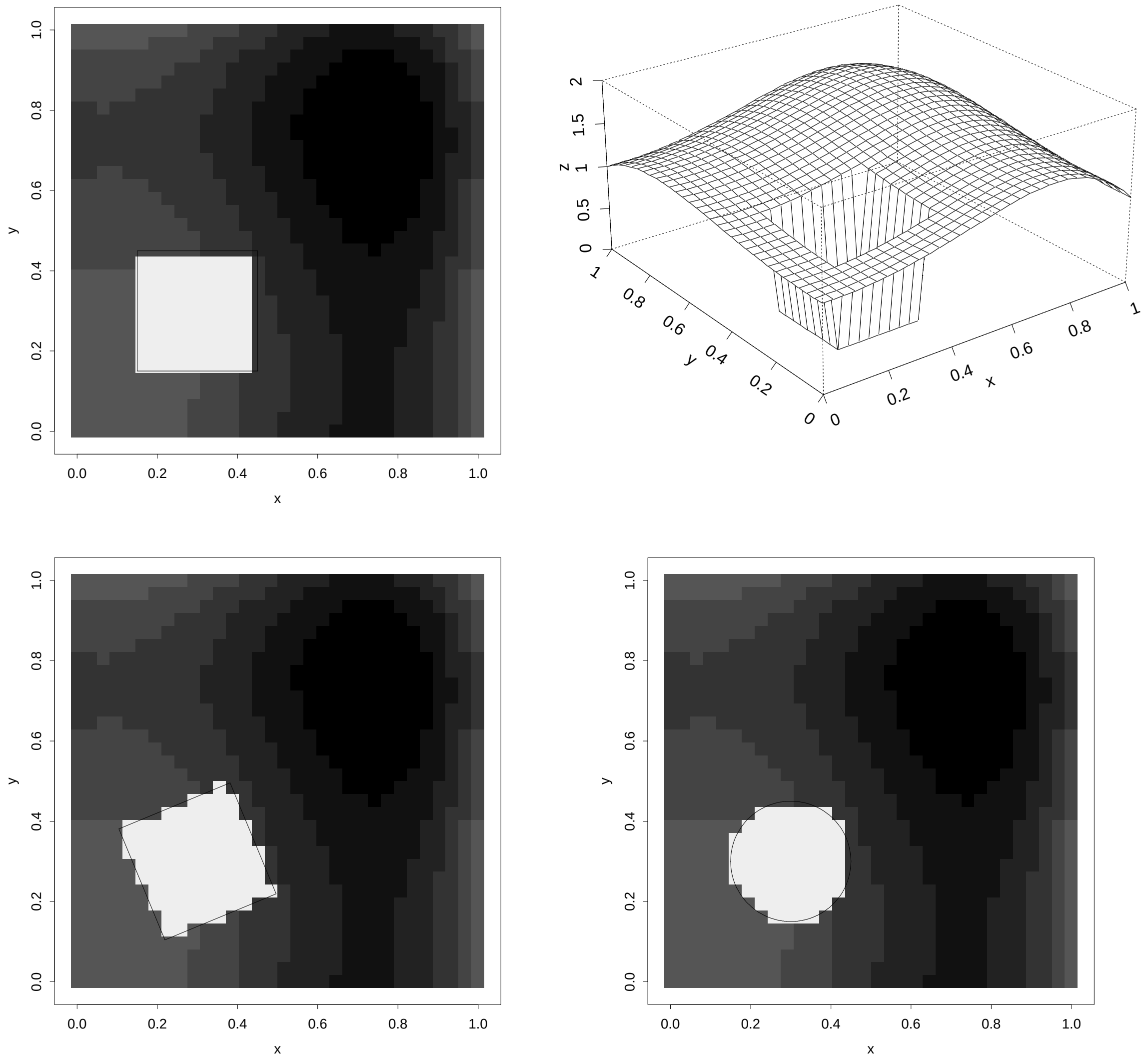


Figure 5.3: Grey scale plot (top left) and perspective plot (top right) of the surface model for the simulation study with a square island aligned with the axes, and grey scale plots for the surface with a square island rotated by $-\frac{\pi}{8}$ about $(0.3, 0.3)$ (bottom left) and a circular island (bottom right).

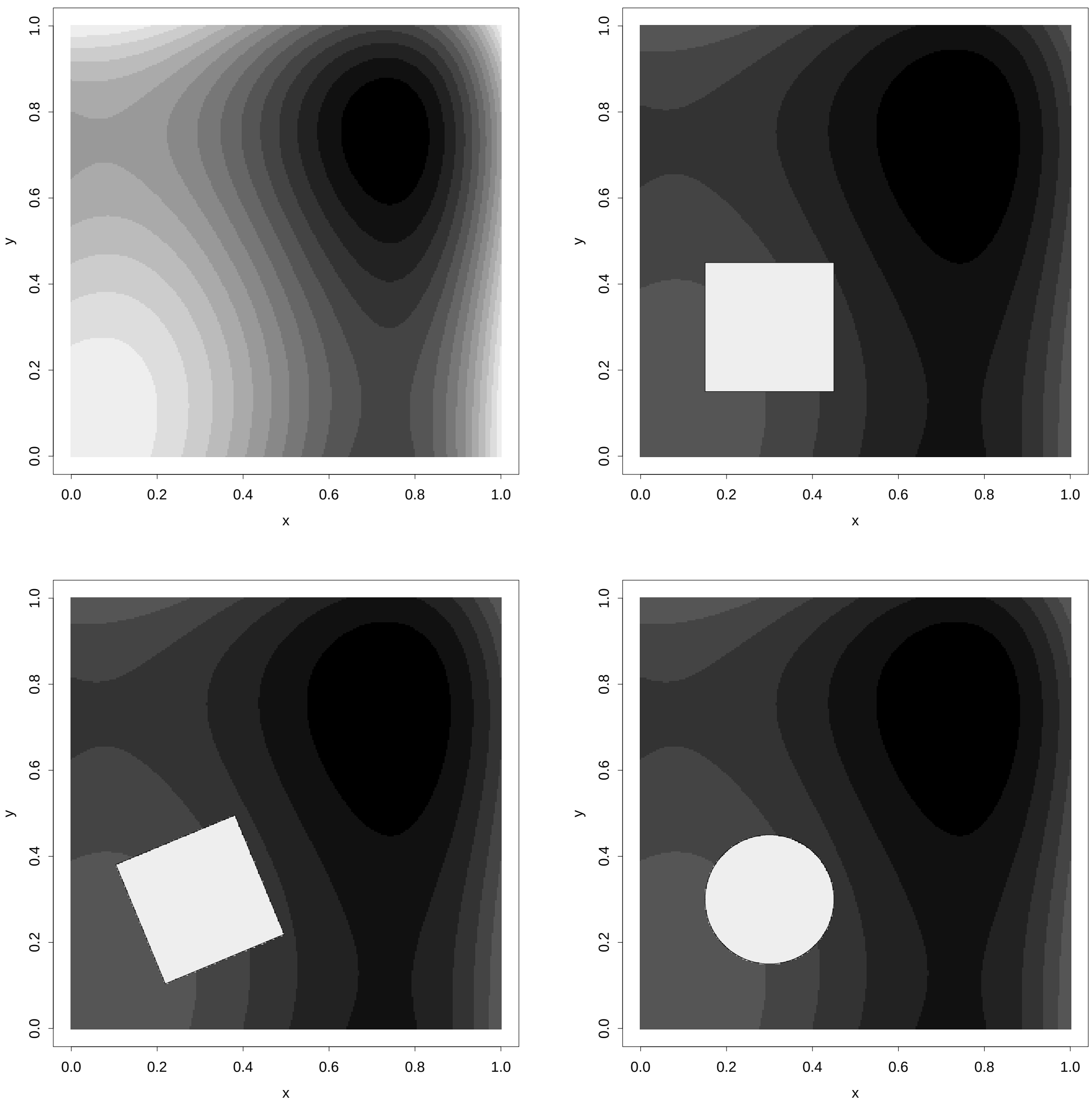


Figure 5.4: Fine resolution grey scale plots for the four simulation surface models displayed in figures 5.2 and 5.3.

generalised least squares in an iterative algorithm. The resulting *prediction* surface is then:

$$\widehat{f}(x,y) = \widehat{m}(x,y) + \widehat{\varepsilon}(x,y) \ .$$

Kriging differs from simple trend surface prediction in that the latter predicts ε as zero.

Note that kriging, as described above, is a method of interpolation, whereas trend surface prediction is a method of smoothing. Since the aim of this simulation study is the recovery of a surface from noisy data, it might seem that trend surface prediction would be more appropriate than kriging. However, as will be illustrated later on, the full kriging model is more robust to the choice of trend surface and is also able to model the discontinuities in the surfaces with islands, which trend surfaces are unable to do.

In the application of kriging in the simulation study a polynomial trend surface will be fitted, with the degree of the trend surface denoted by n_p. Note that the underlying surface is of degree 4. To examine the effect of this parameter, three degrees of trend surface will be fitted: $n_p = 2$, 4 and 6. The covariance function is estimated using the exponential form with parameter $d = 0.7$. This parameter affects how local the resulting covariance function is.

The software used is that described in Chapter 15 of Venables & Ripley (1994), and is available as part of the `spatial` library of functions that is part of the book.

5.2.2 Standard Wavelet Method

The wavelet smoothing methods used here are thoroughly described in Chapter 4. For the simulation study a hard threshold will be used with the threshold constant selected by the universal method (4.12). The simulated data will interpolated to a 32×32 grid, as defined in (4.1)&(4.2). Three different wavelets will be used: the Daubechies extremal phase with $N=2$ and the Daubechies least asymmetric with $N=4$ and $N=10$. For each of these the thresholding will be applied in two ways. Firstly, all five levels of wavelet coefficients will be thresholded. Secondly, the three coarsest scales will be protected, so that the thresholding is applied to levels 3 and 4 only. If the coarsest level to be thresholded is labelled j_t, then the simulation study will use $j_t=0$ and $j_t=3$ for each wavelet.

Since there will be only five levels of wavelet coefficients in the transform, there is not much leeway for choosing the parameter j_t. If j_t is any larger than 3 then little, if any, thresholding will be done. On the other hand, if j_t is much smaller than 3, only very few coefficients will be protected and there will little difference between the resulting surface estimate and that obtained with $j_t=0$.

5.2.3 Special Wavelet Method

For the surfaces with islands it is possible to develop a special wavelet method which uses the information about the island in a direct way, by making a minor addition to the standard wavelet method. Assume that it is known where the island is, what shape it is and that there will be no krill on the island. After the interpolation stage of the standard wavelet method, it is, therefore, possible to set the interpolated values at grid points on the island to zero, along with the necessary values of Γ (i.e. if $\check{z}_{ij}=0$ then $\forall\ k,l\ \Gamma_{ijkl}=0$). The resulting adjusted interpolated data will then be decomposed and thresholded as in the standard wavelet method. As before, three different wavelets will be used with the thresholding applied for $j_t=0$ and $j_t=3$.

It could be argued that the best way to use the information about the island would be to simply set the relevant parts of the final surface estimate to zero. This could be applied equally well to wavelet methods, kriging or any other method of surface estimation. However, the result of doing this is obvious and so not worthy of investigation. In addition, this does not build the feature into the surface model. Doing this would be important in situations where the presence of the feature affected the surrounding surface. Although in this study the underlying smooth surface has been assumed to be independent of the island, this is not very realistic. Some features around the island, such as decreasing depth of water, are bound to affect the krill density. Indeed, this point was hinted at in §5.1 with the discussion of the effect that the flow of water around the island might have on the krill density.

The methods by which the data will be simulated will now be discussed.

Table 5.1: Interpreting the superscript ν in equations (5.1) and (5.2).

<table>
<tr><th>ν</th><th>Type of island</th><th>Region in which data sites are simulated</th></tr>
<tr><td>1</td><td>None</td><td rowspan="4">Unit square</td></tr>
<tr><td>2</td><td>Circular</td></tr>
<tr><td>3</td><td>Square, $\theta=0$</td></tr>
<tr><td>4</td><td>Square, $\theta=-\frac{\pi}{8}$</td></tr>
<tr><td>5</td><td>Circular</td><td rowspan="3">Unit square minus island</td></tr>
<tr><td>6</td><td>Square, $\theta=0$</td></tr>
<tr><td>7</td><td>Square, $\theta=-\frac{\pi}{8}$</td></tr>
</table>

5.3 Simulation Scheme

The basic structure of the simulation scheme will be discussed first, in the context of the smooth, island-free surface. Then in §5.3.2 the treatment of the islands will be considered.

5.3.1 Basic Scheme

For the island-free surface the simulation scheme will generate 100 samples, as follows. For each sample, n locations in the x-y plane will be generated uniformly in the unit square. The true surface will be evaluated at these sites and the values corrupted by i.i.d. Gaussian noise with variance σ_ε^2. The resulting simulated data will be used to form the three different kriging estimates described in §5.2.1 and the six wavelet estimates described in §5.2.2. The surface estimates will be compared by estimating the mean integrated square error:

$$M_k^{\nu\xi}(n, \sigma_\varepsilon) = \mathrm{E}\left[\iint \left(f^\nu(x,y) - \widehat{f}^{\nu\xi}_{n,\sigma_\varepsilon,k}(x,y)\right)^2 dx\, dy\right]. \tag{5.1}$$

The superscript ν takes integer values between 1 and 7 and indicates the type of simulated data, as shown in table 5.1. The values 2 to 7 of ν will be discussed in more detail in §5.3.2. The superscript ξ takes integer values between 1 and 9 and indicates the method of surface estimation used, as shown in table 5.2. The subscript k takes integer values from 1 to 100 and indicates which simulated data sample the surface

Table 5.2: Interpreting the superscript ξ in equations (5.1) and (5.2).

ξ	Method		
		Wavelet	j_t
1	Wavelets	Daubechies extremal phase, $N=2$	0
2			3
3		Daubechies least asymmetric, $N=4$	0
4			3
5		Daubechies least asymmetric, $N=10$	0
6			3
	Kriging	n_p	
7		2	
8		4	
9		6	

estimate (and hence the error estimate) is based on. The function for the particular true surface is f^ν, so note that $f^2 = f^5$, $f^3 = f^6$ and $f^4 = f^7$. The function estimate using method ξ on the k^{th} simulated data set from simulated data type ν with sample size n and noise level σ_ε is $\widehat{f}^{\nu\xi}_{n,\sigma_\varepsilon,k}$. The error for each sample will be calculated as:

$$m_k^{\nu\xi}(n,\sigma_\varepsilon) = \sum_{i=1}^{32}\sum_{j=1}^{32}\left(f^\nu(\check{x}_i,\check{y}_j) - \widehat{f}^{\nu\xi}_{n,\sigma_\varepsilon,k}(\check{x}_i,\check{y}_j)\right)^2 , \tag{5.2}$$

where $\check{x}_i$, $\check{y}_j$ are the locations of the grid points used at the interpolation stage, as given in (4.1) and (4.2) with $J=5$. This is an estimate of $M_k^{\nu\xi}(n,\sigma_\varepsilon)$, but a better estimate will be given by averaging over the 100 replications:

$$\overline{m}^{\nu\xi}(n,\sigma_\varepsilon) = \frac{1}{100}\sum_{k=1}^{100} m_k^{\nu\xi}(n,\sigma_\varepsilon) \ . \tag{5.3}$$

This is the error measure that will be reported for each combination of n, σ_ε, ν and ξ.

This procedure will be repeated for three different sample sizes ($n=40, 80, 160$) and three different noise levels ($\sigma_\varepsilon = 0.01, 0.05, 0.1$).

5.3.2 Island-Related Points

For the surfaces with islands the same basic scheme will be used as for the island-free surface. However, there are some issues regarding the treatment of the island.

Motivated by the real example, the data points should not be generated on the island. This situation then lends itself to use of the special wavelet method described in §5.2.3. This is one approach that will be used ($\nu = 5, 6, 7$), with the same data being given to the kriging methods. However, the wavelet methods are certain to perform much better than kriging in terms of $\overline{m}^{\nu\xi}(n, \sigma_\varepsilon)$ as they have been given information about the island, whereas kriging has not. There is no easy way to adapt the kriging software to give it the same information about the island as the special wavelet method is given, although in principal it may well be possible to adapt the kriging algorithm in such a way. The error will, therefore, be calculated a second time, but only for those grid points not on the island. These *restricted* errors are labelled $r_k^{\nu\xi}(n, \sigma_\varepsilon)$ and calculated as:

$$r_k^{\nu\xi}(n,\sigma_\varepsilon) = \sum_{i,j \in \mathbb{S}^\nu}\sum \left(f^\nu(\check{x}_i, \check{y}_j) - \widehat{f}^{\nu\xi}_{n,\sigma_\varepsilon,k}(\check{x}_i, \check{y}_j)\right)^2 ,$$

where:

$$\mathbb{S}^5 = \Big\{ i,j \in \{(1,\ldots,32),(1,\ldots,32)\} : (\check{x}_i - 0.3)^2 + (\check{y}_i - 0.3)^2 > 0.15^2 \Big\} ,$$

$$\mathbb{S}^6 = \Big\{ i,j \in \{(1,\ldots,32),(1,\ldots,32)\} : |\,\check{x}_i - 0.3\,| > 0.15 \text{ or } |\,\check{y}_i - 0.3\,| > 0.15 \Big\} ,$$

$$\begin{aligned}\mathbb{S}^7 = \Big\{ i,j \in \{(1,\ldots,32),(1,\ldots,32)\} : \big|\,(\check{x}_i - 0.3)\cos\tfrac{\pi}{8} + (\check{y}_i - 0.3)\sin\tfrac{\pi}{8}\,\big| > 0.15 \\ \text{or } \big|\,(\check{y}_i - 0.3)\cos\tfrac{\pi}{8} - (\check{x}_i - 0.3)\sin\tfrac{\pi}{8}\,\big| > 0.15 \Big\} .\end{aligned}$$

The restricted errors are summarised by $\overline{r}^{\nu\xi}(n, \sigma_\varepsilon)$ in the same way that the *full* errors are in (5.3).

As an alternative way of comparing wavelets and kriging for the surfaces with an island, data points will be generated at random over the whole surface, including the island ($\nu = 2, 3, 4$). When the true surface is evaluated at these points, those points that lie on the island are, obviously, set to zero. Noise is then added to all data points. The resulting data is then suitable for use in the standard wavelet method for irregularly spaced data in 2D. This, therefore, provides a comparison of kriging and wavelets in the general setting of noisy observations from a surface containing a discontinuous feature.

5.4 Results

This section will start with an explanation of the ways in which the results of the simulation study are presented. This will be followed, in §5.4.2, by a brief discussion of some general points about the results for the kriging methods. The simulation results will then be discussed in three groups. The results for $\nu = 1$, the surface with no island, will be dealt with in §5.4.3. The results for $\nu = 2, 3, 4$, the surfaces with islands analysed using the standard wavelet method, will be discussed in §5.4.4 and the results for the special wavelet method, $\nu = 5, 6, 7$, in §5.4.5.

5.4.1 Presentation

The results of the simulation study are summarised in tables 5.11 – 5.17, which can be found at the end of this chapter (pages 159 – 165). For each combination of noise, sample size and method, these tables give $\overline{m}^{\nu\xi}(n, \sigma_\varepsilon)$. For the surfaces with islands, the figure in brackets is the restricted error $\overline{r}^{\nu\xi}(n, \sigma_\varepsilon)$. Let the "best" performing wavelet method be judged in terms of minimum average squared error, with the full and restricted errors judged separately. These minimum errors are denoted $\overline{m}^{\nu*}(n, \sigma_\varepsilon)$ and $\overline{r}^{\nu*}(n, \sigma_\varepsilon)$, where for each combination of ν, n and σ_ε:

$$\left.\begin{array}{rcl} \overline{m}^{\nu\xi}(n, \sigma_\varepsilon) & \geqslant & \overline{m}^{\nu*}(n, \sigma_\varepsilon) \\ \overline{r}^{\nu\xi}(n, \sigma_\varepsilon) & \geqslant & \overline{r}^{\nu*}(n, \sigma_\varepsilon) \end{array}\right\} \ \forall\ \xi = 1, \ldots, 6\ .$$

In tables 5.11 – 5.17 $\overline{m}^{\nu*}(n, \sigma_\varepsilon)$ and $\overline{r}^{\nu*}(n, \sigma_\varepsilon)$ are indicated by asterisks.

Figure 5.5, on page 136, shows histograms of the deviations of the errors from the median, scaled by the median absolute deviation from the median. For each combination of noise level and sample size, the deviation of each of the error terms from the 100 replications from their median is scaled by their MAD:

$$\frac{m_k^{\nu\xi}(n, \sigma_\varepsilon) - \underset{k=1,\ldots,100}{\text{median}}\left(m_k^{\nu\xi}(n, \sigma_\varepsilon)\right)}{\underset{k=1,\ldots,100}{\text{MAD}}\left(m_k^{\nu\xi}(n, \sigma_\varepsilon)\right)} \ ,$$

and similarly for the restricted errors. The resulting normalised errors are then combined for all the noise levels and sample sizes for all seven types of simulated data. Note that both the full and restricted errors are used, even though they are clearly not independent. For each combination of n, σ_ε, ν and k the full and restricted errors

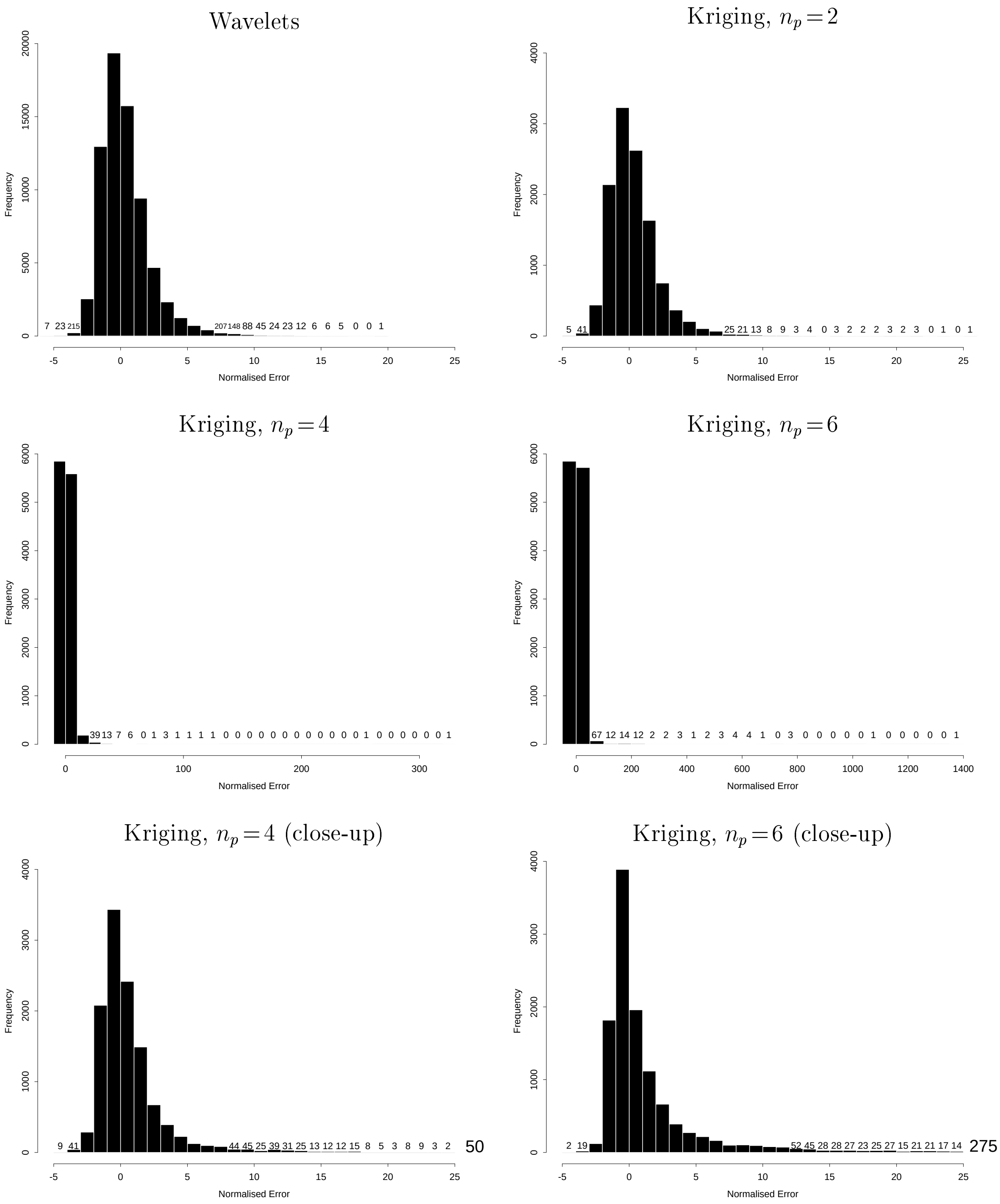


Figure 5.5: Histograms of the normalised errors collated from all simulations. The normalisation consists of subtracting the median and dividing by the median absolute deviation from the median. The wavelet methods are all collected together in the top left plot. The top right plot is for kriging with $n_p = 2$. The middle row shows the normalised errors for kriging with $n_p = 4$ and $n_p = 6$. The bottom row shows close-ups of the plots on the middle row over the interval $[-5, 25]$. The numbers on the plots indicate the heights of the small and invisible bins, and the large numbers at the right hand ends of the plots on the bottom row indicate the number of normalised errors that are > 25.

Table 5.3: For each simulation, the number of error estimates out of the 100 replications for which the normalised error is greater than 25, for the kriging estimates with $n_p=4$ and $n_p=6$. These results are displayed for both the full and restricted errors. The total number of outlying errors for each combination of n and σ_ε are given in the bottom two rows.

	σ_ε	0.01			0.05			0.1		
	n	40	80	160	40	80	160	40	80	160
Table	n_p									
5.11 ($\nu=1$)	4	1	1	0	2	0	0	0	0	0
	6	4	2	0	6	1	2	5	0	1
5.12 ($\nu=2$)	4	1 (1)	0 (0)	1 (1)	5 (0)	1 (1)	0 (0)	2 (2)	0 (0)	0 (0)
	6	6 (6)	0 (0)	0 (0)	6 (6)	4 (4)	0 (0)	4 (4)	0 (0)	1 (1)
5.13 ($\nu=3$)	4	4 (3)	1 (1)	0 (0)	0 (0)	1 (1)	0 (0)	2 (1)	0 (0)	0 (0)
	6	5 (5)	3 (3)	0 (0)	7 (7)	2 (2)	1 (1)	4 (4)	1 (1)	1 (1)
5.14 ($\nu=4$)	4	1 (1)	1 (1)	0 (0)	0 (0)	0 (0)	0 (0)	1 (1)	2 (2)	0 (0)
	6	9 (9)	2 (1)	0 (0)	4 (4)	4 (4)	0 (0)	5 (5)	3 (3)	1 (1)
5.15 ($\nu=5$)	4	0 (0)	0 (0)	0 (0)	0 (1)	0 (0)	0 (0)	0 (1)	0 (0)	0 (0)
	6	3 (3)	0 (2)	0 (0)	8 (8)	1 (2)	0 (0)	6 (6)	0 (0)	0 (1)
5.16 ($\nu=6$)	4	0 (0)	0 (1)	0 (0)	0 (2)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)
	6	3 (5)	0 (2)	0 (0)	6 (6)	0 (1)	0 (1)	7 (7)	1 (4)	0 (0)
5.17 ($\nu=7$)	4	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	1 (1)	0 (0)	0 (0)
	6	2 (3)	0 (0)	0 (2)	2 (4)	0 (1)	0 (0)	5 (5)	0 (2)	0 (0)
Sum	4	12	6	2	10	4	0	12	4	0
	6	63	15	2	74	26	5	67	15	8

are based on the same surface estimate calculated using the same simulated data. The results from all six different wavelet models are grouped together as well, whereas the three kriging methods are kept separate. It is, therefore, important to remember that there six times as many observations in the histogram for the wavelet methods than there are in any of the other histograms. In addition, for each combination of n, σ_ε, ν and k the errors from the six different wavelet methods are correlated, as they are based on the same simulated data.

The top row of figure 5.5 contains the histograms for the wavelet methods and for kriging with $n_p = 2$. The middle row contains the histograms for kriging with $n_p = 4$ and $n_p = 6$. The bottom row contains close-ups of the histograms in the middle row over the interval $[-5, 25]$. Note that this only truncates the data at the right hand end. For the wavelets, kriging with $n_p = 2$ and the close-ups the bin width is 1. For kriging with $n_p = 4$ the bin width is 10 and for kriging with $n_p = 6$ the bin width is 50. The numbers on all the plots give the height of the bins that are very short or invisible. The numbers in the larger font size at the right hand side of the two close-up plots indicate the number of observations that are greater than 25. The distribution of these outlying errors between the different experiments is displayed in table 5.3 on page 137. The bottom row displays the total number of these outliers present in the results for each combination of noise and sample size.

Table 5.4, on page 139, shows, for each of tables 5.11 – 5.17, the number of replications for which the best performing wavelet method out-performs the kriging methods with $n_p = 2$ and $n_p = 4$ (denote this n_s). If a two-sided sign test is performed to test the null hypothesis that the median difference between the errors is zero, then H_0 is accepted at the 5% level if $40 \leqslant n_s \leqslant 60$. The shaded cells indicate that there is significant evidence that the median difference is not zero, with dark grey indicating that wavelet methods are superior (i.e. $n_s > 60$) and light grey that kriging is superior (i.e. $n_s < 40$). Note that for tables 5.15 – 5.17, the results of the simulations where the special wavelet method was used, only the restricted errors have been compared, because the errors for the whole surface are, as expected, vastly superior for the wavelet methods. This will be discussed in more detail later. Of course, the wavelet method with the smallest average squared error need not be the one that gives the largest number of successes, but since the errors for the wavelet models appear reasonably

Table 5.4: For each value of ν, n and σ_ε, the number of occasions out of the 100 replications that the best performing wavelet method has smaller error than the kriging estimates with n_p as shown. The figures in brackets are the results for the restricted errors. Note that for tables 5.15 – 5.17 only the restricted errors are used. The shaded cells indicate significant evidence that the median difference between the errors is non-zero, with light grey indicating that the kriging method is superior and dark grey indicating that wavelet method is superior.

		σ_ε	0.01			0.05			0.1		
		n	40	80	160	40	80	160	40	80	160
Table	n_p										
5.11	2		12	5	0	29	25	43	58	73	79
($\nu=1$)	4		0	0	0	40	10	26	80	84	81
5.12	2		55	18	5	59	27	12	62	40	31
			(63)	(33)	(17)	(67)	(59)	(30)	(75)	(64)	(67)
($\nu=2$)	4		85	61	19	86	61	23	92	78	47
			(91)	(76)	(34)	(94)	(82)	(45)	(97)	(92)	(85)
5.13	2		63	33	19	60	34	23	70	41	42
			(76)	(58)	(26)	(79)	(62)	(51)	(82)	(78)	(74)
($\nu=3$)	4		84	67	37	91	81	50	90	81	58
			(91)	(91)	(49)	(97)	(95)	(72)	(96)	(96)	(93)
5.14	2		63	30	12	53	36	12	56	35	26
			(77)	(57)	(25)	(70)	(65)	(38)	(80)	(75)	(74)
($\nu=4$)	4		87	69	24	91	72	27	96	75	54
			(94)	(90)	(50)	(97)	(90)	(58)	(100)	(93)	(88)
5.15	2		(16)	(0)	(0)	(29)	(14)	(2)	(56)	(43)	(59)
($\nu=5$)	4		(0)	(0)	(0)	(34)	(5)	(2)	(93)	(60)	(72)
5.16	2		(20)	(2)	(0)	(37)	(9)	(4)	(45)	(37)	(69)
($\nu=6$)	4		(0)	(0)	(0)	(38)	(5)	(2)	(82)	(62)	(70)
5.17	2		(22)	(0)	(0)	(28)	(13)	(3)	(38)	(33)	(71)
($\nu=7$)	4		(0)	(0)	(0)	(42)	(12)	(2)	(79)	(64)	(78)

Table 5.5: Normalised errors associated with the island for the simulations in tables 5.13 and 5.14. The averaged squared errors are normalised by the number of grid points over which they are calculated (81 when $\theta = 0$ and 87 when $\theta = -\frac{\pi}{8}$). The asterisks indicate where the normalised error is smaller when $\theta = -\frac{\pi}{8}$ than when $\theta = 0$.

				$\sigma_\varepsilon = 0.01$			$\sigma_\varepsilon = 0.05$			$\sigma_\varepsilon = 0.1$		
				$n = 40$	$n = 80$	$n = 160$	$n = 40$	$n = 80$	$n = 160$	$n = 40$	$n = 80$	$n = 160$
Wavelets	Wavelet	j_t	θ									
	Daubechies Extremal Phase, $N = 2$	0	0	0.400	0.214	0.131	0.433	0.226	0.134	0.604	0.317	0.163
			$\frac{\pi}{8}$	0.400	0.237	0.171	0.523	0.272	0.171	0.607	0.334	0.203
		3	0	0.364	0.209	0.130	0.350	0.196	0.127	0.353	0.220	0.125
			$\frac{\pi}{8}$	0.371	0.233	0.171	0.437	0.249	0.165	0.372	0.246	0.172
	Daubechies Least Asymmetric, $N = 4$	0	0	0.382	0.219	0.132	0.407	0.228	0.143	0.562	0.291	0.175
			$\frac{\pi}{8}$	0.388	0.238	0.171	0.488	0.265	0.175	*0.539	0.304	0.203
		3	0	0.367	0.211	0.131	0.355	0.203	0.133	0.357	0.233	0.139
			$\frac{\pi}{8}$	0.371	0.234	0.171	0.437	0.249	0.165	0.370	0.247	0.174
	Daubechies Least Asymmetric, $N = 10$	0	0	0.442	0.215	0.132	0.477	0.220	0.136	0.607	0.329	0.165
			$\frac{\pi}{8}$	*0.441	0.243	0.173	0.554	0.283	0.178	0.649	0.354	0.219
		3	0	0.363	0.209	0.131	0.349	0.198	0.129	0.349	0.223	0.134
			$\frac{\pi}{8}$	0.370	0.234	0.171	0.434	0.249	0.167	0.367	0.244	0.174
Kriging	n_p											
	2		0	0.319	0.178	0.118	0.289	0.166	0.112	0.313	0.176	0.105
			$\frac{\pi}{8}$	0.327	0.200	0.151	0.376	0.218	0.143	*0.301	0.202	0.145
	4		0	0.254	0.162	0.115	0.226	0.150	0.109	0.237	0.159	0.102
			$\frac{\pi}{8}$	0.265	0.183	0.148	0.312	0.201	0.140	0.241	0.187	0.142
	6		0	0.448	0.144	0.107	0.456	0.134	0.102	0.408	0.145	0.096
			$\frac{\pi}{8}$	0.923	0.168	0.139	0.491	0.175	0.132	0.756	0.170	0.135

well behaved, in that the mean and median are generally close, the mean error should, at least, be a useful guide.

For the two square islands, when the standard wavelet method is applied ($\nu=3$ & 4, tables 5.13 & 5.14), table 5.5, on page 140, shows the average squared error associated with the island $(\overline{m}^{\nu\xi}(n,\sigma_\varepsilon) - \overline{r}^{\nu\xi}(n,\sigma_\varepsilon))$, divided by the number of grid points that lie on the island. For the square island with $\theta=0$, 81 grid points are covered by the island, and for the square island with $\theta=-\frac{\pi}{8}$, 87 grid points are covered. Note that the number of grid points that the island covers is the number of individual error terms that contribute to the error associated with the island, so this is scaling allows these errors to be compared fairly.

5.4.2 General Comment on Kriging Results

Perhaps the most striking feature of the results in tables 5.11 – 5.17 is the variation in the results of the kriging estimates for each combination of ν, n and σ_ε. The choice of n_p is obviously very important and has much more effect than the choice of wavelet and j_t does on the performance of the wavelet methods. Of particular note is the poor performance of kriging with $n_p = 6$, especially for small sample sizes and high noise levels. Figure 5.5 illustrates that kriging with $n_p=6$ (and to a lesser extent with $n_p=4$) has a large number of outlying errors. In other words, when it gets it wrong it can go badly wrong. It is interesting to note from table 5.3 that the distribution of these extreme values clearly depends on the sample size n, but that the noise level σ_ε does not appear to have much effect. This leads to a probable explanation for these poor results. A polynomial of degree 6 requires 28 coefficients to be estimated for the trend surface alone. When there are only 40 data points to start with one can not hope to do too well. When $n=160$, however, the performance of kriging with $n_p=6$ is generally comparable to that with $n_p=2$ and 4 and sometimes it does better.

Figure 5.5 also illustrates that apart from these freak results, the distribution of the errors about the median is similar for all methods. It is, therefore, fair to say that the mean of the errors will be skewed by these outlying results. It could, therefore, be argued that the median would provide a better summary of the errors than the mean. However, these outliers are not mistakes and so need to be taken into consideration. Besides, where kriging with $n_p=6$ performs badly in terms of mean error, the median

Table 5.6: Squared errors for the kriging estimates shown in figure 5.6, which are based on a simulated set of 40 points from the surface with no island with noise level 0.05.

	Trend Surface	Prediction Surface
$n_p\!=\!2$	15.60	3.15
$n_p\!=\!4$	1.84	2.07
$n_p\!=\!6$	13.49	13.42

error is still generally much larger than any of the errors for $n_p\!=\!2$ or $n_p\!=\!4$.

On the other hand, notice that $\xi\!=\!7$ generally does better than $\xi\!=\!8$, even though the true order of the polynomial is 4. This can be explained in two ways. Firstly, figure 5.5 illustrates that $\xi\!=\!8$ is still prone to some large outliers. Secondly, recall that kriging consists of two parts. The fluctuation term ε can adapt a quadratic surface to fit closely to the true quartic. This will be examined again later, in §5.4.3 and §5.4.4, and is illustrated in figures 5.6 and 5.8 on pages 143 & 149.

5.4.3 Surface Without Island - Table 5.11

General Trends

As might be expected, the errors are monotone decreasing in n and monotone increasing in σ_ε:

$$\overline{m}^{1,\xi}(40,\sigma_\varepsilon) \;>\; \overline{m}^{1,\xi}(80,\sigma_\varepsilon) > \overline{m}^{1,\xi}(160,\sigma_\varepsilon) \begin{cases} \text{for } \xi = 1,\dots,9 \\ \text{and } \sigma_\varepsilon = 0.01, 0.05, 0.1\ , \end{cases}$$

$$\overline{m}^{1,\xi}(n,0.01) < \overline{m}^{1,\xi}(n,0.05) < \;\overline{m}^{1,\xi}(n,0.1) \begin{cases} \text{for } \xi = 1,\dots,9 \\ \text{and } n = 40, 80, 160\ . \end{cases}$$

Kriging

The best kriging method for the lowest noise level is that with $n_p\!=\!4$ ($\xi\!=\!8$), for the highest noise level it is $\xi\!=\!7$, with the best for $\sigma_\varepsilon\!=\!0.05$ depending on the sample size. To illustrate the roles of the trend and fluctuation terms in the kriging estimate, one set of simulated data will now be examined more thoroughly. This set is not one that comes from the main simulation study.

Figure 5.6 shows the trend, fluctuation and prediction surfaces for one realisation

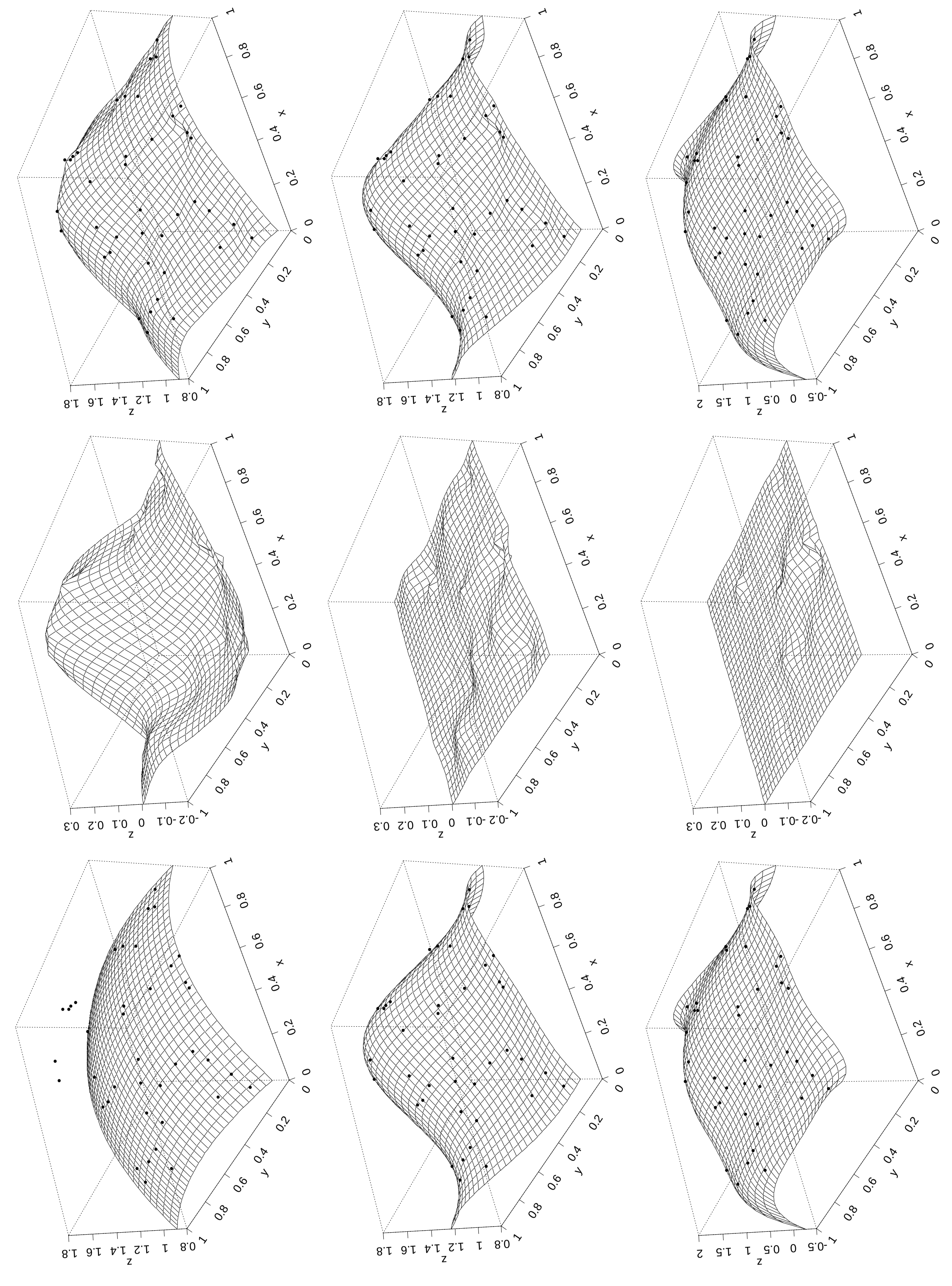


Figure 5.6: Trend (left-hand column), fluctuation (middle column) and prediction (right-hand column) surfaces for kriging estimates of a simulated set of 40 points from the surface with no island with noise level 0.05. The first row is for $n_p = 2$, the second for $n_p = 4$ and the third for $n_p = 6$. Note that all three fluctuation surfaces are plotted to the same scale and that for each value of n_p both the trend and prediction surfaces are plotted to the same scale.

with $n=40$ and $\sigma_\varepsilon=0.05$ for $n_p=2,4,6$. In addition, table 5.6 (page 142) shows the squared errors of both the trend and prediction surfaces. As would be expected, the trend surface alone with $n_p=2$ does not fit the data well, as the true surface is really quartic. However, with the addition of the fluctuation term, the prediction surface does fit the data quite well. The fluctuation surface can be seen to contain much of the shape of the true surface. Naturally, when $n_p=4$ the trend surface estimates the true surface well and in fact, the prediction surface does slightly less well. With $n_p=6$ neither the trend or prediction surfaces do very well, and this can be seen to be caused by some artifacts appearing around the boundary of the unit square. For both $n_p=4$ and $n_p=6$ the fluctuation contains only small scale details. It could be argued that it would be best to only look at the trend surface for $n_p=4$, but this assumes too much that wouldn't be known in a real situation where the true surface is unlikely to be exactly polynomial anyway. The cost of adding in the fluctuation term is small and in general having this term vastly improves the performance if the value of n_p is under-specified.

Wavelets

The best performing wavelet method for $\sigma_\varepsilon=0.01$ and $\sigma_\varepsilon=0.05$ is $\xi=6$, except for $n=40$ and $\sigma_\varepsilon=0.05$, for which $\xi=4$ is the best. This is also the best method for $\sigma_\varepsilon=0.1$ with $n=40$ and 80, and $\xi=5$ performs best for $n=160$. It seems that the less smooth $N=4$ Daubechies wavelet does better for smaller samples and higher noise levels. This may be connected with the vanishing moments property of wavelets. Recall from (2.20), (2.21) and (2.22) that the Daubechies wavelet with filter number N has $N-1$ vanishing moments. The true wavelet coefficients of the underlying surface will, therefore, all be zero for $N=10$, but not for $N=4$. However, this will not be exactly true for the DWT coefficients, because of the approximation $c_{Jk}\approx 2^{-J/2}f(t_k)$ made in the DWT algorithm (see §2.2.2) and also because of boundary effects. However, it will be true that the DWT coefficients of a noise-free sample from the true surface will be smaller for the wavelet with $N=10$ than those for wavelets with $N=2$ or $N=4$. A possible explanation, therefore, for why $\xi=6$ is generally the best performing wavelet method is that these wavelet coefficients provide the sparsest representation of the true surface and so are better able to distinguish between signal and noise at the thresholding stage.

On examining the average squared errors for the wavelet methods it is apparent that $\overline{m}^{1,\xi}(n,\sigma_\varepsilon)$ is smaller for each combination of wavelet, n and σ_ε when $j_t=3$ than when $j_t=0$. There is one exception to this, as follows:

$$\overline{m}^{1,5}(160,0.1) < \overline{m}^{1,6}(160,0.1) \ .$$

Notice also that when $j_t=3$, there is less difference between the performance of the three different wavelets than when $j_t=0$.

The relative differences between the errors when $j_t=0$ and $j_t=3$ (measured as, for example, $\overline{m}^{1,4}(80,0.05)/\overline{m}^{1,3}(80,0.05)$) are in general bigger when n is small and σ_ε is large. However, it is interesting to note that although the relative differences are similar for all three wavelets when $\sigma_\varepsilon=0.01$, they are noticeably smaller for the Daubechies least asymmetric wavelet, $N=10$ for the higher noise levels. This can be illustrated with the following example:

$$\begin{array}{ll} & \overline{m}^{1,2}(160,0.01)/\overline{m}^{1,1}(160,0.01) = 0.947 \\ \text{and} & \overline{m}^{1,6}(160,0.01)/\overline{m}^{1,5}(160,0.01) = 0.948 \ , \\ \text{but} & \overline{m}^{1,2}(40,0.1)/\overline{m}^{1,1}(40,0.1) = 0.437 \\ \text{and} & \overline{m}^{1,6}(40,0.1)/\overline{m}^{1,5}(40,0.1) = 0.827 \ . \end{array}$$

In other words, at the higher noise levels, the improvement on using $j_t=3$ rather than $j_t=0$ is smaller for the least asymmetric wavelet on $N=10$ than for the other two. The vanishing moments property also offers an explanation for this pattern. If it is easier to distinguish between signal and noise in the coefficients when $N=10$, then using using $j_t=3$ offers less of an advantage because there is a smaller gain in terms of protecting important signal coefficients than for the less smooth wavelets with lower filter numbers.

To illustrate wavelets in action on data of type $\nu=1$, figure 5.7 (page 147) contains wavelet estimates based on the same data as was used to construct figure 5.6 for the examination of kriging in the previous subsection. The squared errors for each surface are displayed in table 5.7 (page 146). It is interesting to note that the difference between

Table 5.7: Squared errors for the wavelet estimates shown in figure 5.7, which are constructed using the standard wavelet method for irregularly spaced data. They are based on a simulated set of 40 points from the surface with no island with noise level 0.05.

Wavelet	$j_t=0$	$j_t=3$
Daubechies extremal phase, $N=2$	6.82	4.35
Daubechies least asymmetric, $N=4$	6.92	3.80
Daubechies least asymmetric, $N=10$	4.90	3.82

the pictures for $j_t=0$ and $j_t=3$ is much less for the Daubechies least asymmetric wavelet on $N=10$ than for the other two wavelets. This is reflected in the error values in table 5.7.

Comparison

Consider the relevant rows of table 5.4. Notice immediately that wavelets only perform significantly better, in terms of the number of successes when $\sigma_\varepsilon = 0.1$. It is also interesting to note that for fixed n, the number of successes against kriging with both $n=4$ and $n=2$ increase with the noise level σ_ε. In other words the relative performance of wavelet methods improves with increasing σ_ε. There is no particular pattern as to what happens for fixed σ_ε and varying n.

5.4.4 Surfaces with Islands using the Standard Wavelet Method for Irregularly Spaced Data - Tables 5.12 – 5.14

General Trends

To get an indication of the effect of the island, compare the restricted error measures in tables 5.12 – 5.14 with the errors for the island free surface in table 5.11. It is clear that $\overline{r}^{\nu\xi}(n,\sigma_\varepsilon) >> \overline{m}^{\nu\xi}(n,\sigma_\varepsilon)$, for $\nu = 2,3,4$, even though the restricted errors are measured over fewer points. The presence of the island evidently damages the ability of all the methods to estimate the smooth part of the surface.

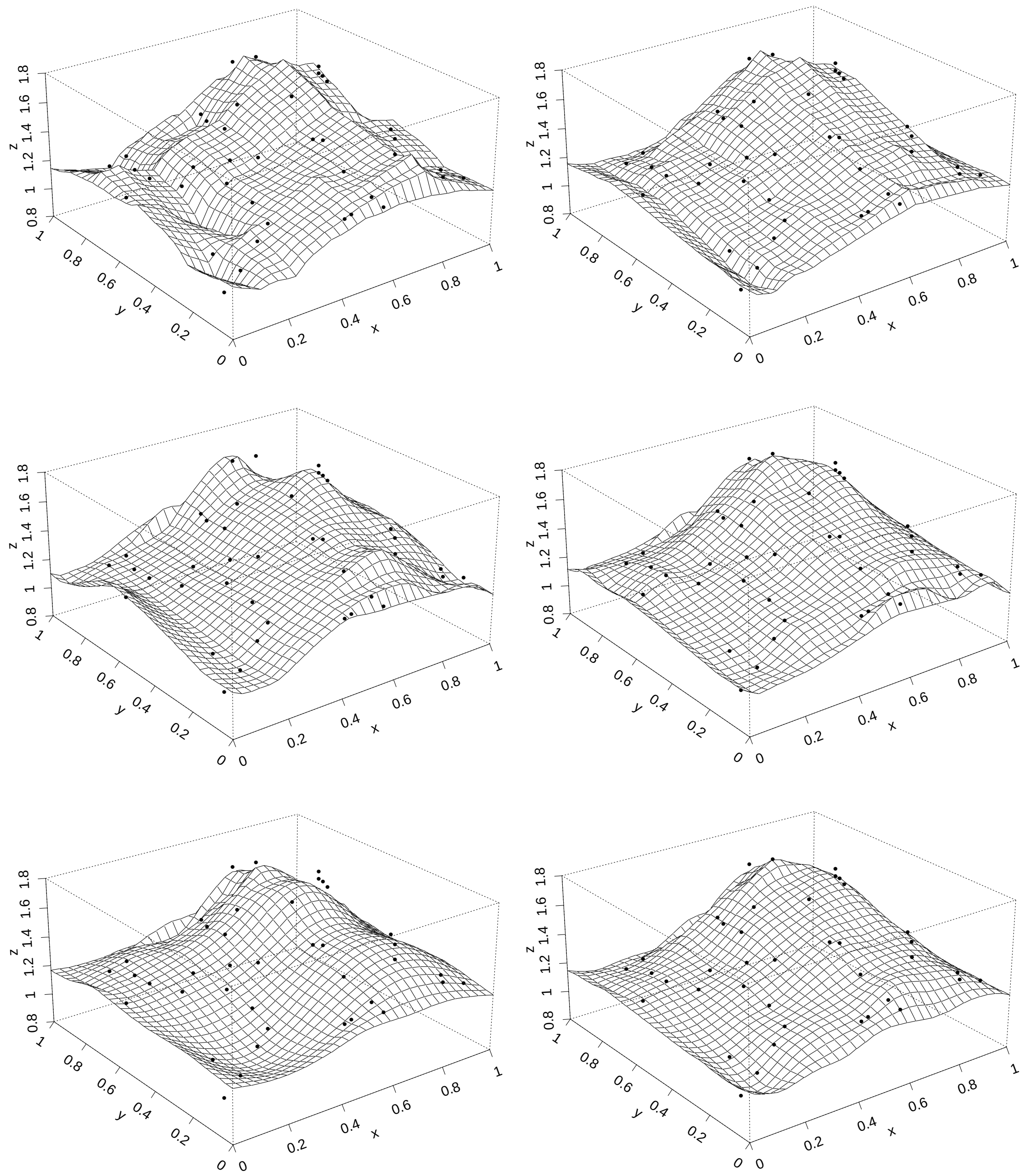


Figure 5.7: Wavelet estimates of the simulated set of 40 points from the surface with no island with noise level 0.05, as used in figure 5.6. The left-hand column contains the estimates with $j_t = 0$ and the right-hand column those with $j_t = 3$. The top row contains the estimates found using the Daubechies extremal phase wavelet on $N = 2$, the middle row those using the Daubechies least asymmetric wavelet on $N = 4$ and the bottom row those using the Daubechies least asymmetric wavelet on $N = 10$.

Table 5.8: Squared errors for the kriging estimates shown in figure 5.8, which are based on a simulated set of 80 points from the surface with a square island aligned with the axes with noise level 0.05.

	Trend Surface	Prediction Surface
$n_p = 2$	168.03	39.39
$n_p = 4$	69.83	45.66
$n_p = 6$	155.46	155.78

It is interesting to note that, although both the full and restricted errors are monotone decreasing in the sample size n:

$$\left.\begin{array}{ccccc} \overline{m}^{\nu,\xi}(40,\sigma_\varepsilon) & > & \overline{m}^{\nu,\xi}(80,\sigma_\varepsilon) & > & \overline{m}^{\nu,\xi}(160,\sigma_\varepsilon) \\ \overline{r}^{\nu,\xi}(40,\sigma_\varepsilon) & > & \overline{r}^{\nu,\xi}(80,\sigma_\varepsilon) & > & \overline{r}^{\nu,\xi}(160,\sigma_\varepsilon) \end{array}\right\} \begin{array}{l} \text{for } \xi = 1,\ldots,9\ , \nu = 2,3,4 \\ \text{and } \sigma_\varepsilon = 0.01, 0.05, 0.1\ , \end{array}$$

they are not monotone increasing with σ_ε. Although the general trend is for increasing error with σ_ε, there are several exceptions to this. For example:

$$\overline{m}^{2,6}(40, 0.05) > \overline{m}^{2,6}(40, 0.1)\ ,$$

$$\overline{m}^{3,8}(40, 0.01) > \overline{m}^{3,8}(40, 0.05)\ ,$$

$$\overline{r}^{4,4}(80, 0.01) > \overline{r}^{4,4}(80, 0.05)\ .$$

Kriging

The best kriging method is $\xi = 7$ for all cases, with both the restricted and the full errors, for all three tables ($\nu = 2, 3, 4$). This is in contrast to the errors for $\nu = 1$ where $n_p = 4$ proved superior in some cases. Recall that the smooth part of the surface is a polynomial of degree 4. It might have been expected that the extra parameters available with $n_p = 4$ or $n_p = 6$ would help to model the island, but this is apparently not the case. Perhaps the island is really modelled by the error part of the kriging model. After all, a polynomial will not be able to model the discontinuity involved in the island well, unless it is of very high order and a very large sample of data is available. To investigate this further, a specific example will be looked at in more detail. The simulated data used will not be one of the realisations from the main simulation study.

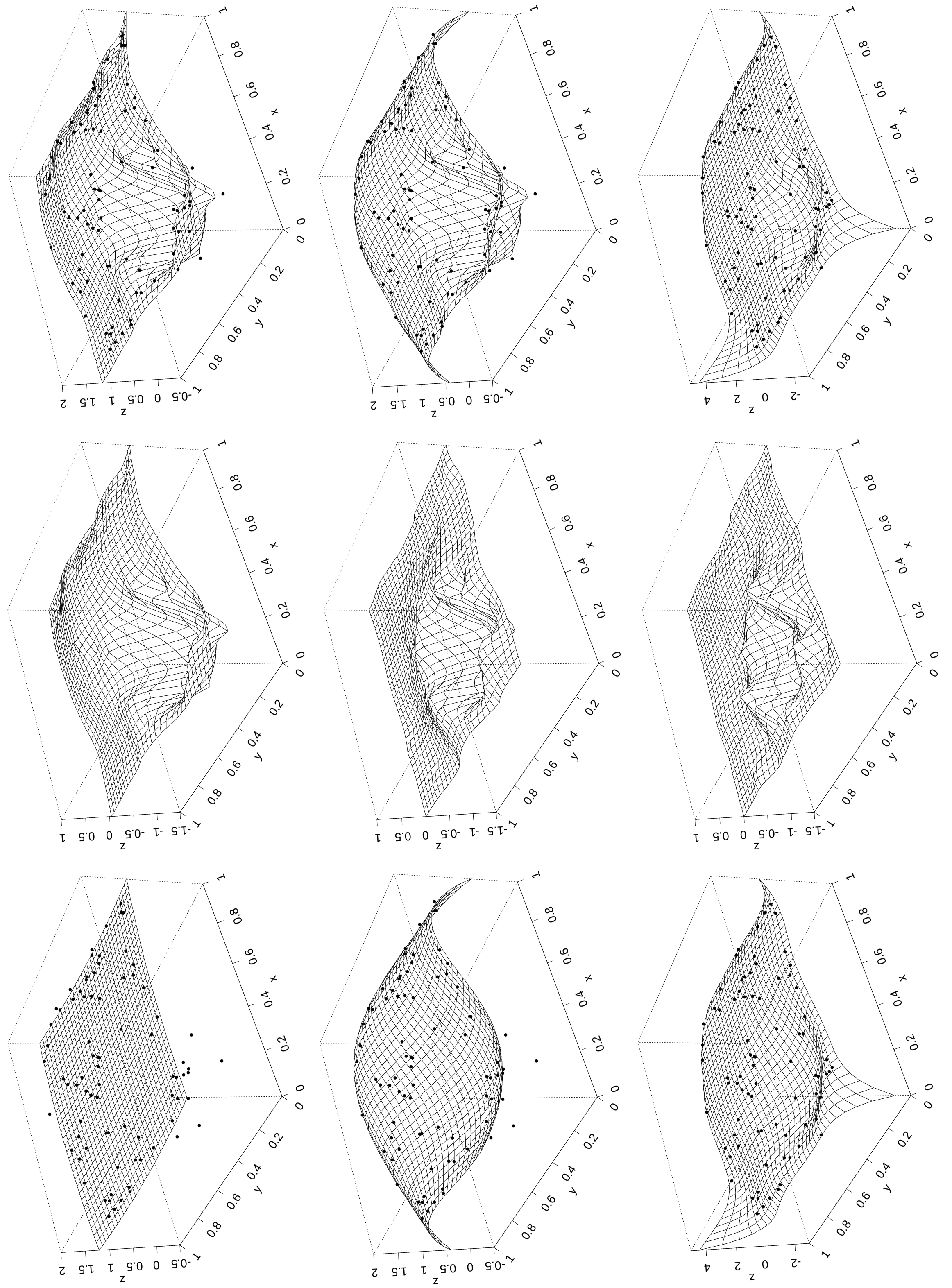


Figure 5.8: Trend (left-hand column), fluctuation (middle column) and prediction (right-hand column) surfaces for kriging estimates of a simulated set of 80 points from the surface with a square island aligned with the axes with noise level 0.05. The first row is for $n_p = 2$, the second for $n_p = 4$ and the third for $n_p = 6$. Note that all three fluctuation surfaces are plotted to the same scale and that for each value of n_p both the trend and prediction surfaces are plotted to the same scale.

Figure 5.8 (page 149) shows the trend, fluctuation and prediction surfaces for one realisation of data from the surface with the square island aligned with the axes ($\theta=0$) with $n=80$ and $\sigma_\varepsilon=0.05$ for $n_p=2,4,6$. In addition, table 5.8 (page 148) shows the squared errors of both the trend and prediction surfaces. The trend surface with $n_p=2$ is little more than a constant, and nearly all the detail in the prediction surface is modelled by the fluctuation term. This allows the kriging to model the shape of the true surface quite well. When $n_p=4$ the trend surface plays much more of a role, but again the detail around the island comes from the fluctuation term. Notice that both estimates go astray round the edges, as happened for $n_p=6$ with the surface with no island in figure 5.6. This is again a problem with $n_p=6$ here. However, notice that the trend surface does model the shape of the island, with the fluctuation adding in some extra detail around the edges.

Wavelets

As with $\nu=1$, the error is generally less for $j_t=3$ than $j_t=0$. This is in fact always true for the full errors, with only two exceptions for the restricted errors:

$$\overline{r}^{2,5}(40,0.1) < \overline{r}^{2,6}(40,0.1) \ ,$$

$$\overline{r}^{3,5}(40,0.1) < \overline{r}^{3,6}(40,0.1) \ .$$

However, the relative improvement with the full error measure is similar for all three wavelets used, although when the restricted measure is considered, the improvement is less for the least asymmetric wavelet, $N=10$, than for the other two, as was observed for the surface without an island.

The best performing method is again generally $\xi=6$. The exceptions to this for the full errors are that $\xi=2$ is superior for three cases with $n=160$. These are for $\sigma_\varepsilon=0.1$ with $\nu=3$ and $\nu=4$ and for $\sigma_\varepsilon=0.05$ with $\nu=4$. For the restricted errors, $\xi=4$ is superior for $n=40$ and $\sigma_\varepsilon=0.05$ for all three tables ($\nu=2,3,4$) and $\xi=5$ is best for $n=40$, $\sigma_\varepsilon=0.1$ for $\nu=2,3$.

To illustrate the standard wavelet method in action on data of type $\nu=2,3$ or 4, figure 5.9 (page 153) contains wavelet estimates based on the same data as was used to construct figure 5.8 for the examination of kriging above. The squared errors of each surface are given in table 5.9 (page 152). It is interesting to note that there is little difference between corresponding pictures for the different values of j_t, except for small amounts of visible smoothing off the island, which is most obvious for the extremal phase wavelet with $N=2$. Comparing tables 5.8 and 5.9, notice that all six wavelet methods perform substantially better than any of the trend surfaces.

The final thing to look for in the errors for the wavelet methods is to compare tables 5.13 and 5.14 to see if there is any indication of a difference in performance when the square island is rotated away from alignment with the axes. For $\sigma_\varepsilon=0.01$ and 0.05 the restricted error is always smaller for $\nu=4$ than $\nu=3$, i.e. for the island rotated away from the axes. However, the full error measure is smaller for $\nu=3$ except for $\xi=3$, $\sigma_\varepsilon=0.05$ and $n=80$ and, strangely enough, $\sigma_\varepsilon=0.01$ and $n=40$ for all values of ξ. For $\sigma_\varepsilon=0.1$ the restricted errors are actually smaller for $\nu=3$ for $n=40$ and $\xi=2,4,5$ and the full errors are also smaller for most cases, the exceptions being $\xi=3$ for all values of n and also for $n=160$, $\xi=1$ and 4. So, in the majority of cases, the full error is less when the island is aligned with the axes, but the restricted error is greater, so that the error connected with the island is less.

The error associated with the island $(\overline{m}^{\nu\xi}(n,\sigma_\varepsilon)-\overline{r}^{\nu\xi}(n,\sigma_\varepsilon))$ is, in fact, always less for $\nu=3$. This is potentially misleading, since, as mentioned in §5.4.1, the error associated with the island with $\theta=-\frac{\pi}{8}$ is calculated over 87 grid points but the error associated with the island with $\theta=0$ is calculated over only 81 grid points. However, if the normalised island-related errors in table 5.5 are considered, it can be seen that there are only two cases where this normalised error is smaller when $\theta=-\frac{\pi}{8}$. Although this all seems to indicate that there is a significant effect of the rotation of the island, exactly the same patterns can be found in the normalised island errors for the kriging methods, so it cannot be concluded that the rotation of the island makes a difference. Since there is no reason why kriging should favour any particular orientation of island, it must be concluded that the observed effect is data related.

Comparison

Looking at table 5.4, there is a similar pattern as to where wavelets perform significantly better than kriging over the whole surface for $\nu=2,3,4$. For $n_p=2$, wavelets do better when $n=40$ for $\sigma_\varepsilon=0.01$ with $\nu=3,4$ and $\sigma_\varepsilon=0.1$ with $\nu=2,3$. However, note

Table 5.9: Squared errors for the wavelet estimates shown in figure 5.9, which are constructed using the standard wavelet method for irregularly spaced data. They are based on a simulated set of 80 points from the surface with a square island aligned with the axes, with noise level 0.05.

Wavelet	$j_t=0$	$j_t=3$
Daubechies extremal phase, $N=2$	44.97	37.90
Daubechies least asymmetric, $N=4$	44.22	38.97
Daubechies least asymmetric, $N=10$	39.75	35.95

that for the remaining cases with $n=40$ wavelets do have more successes than kriging, although not significantly so. For $n_p=4$, wavelets do significantly better when $n=40$ and $n=80$ for all three values of σ_ε and all three values of ν.

Note that for all values of n_p, n and σ_ε the number of successes that wavelets have is greater for the restricted error measurement than the full error. Going back to table 5.12, if the error over the island is computed as the difference between the $\overline{m}$ and the $\overline{r}$ then it is apparent that all kriging estimates have lower error than any of the wavelet methods over the island, except when $n_p=6$, $n=40$ and $\sigma_\varepsilon=0.05$ or $\sigma_\varepsilon=0.1$. The same trend can be seen from table 5.5 for $\nu=3$ and 4. The only exceptions are that wavelets do better than kriging with $n_p=6$ when $n=40$ for all values of σ_ε, for both $\nu=3$ and $\nu=4$. So kriging almost always captures the island better than the standard wavelet method, although this is sometimes at the expense of the performance over the smooth part of the surface.

Consider general trends in table 5.4 for $\nu=2,3,4$. For both the full and restricted error measures, and for both $n_p=2$ and $n_p=4$, for fixed σ_ε the number of successes that wavelet methods have decreases (or is at least non-increasing) as n increases, except for two exceptions, where the error for $n=80$ is less than the error when $n=160$. These both occur for $n_p=2$ and $\sigma_\varepsilon=0.1$, one for the restricted error measures with $\nu=2$ and one for the full error measures with $\nu=3$. Also, for fixed n, the number of successes is generally increasing (or at least non-decreasing) with σ_ε, although there are rather more exceptions to this trend. There are no exceptions for $\nu=2$, but for $\nu=3$ and $n=40$ with $n_p=2$ the number of successes with the full error measure is greater for $\sigma_\varepsilon=0.01$ than for $\sigma_\varepsilon=0.05$. For $n_p=4$ the number of successes associated

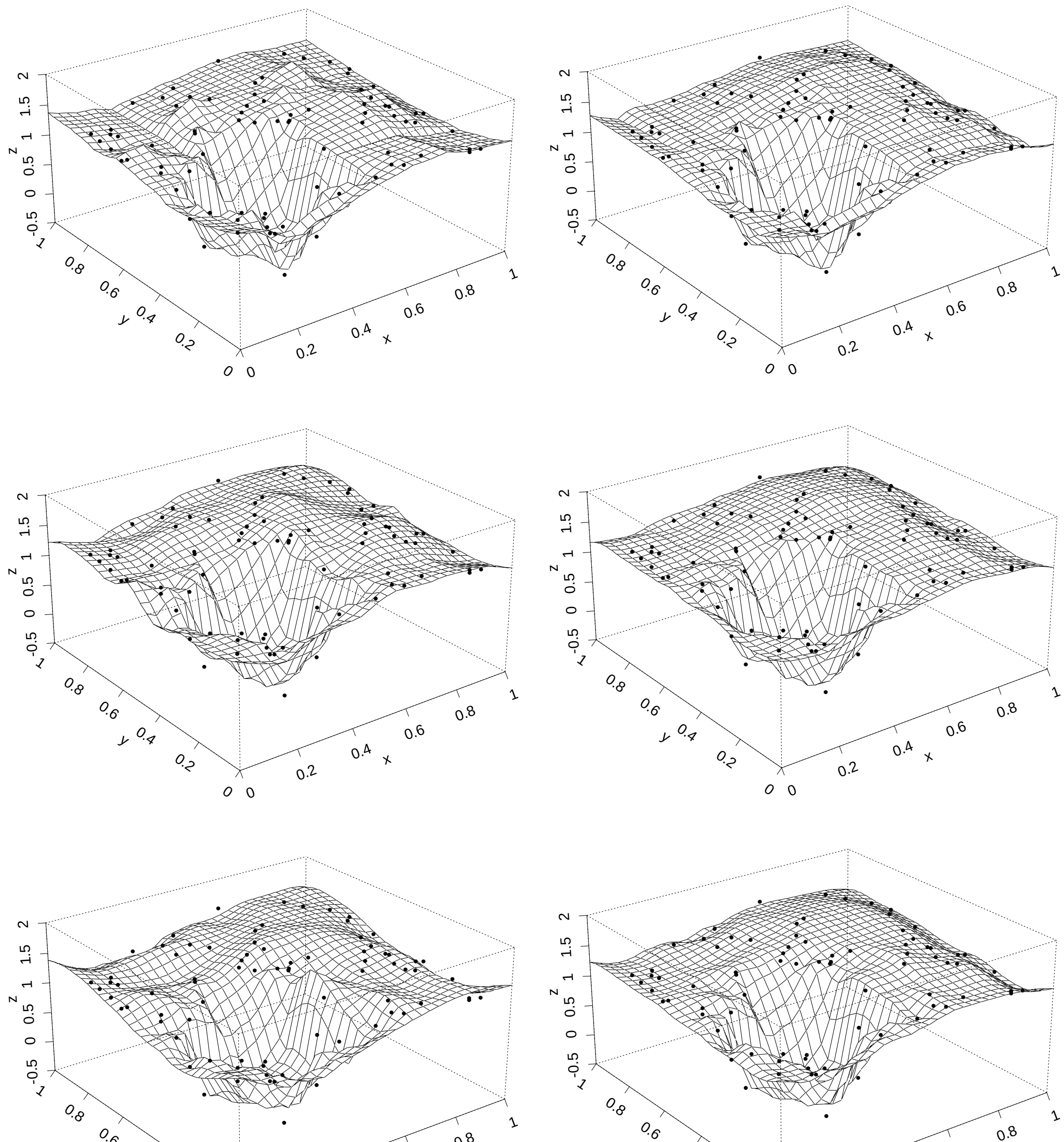


Figure 5.9: Wavelet estimates (using the standard wavelet method for irregularly spaced data) of the simulated set of 80 points from the surface with a square island aligned with the axes with noise level 0.05, as used in figure 5.8. The left-hand column contains the estimates with $j_t = 0$ and the right-hand column those with $j_t = 3$. The top row contains the estimates found using the Daubechies extremal phase wavelet on $N = 2$, the middle row those using the Daubechies least asymmetric wavelet on $N = 4$ and the bottom row those using the Daubechies least asymmetric wavelet on $N = 10$.

with both the full and restricted errors is greater for $\sigma_\varepsilon = 0.05$ than for $\sigma_\varepsilon = 0.1$. For $\nu = 4$ and $n_p = 2$, the number of successes for both the full and restricted errors with $n = 40$ are greater for $\sigma_\varepsilon = 0.01$ than for $\sigma_\varepsilon = 0.05$ (in fact, for the full errors the number of successes for $\sigma_\varepsilon = 0.01$ is greater than that for $\sigma_\varepsilon = 0.1$ as well) and also when $n = 80$ the number of successes with the full errors is greater for $\sigma_\varepsilon = 0.05$ than for $\sigma_\varepsilon = 0.1$.

What these trends show is that the relative performance of wavelet methods against kriging improves as the sample size decreases and the noise level increases.

5.4.5 Surfaces with Islands using the Special Wavelet Method for Irregularly Spaced Data - Tables 5.15 – 5.17

General Trends

The most striking feature of these tables is that the full error measures for the kriging methods are very large. This is not unexpected, however, because, as mentioned earlier, kriging receives no information at about the island, so is obviously not able to estimate it.

Now look at some general trends. These errors are, unlike those from tables 5.12 – 5.14 monotone increasing in σ_ε:

$$\left.\begin{array}{ccccc} \overline{m}^{\nu,\xi}(n,0.01) & < & \overline{m}^{\nu,\xi}(n,0.05) & < & \overline{m}^{\nu,\xi}(n,0.1) \\ \overline{r}^{\nu,\xi}(n,0.01) & < & \overline{r}^{\nu,\xi}(n,0.05) & < & \overline{r}^{\nu,\xi}(n,0.1) \end{array}\right\} \begin{array}{l} \text{for } \xi = 1,\ldots,9,\ \nu = 5,6,7 \\ \text{and } n = 40, 80, 160 \ . \end{array}$$

However, there are two exceptions to these errors being monotone decreasing in n:

$$\overline{m}^{5,8}(80,0.01) < \overline{m}^{5,8}(160,0.01) \ ,$$

$$\overline{m}^{6,8}(80,0.1) < \overline{m}^{6,8}(160,0.1) \ .$$

However, the differences are very small. Also, both of these exceptions occur for full error estimates for kriging methods and as mentioned above these have little meaning.

Wavelets

Recall from §5.2.3 that the special wavelet method inputs the information about the island after the interpolation stage, but before the wavelet thresholding. This means that the resulting surface estimate will not necessarily be exactly zero over the island and so there will be some error associated with the island. The full and restricted error measures will still, therefore, be different for the results of the special wavelet method. Looking now at the results, it is reassuring that the full and restricted error estimates are actually *very* similar, i.e. there is little error associated with the island.

As with the standard method, the special wavelet methods perform better with $j_t = 3$, although here this is always true. Looking at the ratios of the errors when $j_t = 3$ over those when $j_t = 0$, the relative improvement with $j_t = 3$ is larger for small n and large σ_ε. However, there is no longer any obvious pattern of differences between the different wavelets, for either the full or restricted errors. Note that the relative improvement is much bigger than that observed with the standard wavelet method.

The best performing wavelet method is no longer mainly $\xi = 6$. For $\nu = 5$ the best method is $\xi = 4$ except for the following cases when $n = 160$. For $\sigma_\varepsilon = 0.01$, $\xi = 6$ is the best for the full and restricted errors, whilst $\xi = 2$ is the best for the full and restricted errors with $\sigma_\varepsilon = 0.05$ and the full error only with $\sigma_\varepsilon = 0.1$. For $\nu = 6$ the best method is always the same for the full and restricted errors. It is, again, mainly $\xi = 4$ except for $n = 160$ and $\sigma_\varepsilon = 0.1$ when $\xi = 2$ performs best and $n = 80, 160$ when $\sigma_\varepsilon = 0.01$ when $\xi = 6$ is the best. Finally, for $\nu = 7$, $\xi = 4$ is generally best for $\sigma_\varepsilon = 0.01, 0.05$ except for $n = 160$, where for $\sigma_\varepsilon = 0.01$ $\xi = 6$ is better for the restricted error measure and for $\sigma_\varepsilon = 0.05$ where $\xi = 2$ is best for both the full and restricted errors. For $\sigma_\varepsilon = 0.1$, $\xi = 2$ is best except for the restricted error with $n = 40$ for which $\xi = 4$ performs marginally better. So, it seems as though the special wavelet method prefers a less smooth wavelet than the ordinary method does.

As an example, figure 5.10 (page 157) shows the special wavelet method with $\xi = 1, \ldots, 6$ in action on a simulated data set of type $\nu = 6$ of size 80 with noise level 0.1. The squared errors of these surfaces are shown in table 5.10 (page 156). One striking feature is that the least asymmetric wavelet on $N = 10$ appears to round off the corners of the square island. It may well be this sort of smoothing effect that results in the wavelets with lower filter number doing better with the special wavelet method.

Since the special wavelet method is given detailed, noise-free information about the island, it is reasonable to expect it to perform better than the standard wavelet method for irregularly spaced data. However, if tables 5.12, 5.13 and 5.14 are compared with

Table 5.10: Squared errors for the wavelet estimates shown in figure 5.10, which are constructed using the special wavelet method. They are based on a simulated set of 80 points from the surface with a square island aligned with the axes, with noise level 0.1.

Wavelet	$j_t = 0$	$j_t = 3$
Daubechies extremal phase, $N = 2$	15.02	8.12
Daubechies least asymmetric, $N = 4$	12.95	6.63
Daubechies least asymmetric, $N = 10$	14.16	8.00

tables 5.15, 5.16 and 5.17 it is clear that the errors for the special wavelet method are *much* smaller than those for the standard wavelet method. In fact, by comparison with table 5.11, the full errors for the special wavelet method, especially when $j_t = 3$, compare very favourably with the errors for the surface without the island. So the simple adaptation of the standard wavelet method for irregularly spaced data allows the island to be incorporated into the wavelet model with minimal loss of performance, compared to the estimation of a smooth surface.

It is difficult to compare the results for the two different square islands, for a number of reasons. Firstly, as mentioned above, there is very little error associated with the islands, and secondly it is not possible to look at the kriging errors over the island and so it is not possible to tell if any effect is data based, as appeared to be the case with the standard wavelet method. However, the full errors are generally less when $\theta = 0$ than when $\theta = -\frac{\pi}{8}$.

Kriging

For reasons already mentioned, the full error measures will be ignored. The restricted error measures show very similar patterns as the errors for $\nu = 1$, but note that, although they are measured relative to the same smooth surface, the restricted error measures for $\nu = 5, 6, 7$ are based on fewer points, so can be expected to be slightly smaller.

Comparison

It is obviously difficult to compare the performances of the wavelet and kriging methods over the whole surface, because the full error measures are not comparable. The pattern

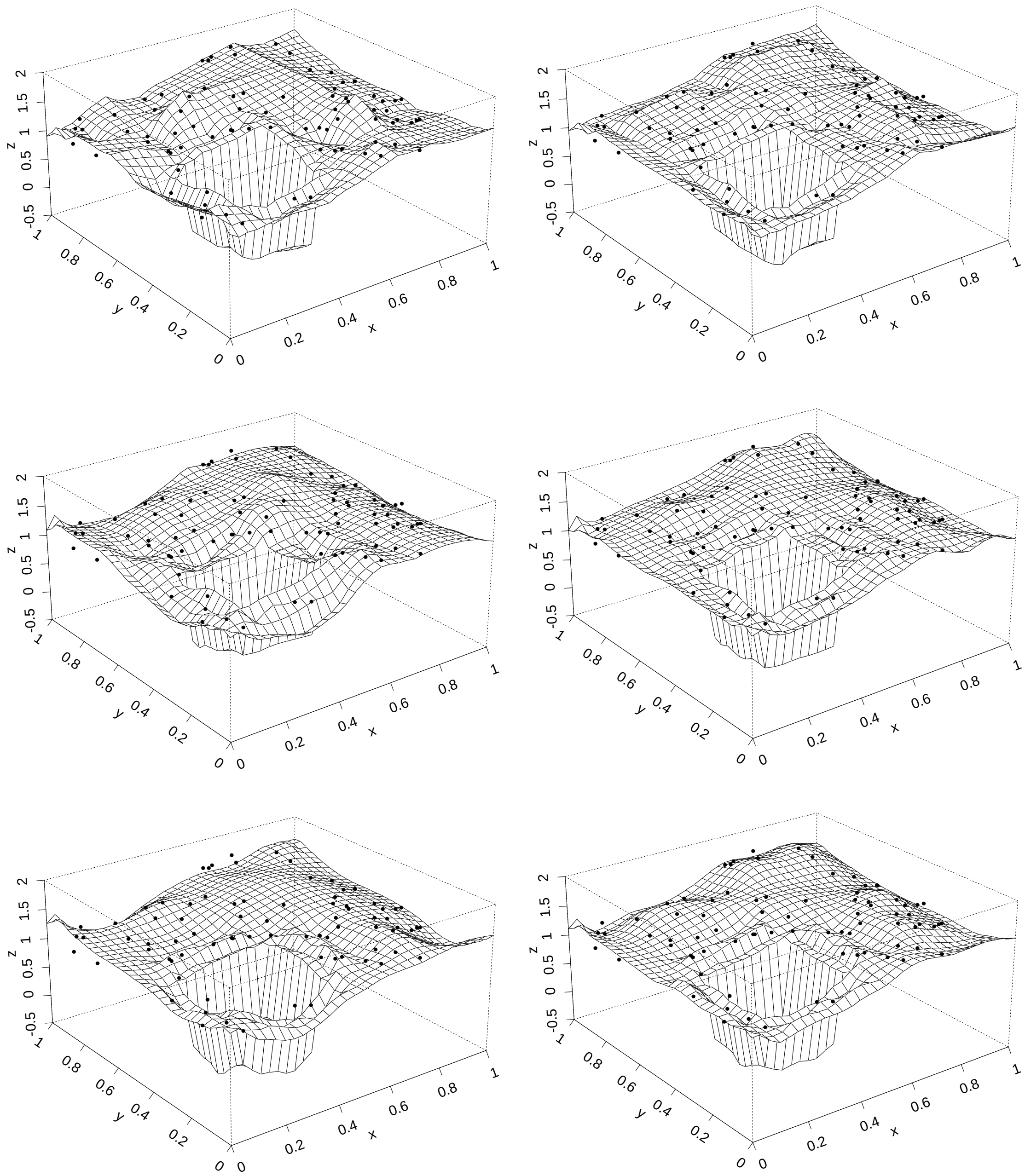


Figure 5.10: Wavelet estimates (using the special wavelet method) of a simulated set of 80 points from the surface with a square island aligned with the axes with noise level 0.1. The left-hand column contains the estimates with $j_t = 0$ and the right-hand column those with $j_t = 3$. The top row contains the estimates found using the Daubechies extremal phase wavelet on $N = 2$, the middle row those using the Daubechies least asymmetric wavelet on $N = 4$ and the bottom row those using the Daubechies least asymmetric wavelet on $N = 10$.

of success for wavelets based on the restricted errors is similar to that seen for the full errors with $\nu = 1$. Wavelets only do significantly better for some cases with the highest noise level. The important point here though is that a simple modification to the wavelet methodology for irregularly spaced data allows the wavelets to model the island with minimal increase in error.

5.4.6 Summary of Conclusions

It is not possible to draw hard and fast conclusions from a study of this nature, however, the following comments provide a brief summary of the information in the data generated.

- The wavelet methodology for irregularly spaced data outperforms kriging with $n_p = 2$ only for high noise levels and small sample sizes.

- When a discontinuous feature like the island is present, kriging methods are very sensitive to the choice of n_p, but the wavelet methods are robust to the choice of wavelet and j_t. When the island is present, wavelets outperform kriging with $n_p = 4$ except for the largest sample sizes with medium to low noise levels, and always does much better than kriging with $n_p = 6$.

- The special wavelet method, which is a simple adaptation of the standard wavelet method for irregularly spaced data that uses information about a known feature, allows a discontinuous feature, like the islands, to be included in the wavelet modelling with minimal loss of performance, compared with the estimation of a smooth surface.

Table 5.11: Simulation results for the surface without an island. Noisy data points are simulated from all parts of the surface and the standard wavelet method for irregularly spaced data in 2D is applied. The full errors, $\overline{m}^{1,\xi}(n,\sigma_\varepsilon)$, are displayed. The asterisks indicate the minimum full error over the 6 wavelet methods. For the wavelet methods, j_t is the coarsest level of coefficients to be thresholded. For the kriging methods, n_p is the degree of polynomial trend surface to be fitted. The noise level in the simulated data is σ_ε and the number of simulated points in each of the 100 samples is n.

			$\sigma_\varepsilon=0.01$			$\sigma_\varepsilon=0.05$			$\sigma_\varepsilon=0.1$		
			$n=40$	$n=80$	$n=160$	$n=40$	$n=80$	$n=160$	$n=40$	$n=80$	$n=160$
Wavelets	Wavelet	j_t									
	Daubechies Extremal Phase, $N=2$	0	6.17	2.96	1.56	11.31	5.82	3.12	21.01	13.34	8.15
		3	5.08	2.70	1.48	5.85	3.47	2.07	9.18	6.22	4.21
	Daubechies Least Asymmetric, $N=4$	0	5.86	2.61	1.42	10.97	5.13	2.64	22.95	12.96	6.84
		3	4.75	2.54	1.40	*5.25	3.07	1.84	*8.51	*5.86	3.99
	Daubechies Least Asymmetric, $N=10$	0	5.49	2.76	1.42	6.75	3.71	2.28	10.56	6.33	*3.94
		3	*4.74	*2.50	*1.35	5.35	*3.06	*1.75	8.73	5.95	4.01
Kriging	n_p										
	2		3.27	1.19	0.49	4.62	2.41	1.57	8.94	6.56	5.23
	4		0.28	0.09	0.06	7.54	2.00	1.37	23.78	8.71	5.76
	6		80.00	0.73	0.11	1585.25	10.94	2.63	5238.24	50.31	11.81

Table 5.12: Simulation results for the surface with the round island. Noisy data points are simulated from all parts of the surface and the standard wavelet method for irregularly spaced data in 2D is applied. The full errors, $\overline{m}^{2,\xi}(n,\sigma_\varepsilon)$, are displayed, with the restricted errors, $\overline{r}^{2,\xi}(n,\sigma_\varepsilon)$, given in brackets. The asterisks indicate the minimum full and restricted errors over the 6 wavelet methods. For the wavelet methods, j_t is the coarsest level of coefficients to be thresholded. For the kriging methods, n_p is the degree of polynomial trend surface that is fitted. The noise level in the simulated data is σ_ε and the number of simulated points in each of the 100 samples is n.

			$\sigma_\varepsilon=0.01$			$\sigma_\varepsilon=0.05$			$\sigma_\varepsilon=0.1$		
			$n=40$	$n=80$	$n=160$	$n=40$	$n=80$	$n=160$	$n=40$	$n=80$	$n=160$
Wavelets	Wavelet	j_t									
	Daubechies Extremal Phase, $N=2$	0	51.32 (22.35)	32.09 (15.27)	20.34 (10.46)	67.29 (29.43)	36.85 (17.27)	23.88 (12.47)	91.42 (37.89)	51.03 (25.64)	33.66 (18.85)
		3	45.34 (18.25)	30.86 (14.43)	20.14 (10.29)	53.52 (21.86)	30.36 (12.99)	21.44 (10.70)	52.49 (20.92)	33.75 (16.34)	24.38 (12.80)
	Daubechies Least Asymmetric, $N=4$	0	49.78 (21.54)	32.17 (15.30)	20.37 (10.46)	64.09 (28.33)	35.11 (16.14)	23.92 (12.39)	86.64 (36.66)	45.41 (23.34)	31.31 (17.19)
		3	44.70 (17.59)	30.66 (14.16)	20.04 (10.17)	52.68 *(21.08)	30.28 (12.70)	21.43 (10.58)	51.91 (20.28)	33.55 (16.07)	24.51 (12.71)
	Daubechies Least Asymmetric, $N=10$	0	51.51 (19.38)	32.05 (14.94)	20.31 (10.34)	62.78 (21.85)	34.67 (14.57)	23.36 (11.79)	70.74 *(18.72)	45.06 (17.90)	29.25 (14.91)
		3	*44.59 *(17.54)	*30.47 *(13.99)	*19.95 *(10.06)	*52.50 (21.08)	*29.68 *(12.48)	*21.26 *(10.46)	*51.27 (20.27)	*32.37 *(15.64)	*23.57 *(12.25)
Kriging	n_p										
	2		45.57 (21.23)	26.96 (12.74)	17.40 (8.58)	56.18 (28.28)	27.74 (12.74)	19.21 (9.69)	53.55 (26.38)	31.47 (16.54)	22.32 (12.99)
	4		152.05 (131.61)	36.97 (23.83)	18.69 (10.05)	188.78 (164.76)	45.95 (32.09)	20.03 (10.72)	237.79 (215.59)	42.83 (28.84)	23.67 (14.51)
	6		9198.59 (9171.83)	187.38 (176.75)	34.90 (26.85)	18986.88 (18921.13)	320.90 (309.35)	37.84 (29.24)	38876.81 (38844.50)	176.62 (165.08)	42.74 (34.22)

Table 5.13: Simulation results for the surface with the square island aligned with the axes ($\theta = 0$). Noisy data points are simulated from all parts of the surface and the standard wavelet method for irregularly spaced data in 2D is applied. The full errors, $\overline{m}^{3,\xi}(n, \sigma_\varepsilon)$, are displayed, with the restricted errors, $\overline{r}^{3,\xi}(n, \sigma_\varepsilon)$, given in brackets. The asterisks indicate the minimum full and restricted errors over the 6 wavelet methods. For the wavelet methods, j_t is the coarsest level of coefficients to be thresholded. For the kriging methods, n_p is the degree of polynomial trend surface that is fitted. The noise level in the simulated data is σ_ε and the number of simulated points in each of the 100 samples is n.

			$\sigma_\varepsilon = 0.01$			$\sigma_\varepsilon = 0.05$			$\sigma_\varepsilon = 0.1$		
			$n = 40$	$n = 80$	$n = 160$	$n = 40$	$n = 80$	$n = 160$	$n = 40$	$n = 80$	$n = 160$
Wavelets	Wavelet	j_t									
	Daubechies Extremal Phase, $N = 2$	0	65.38 (33.02)	36.63 (19.25)	24.65 (14.05)	68.60 (33.50)	43.51 (25.17)	28.78 (17.91)	95.59 (46.64)	58.32 (32.64)	39.62 (26.39)
		3	57.54 (28.05)	34.99 (18.10)	24.33 (13.79)	52.30 (23.93)	35.58 (19.69)	25.80 (15.50)	55.89 (27.32)	38.90 (21.10)	*28.62 (18.49)
	Daubechies Least Asymmetric, $N = 4$	0	62.89 (31.94)	37.34 (19.62)	24.79 (14.07)	64.55 (31.56)	42.02 (23.56)	29.49 (17.87)	89.12 (43.61)	53.43 (29.89)	38.63 (24.45)
		3	57.02 (27.32)	34.93 (17.83)	24.33 (13.69)	51.99 *(23.22)	36.04 (19.58)	26.33 (15.54)	55.52 (26.59)	40.09 (21.25)	30.36 (19.12)
	Daubechies Least Asymmetric, $N = 10$	0	65.03 (29.25)	36.03 (18.59)	24.45 (13.76)	64.45 (25.83)	39.13 (21.30)	27.15 (16.16)	74.20 *(25.07)	50.19 (23.52)	34.07 (20.74)
		3	*56.74 *(27.30)	*34.52 *(17.56)	*24.08 *(13.46)	*51.51 (23.22)	*35.14 *(19.14)	*25.50 *(15.02)	*54.85 (26.57)	*38.56 *(20.48)	28.77 *(17.91)
Kriging	n_p										
	2		64.54 (38.69)	32.42 (17.97)	22.19 (12.63)	56.16 (32.73)	33.46 (19.99)	23.93 (14.88)	60.32 (34.99)	37.13 (22.91)	27.96 (19.45)
	4		403.24 (382.68)	58.34 (45.24)	23.59 (14.31)	207.97 (189.69)	50.45 (38.33)	26.46 (17.67)	223.51 (204.31)	60.27 (47.41)	30.21 (21.93)
	6		27561.33 (27525.02)	348.87 (337.21)	36.20 (27.56)	32684.55 (32647.61)	207.33 (196.46)	44.09 (35.85)	13049.26 (13016.19)	292.03 (280.28)	51.69 (43.94)

Table 5.14: Simulation results for the surface with the square island rotated out of alignment with the axes ($\theta = -\frac{\pi}{8}$). Noisy data points are simulated from all parts of the surface and the standard wavelet method for irregularly spaced data in 2D is applied. The full errors, $\overline{m}^{4,\xi}(n, \sigma_\varepsilon)$, are displayed, with the restricted errors, $\overline{r}^{4,\xi}(n, \sigma_\varepsilon)$, given in brackets. The asterisks indicate the minimum full and restricted errors over the 6 wavelet methods. For the wavelet methods, j_t is the coarsest level of coefficients to be thresholded. For the kriging methods, n_p is the degree of polynomial trend surface that is fitted. The noise level in the simulated data is σ_ε and the number of simulated points in each of the 100 samples is n.

			$\sigma_\varepsilon = 0.01$			$\sigma_\varepsilon = 0.05$			$\sigma_\varepsilon = 0.1$		
			$n=40$	$n=80$	$n=160$	$n=40$	$n=80$	$n=160$	$n=40$	$n=80$	$n=160$
Wavelets	Wavelet	j_t									
	Daubechies Extremal Phase, $N=2$	0	61.61 (26.82)	38.45 (17.79)	27.19 (12.31)	77.77 (32.29)	43.87 (20.23)	29.74 (14.84)	98.20 (45.39)	58.66 (29.57)	39.24 (21.59)
		3	54.28 (21.99)	36.85 (16.56)	26.94 (12.10)	60.41 (22.43)	36.92 (15.23)	*26.83 (12.49)	59.72 (27.36)	40.35 (18.92)	*29.29 (14.36)
	Daubechies Least Asymmetric, $N=4$	0	59.12 (25.40)	38.49 (17.79)	27.24 (12.31)	72.59 (30.16)	41.65 (18.62)	29.60 (14.42)	88.26 (41.40)	52.70 (26.28)	36.25 (18.56)
		3	53.55 (21.31)	36.64 (16.26)	26.84 (11.98)	59.80 *(21.80)	36.70 (15.00)	26.84 (12.45)	58.80 (26.65)	40.10 (18.65)	29.59 (14.43)
	Daubechies Least Asymmetric, $N=10$	0	62.68 (24.29)	38.71 (17.56)	27.40 (12.36)	72.28 (24.06)	42.30 (17.72)	29.66 (14.19)	83.83 (27.37)	53.47 (22.67)	37.04 (18.01)
		3	*53.40 *(21.21)	*36.46 *(16.06)	*26.77 *(11.87)	*59.63 (21.85)	*36.49 *(14.84)	26.85 *(12.29)	*58.49 *(26.56)	*39.63 *(18.39)	29.32 *(14.20)
Kriging	n_p										
	2		59.70 (31.28)	33.97 (16.60)	23.78 (10.66)	61.63 (28.90)	35.33 (16.34)	24.07 (11.62)	63.18 (36.97)	38.56 (20.94)	27.80 (15.20)
	4		228.10 (205.05)	50.87 (34.95)	24.85 (12.01)	167.44 (140.31)	50.73 (33.24)	25.36 (13.19)	283.10 (262.13)	68.61 (52.35)	29.85 (17.53)
	6		34011.44 (33931.12)	232.52 (217.95)	38.11 (26.02)	14908.42 (14865.67)	280.44 (265.23)	37.37 (25.85)	40635.51 (40569.75)	334.75 (319.96)	57.61 (45.86)

Table 5.15: Simulation results for the surface with the round island. Noisy data points are simulated from those parts of the surface not on the island and the special wavelet method for irregularly spaced data in 2D in the presence of a known feature is applied. The full errors, $\overline{m}^{5,\xi}(n,\sigma_\varepsilon)$, are displayed, with the restricted errors, $\overline{r}^{5,\xi}(n,\sigma_\varepsilon)$, given in brackets. The asterisks indicate the minimum full and restricted errors over the 6 wavelet methods. For the wavelet methods, j_t is the coarsest level of coefficients to be thresholded. For the kriging methods, n_p is the degree of polynomial trend surface that is fitted. The noise level in the simulated data is σ_ε and the number of simulated points in each of the 100 samples is n.

			$\sigma_\varepsilon=0.01$			$\sigma_\varepsilon=0.05$			$\sigma_\varepsilon=0.1$		
			$n=40$	$n=80$	$n=160$	$n=40$	$n=80$	$n=160$	$n=40$	$n=80$	$n=160$
Wavelets	Wavelet	j_t									
	Daubechies Extremal Phase, $N=2$	0	6.38 (6.34)	3.09 (3.09)	1.52 (1.52)	11.17 (10.88)	6.33 (6.24)	3.78 (3.74)	28.94 (26.27)	15.61 (14.56)	10.98 (10.47)
		3	4.88 (4.88)	2.75 (2.75)	1.46 (1.46)	5.86 (5.85)	3.53 (3.52)	*2.38 *(2.36)	9.18 (9.10)	6.26 (6.16)	*4.82 (4.71)
	Daubechies Least Asymmetric, $N=4$	0	8.06 (7.86)	3.31 (3.30)	1.58 (1.58)	12.53 (12.06)	6.88 (6.61)	4.23 (4.11)	24.80 (23.34)	14.79 (14.02)	9.61 (8.98)
		3	*4.33 *(4.31)	*2.61 *(2.61)	1.41 (1.41)	*5.17 *(5.14)	*3.26 *(3.24)	2.40 (2.37)	*8.44 *(8.34)	*6.17 *(6.05)	4.83 *(4.71)
	Daubechies Least Asymmetric, $N=10$	0	10.01 (9.18)	4.21 (4.06)	1.98 (1.94)	13.17 (11.56)	8.24 (7.40)	5.16 (4.76)	26.84 (18.79)	15.22 (12.40)	11.44 (9.69)
		3	4.75 (4.66)	2.66 (2.64)	*1.40 *(1.39)	5.94 (5.80)	3.93 (3.83)	2.87 (2.77)	9.50 (9.20)	7.19 (6.85)	5.93 (5.57)
Kriging	n_p										
	2		100.19 (2.96)	94.99 (1.13)	92.65 (0.43)	102.42 (4.50)	96.41 (2.33)	93.73 (1.50)	107.20 (8.81)	101.09 (5.70)	96.87 (4.83)
	4		89.49 (0.23)	89.42 (0.07)	89.56 (0.05)	95.30 (5.61)	91.41 (1.91)	91.12 (1.33)	114.09 (24.14)	98.09 (7.02)	94.62 (5.14)
	6		116.54 (27.44)	89.82 (0.45)	89.61 (0.09)	1206.38 (1102.38)	104.04 (14.23)	91.98 (2.36)	4526.67 (4420.94)	132.74 (39.57)	99.04 (9.62)

Table 5.16: Simulation results for the surface with the square island aligned with the axes ($\theta = 0$). Noisy data points are simulated from those parts of the surface not on the island and the special wavelet method for irregularly spaced data in 2D in the presence of a known feature is applied. The full errors, $\overline{m}^{6,\xi}(n, \sigma_\varepsilon)$, are displayed, with the restricted errors, $\overline{r}^{6,\xi}(n, \sigma_\varepsilon)$, given in brackets. The asterisks indicate the minimum full and restricted errors over the 6 wavelet methods. For the wavelet methods, j_t is the coarsest level of coefficients to be thresholded. For the kriging methods, n_p is the degree of polynomial trend surface that is fitted. The noise level in the simulated data is σ_ε and the number of simulated points in each of the 100 samples is n.

			$\sigma_\varepsilon = 0.01$			$\sigma_\varepsilon = 0.05$			$\sigma_\varepsilon = 0.1$		
			$n=40$	$n=80$	$n=160$	$n=40$	$n=80$	$n=160$	$n=40$	$n=80$	$n=160$
Wavelets	Wavelet	j_t									
	Daubechies Extremal Phase, $N=2$	0	6.03 (6.02)	2.65 (2.64)	1.52 (1.52)	10.36 (10.15)	5.85 (5.79)	3.45 (3.42)	25.33 (23.46)	13.67 (13.17)	9.92 (9.67)
		3	4.74 (4.74)	2.38 (2.38)	1.46 (1.46)	5.68 (5.66)	3.37 (3.36)	2.28 (2.26)	9.21 (9.13)	6.18 (6.11)	*4.54 *(4.47)
	Daubechies Least Asymmetric, $N=4$	0	6.95 (6.87)	2.83 (2.82)	1.55 (1.54)	12.50 (11.96)	6.38 (6.20)	3.67 (3.59)	26.13 (23.20)	14.97 (13.71)	10.00 (9.30)
		3	*4.29 *(4.28)	2.31 (2.31)	1.42 (1.42)	*5.06 *(5.04)	*3.14 *(3.12)	*2.27 *(2.25)	*8.80 *(8.72)	*6.17 *(6.08)	4.86 (4.73)
	Daubechies Least Asymmetric, $N=10$	0	8.13 (7.70)	3.48 (3.38)	1.89 (1.84)	11.34 (10.46)	7.32 (6.70)	4.52 (4.21)	21.91 (17.26)	13.51 (11.71)	9.91 (8.53)
		3	4.47 (4.44)	*2.25 *(2.23)	*1.40 *(1.39)	5.60 (5.55)	3.58 (3.52)	2.58 (2.51)	9.40 (9.29)	6.76 (6.61)	5.36 (5.13)
Kriging	n_p										
	2		120.03 (3.17)	113.22 (1.11)	109.94 (0.40)	120.35 (4.35)	113.67 (2.22)	111.12 (1.47)	126.21 (8.51)	117.06 (5.68)	115.67 (4.64)
	4		106.09 (0.23)	105.93 (0.07)	105.78 (0.05)	111.77 (6.78)	107.11 (1.80)	107.04 (1.29)	127.81 (20.55)	112.03 (7.72)	112.34 (4.94)
	6		144.98 (37.86)	106.35 (0.47)	105.88 (0.09)	1246.02 (1130.90)	116.73 (12.38)	108.70 (2.40)	4371.88 (4210.33)	163.01 (60.42)	116.80 (8.96)

Table 5.17: Simulation results for the surface with the square island rotated out of alignment with the axes ($\theta = -\frac{\pi}{8}$). Noisy data points are simulated from those parts of the surface not on the island and the special wavelet method for irregularly spaced data in 2D in the presence of a known feature is applied. The full errors, $\overline{m}^{7,\xi}(n, \sigma_\varepsilon)$, are displayed, with the restricted errors, $\overline{r}^{7,\xi}(n, \sigma_\varepsilon)$, given in brackets. The asterisks indicate the minimum full and restricted errors over the 6 wavelet methods. For the wavelet methods, j_t is the coarsest level of coefficients to be thresholded. For the kriging methods, n_p is the degree of polynomial trend surface that is fitted. The noise level in the simulated data is σ_ε and the number of simulated points in each of the 100 samples is n.

			$\sigma_\varepsilon = 0.01$			$\sigma_\varepsilon = 0.05$			$\sigma_\varepsilon = 0.1$		
			$n = 40$	$n = 80$	$n = 160$	$n = 40$	$n = 80$	$n = 160$	$n = 40$	$n = 80$	$n = 160$
Wavelets	Wavelet	j_t									
	Daubechies Extremal Phase, $N = 2$	0	6.87 (6.82)	3.14 (3.12)	1.69 (1.69)	11.19 (10.81)	6.58 (6.47)	3.86 (3.80)	26.43 (24.40)	15.08 (14.15)	10.54 (10.04)
		3	4.61 (4.61)	2.59 (2.59)	1.58 (1.58)	5.62 (5.59)	3.23 (3.20)	*2.22 *(2.20)	*9.26 (9.19)	*6.39 *(6.29)	*4.41 *(4.29)
	Daubechies Least Asymmetric, $N = 4$	0	8.69 (8.42)	3.44 (3.42)	1.76 (1.75)	14.64 (13.76)	7.39 (7.04)	4.00 (3.88)	25.85 (23.53)	16.72 (15.21)	10.93 (9.90)
		3	*4.01 *(4.00)	*2.46 *(2.46)	*1.53 (1.53)	*5.22 *(5.16)	*3.14 *(3.10)	2.35 (2.31)	9.29 *(9.13)	6.72 (6.55)	5.12 (4.91)
	Daubechies Least Asymmetric, $N = 10$	0	11.98 (10.74)	5.58 (5.17)	2.38 (2.32)	16.98 (14.30)	9.80 (8.71)	6.04 (5.52)	34.03 (22.60)	19.08 (15.41)	13.59 (11.50)
		3	4.51 (4.44)	2.52 (2.49)	1.54 *(1.52)	5.97 (5.84)	3.61 (3.53)	2.69 (2.62)	10.19 (9.87)	7.64 (7.31)	5.96 (5.61)
Kriging	n_p										
	2		129.72 (3.00)	123.57 (1.13)	119.92 (0.42)	131.15 (4.23)	124.55 (2.30)	120.96 (1.46)	137.50 (8.76)	129.05 (5.71)	124.19 (4.67)
	4		116.23 (0.26)	115.91 (0.07)	115.66 (0.05)	122.48 (5.89)	117.66 (1.85)	116.88 (1.28)	152.22 (35.22)	123.61 (7.07)	120.67 (5.15)
	6		228.83 (109.05)	116.47 (0.43)	115.62 (0.10)	1792.75 (1653.46)	127.61 (12.11)	118.10 (2.37)	15561.85 (15304.80)	156.93 (38.64)	125.15 (9.28)

Bibliography

Abramovich, F., Sapatinas, T., & Silverman, B.W. 1998. Wavelet Thresholding via a Bayesian Approach. *Journal of The Royal Statistical Society Series B*, **60**(4), 725–749.

Abramovich, F., Sapatinas, T., & Silverman, B.W. 2000a. Stochastic Expansions in an Overcomplete Wavelet Dictionary. *Probability Theory and Related Fields*. To Appear.

Abramovich, F., Bailey, T.C., & Sapatinas, T. 2000b. Wavelet Analysis and its Statistical Applications. *The Statistician*, **49**, 1–29.

Aitken, C.G.G. 1995. *Statistics and the Evaluation of Evidence for Forensic Scientists.* Chichester: Wiley.

Akima, H. 1978a. Algorithm 526. Bivariate Interpolation and Smooth Surface Fitting for Irregularly Distributed Data Points. *ACM Transactions on Mathematical Software*, **4**(2), 160–164.

Akima, H. 1978b. A Method of Bivariate Interpolation and Smooth Surface Fitting for Irregularly Distributed Data Points. *ACM Transactions on Mathematical Software*, **4**(2), 148–159.

Antoniadis, A. 2000. Wavelets in Statistics: A Review (with discussion). *Journal of the Italian Statistical Society*. To Appear.

Bean, S.J., & Tsokos, C.P. 1980. Developments in Nonparametric Density Estimation. *International Statistical Review*, **48**, 267–287.

Bengtsson, B-E., & Nordbeck, S. 1964. Construction of Isarithms and Isarithmic Maps by Computers. *BIT*, **4**, 87–105.

Beylkin, G. 1992. On the Representation of Operators in Bases of Compactly Supported Wavelets. *SIAM Journal on Numerical Analysis*, **29**(6), 1716–1740.

Brislawn, C.M. 1996. The FBI Fingerprint Image Compression Standard. *http://www.c3.lanl.gov/~brislawn/FBI/FBI.html*.

Čencov, N.N. 1962. Evaluation of an Unknown Distribution Density from Observations. *Soviet Mathematics - Doklady*, **3**, 1559–1562.

Clutton-Brock, M. 1990. Density Estimation Using Exponentials of Orthogonal Series. *Journal of the American Statistical Association*, **85**(411), 760–764.

Daubechies, I. 1988. Orthonormal Bases of Compactly Supported Wavelets. *Communications on Pure and Applied Mathematics*, **41**, 909–996.

Daubechies, I. 1992. *Ten Lectures on Wavelets*. Philadelphia: Society for Industrial and Applied Mathematics.

Daubechies, I., & Lagarias, J.C. 1992. Two-Scale Difference Equations II. Local Regularity, Infinite Products of Matrices and Fractals. *SIAM Journal on Mathematical Analysis*, **23**(4), 1031–1079.

Daubechies, I., Guskov, I., Schröder, P., & Sweldens, W. 1999. Wavelets on Irregular Point Sets. *Philosophical Transactions of the Royal Society Series A*, **357**(1760), 2397–2413.

Davis, J.C. 1973. *Statistics and Data Analysis in Geology*. New York: Wiley.

Delyon, B., & Juditski, A. 1995. Estimating Wavelet Coefficients. *Pages 151–168 of:* Antoniadis, A., & Oppenheim, G. (eds), *Wavelets and Statistics*. Lecture Notes in Statistics, no. 103. New York: Springer-Verlag.

Diggle, P.J. 1990. *Time Series: a Biostatistical Introduction*. Oxford: Oxford University Press.

Diggle, P.J., & Hall, P. 1986. The Selection of Terms in an Orthogonal Series Density Estimator. *Journal of the American Statistical Association*, **81**(393), 230–233.

Donoho, D.L., & Johnstone, I.M. 1994. Ideal Spatial Adaptation by Wavelet Shrinkage. *Biometrika*, **81**(3), 425–455.

Donoho, D.L., & Johnstone, I.M. 1995. Adapting to Unknown Smoothness via Wavelet Shrinkage. *Journal of the American Statistical Association*, **90**(432), 1200–1224.

Donoho, D.L., Johnstone, I.M., Kerkyacharian, G., & Picard, D. 1995. Wavelet Shrinkage: Asymptopia? *Journal of The Royal Statistical Society Series B*, **57**(2), 301–369.

Donoho, D.L., Johnstone, I.M., Kerkyacharian, G., & Picard, D. 1996. Density Estimation by Wavelet Thresholding. *The Annals of Statistics*, **24**(2), 508–539.

Engel, J. 1990. Density Estimation with Haar Series. *Statistics & Probability Letters*, **9**, 111–117.

Goel, P., & Vidakovic, B. 1995. *Wavelet Transformations as Diversity Enhancers.* Discussion Paper 95-04. ISDS, Duke University.

Good, I.J., & Gaskins, R.A. 1971. Nonparametric Roughness Penalties for Probability Densities. *Biometrika*, **58**(2), 255–277.

Green, P.J., & Sibson, R. 1978. Computing Dirichlet Tesselations in the Plane. *The Computer Journal*, **21**(2), 168–173.

Hall, P. 1983. Orthogonal Series Methods for both Qualitative and Quantitative Data. *The Annals of Statistics*, **11**(3), 1004–1007.

Hall, P., & Nason, G.P. 1997. On Choosing a Non-Integer Resolution Level when using Wavelet Methods. *Statistics & Probability Letters*, **34**(1), 5–11.

Hall, P., & Patil, P. 1995. Formulae for Mean Integrated Squared Error of Nonlinear Wavelet-Based Density Estimators. *The Annals of Statistics*, **23**(3), 905–928.

Izenman, A.J. 1991. Recent Developments in Nonparametric Density Estimation. *Journal of the American Statistical Association*, **86**(413), 205–224.

Jawerth, B., & Sweldens, W. 1994. An Overview of Wavelet Based Multiresolution Analyses. *SIAM Review*, **36**(3), 377–412.

Johnstone, I.M., & Silverman, B.W. 1997. Wavelet Threshold Estimators for Data with Correlated Noise. *Journal of The Royal Statistical Society Series B*, **59**(2), 319–351.

Koo, J-Y., & Kim, W-C. 1996. Wavelet Density Estimation by Approximation of Log-Densities. *Statistics & Probability Letters*, **26**, 271–278.

Kovac, A., & Silverman, B.W. 2000. Extending the Scope of Wavelet Regression Methods by Coefficient-Dependent Thresholding. *Journal of the American Statistical Association*, **95**. To Appear.

Kronmal, R., & Tarter, M. 1968. The Estimation of Probability Densities and Cumulatives by Fourier Series Methods. *Journal of the American Statistical Association*, **63**, 925–952.

Lawson, C.L. 1977. Software for C^1 Surface Interpolation. *Pages 161–194 of:* Rice, J.R. (ed), *Mathematical Software III*. Academic Press.

Lepski, O.V., Mammen, E., & Spokoiny, V.G. 1997. Optimal Spatial Adaptation to Inhomogeneous Smoothness: An Approach Based on Kernel Estimates with Variable Bandwidth Selectors. *The Annals of Statistics*, **25**(3), 929–947.

Lloyd, E.L. 1977. On Triangulations of a Set of Points in the Plane. *Proceedings of the* 18^{th} *Annual IEEE Conference on the Foundations of Computer Science*, 228–240.

Mallat, S.G. 1989a. Multiresolution Approximations and Wavelet Orthonormal Bases of $L^2(\mathbb{R})$. *Transactions of the American Mathematical Society*, **315**(1), 69–87.

Mallat, S.G. 1989b. A Theory for Multiresolution Signal Decomposition: The Wavelet Representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**(7), 674–693.

Marron, J.S., & Wand, M.P. 1992. Exact Mean Integrated Squared Error. *The Annals of Statistics*, **20**(2), 712–736.

Masry, E. 1994. Probability Density Estimation from Dependent Observations using Wavelets Orthonormal Bases. *Statistics & Probability Letters*, **21**(3), 181–194.

Masry, E. 1997. Multivariate Probability Density Estimation by Wavelet Methods: Strong Consistency and Rates for Stationary Time Series. *Stochastic Processes and their Applications*, **67**(2), 177–193.

McCullagh, M.J., & Ross, C.G. 1980. Delaunay Triangulation of a Random Data Set for Isarithmic Mapping. *The Cartographic Journal*, **17**(2), 93–99.

McLain, D.H. 1976. Two Dimensional Interpolation from Random Data. *The Computer Journal*, **19**(2), 178–181.

Meyer, Y. 1992. *Wavelets and Operators.* Cambridge: Cambridge University Press.

Morlet, J., Arens, G., Fourgeau, E., & Giard, D. 1982. Wave Propagation and Sampling Theory. *Geophysics*, **47**(2), 203–236.

Nason, G.P. 1994. *Wavelet Regression by Cross-Validation.* Technical Report 447. Stanford University.

Nason, G.P. 1996. Wavelet Shrinkage using Cross-Validation. *Journal of The Royal Statistical Society Series B*, **58**(2), 463–479.

Nason, G.P., & Silverman, B.W. 1994. The Discrete Wavelet Transform in S. *Journal of Computational and Graphical Statistics*, **3**, 163–191.

Nason, G.P., & Silverman, B.W. 1995. The Stationary Wavelet Transform and some Statistical Applications. *Pages 281–300 of:* Antoniadis, A., & Oppenheim, G. (eds), *Wavelets and Statistics.* Lecture Notes in Statistics, no. 103. New York: Springer-Verlag.

Okabe, A., Boots, B., & Sugihara, K. 1992. *Spatial Tessellations - Concepts and Applications of Voronoi Diagrams.* Chichester: Wiley.

Park, B.U., & Marron, J.S. 1990. Comparison of Data-Driven Bandwidth Selectors. *Journal of the American Statistical Association*, **85**, 66–72.

Penev, S., & Dechevsky, L. 1997. On Non-Negative Wavelet-Based Density Estimators. *Journal of Nonparametric Statistics*, **7**, 365–394.

Pensky, M., & Vidakovic, B. 1998. *On Non-Equally Spaced Wavelet Regression.* Discussion Paper 98-06. ISDS, Duke University.

Pinheiro, A., & Vidakovic, B. 1995. *Estimating the Square Root of a Density via Compactly Supported Wavelets.* Discussion Paper 95-14. ISDS, Duke University.

Priestley, M.B. 1981. *Spectral Analysis and Time Series.* Probability and Mathematical Statistics. London: Academic Press.

Rafajlowicz, E. 1997. Consistency of Orthogonal Series Density Estimators Based on Grouped Observations. *IEEE Transactions on Information Theory*, **43**(1), 283–285.

Ripley, B.D. 1981. *Spatial Statistics.* New York: Wiley.

Roeder, K. 1994. DNA Fingerprinting: A Review of the Controversy. *Statistical Science*, **9**(2), 222–278.

Sabin, M.A. 1976. Errata. *The Computer Journal*, **19**(4), 384.

Safavi, A.A., Chen, J., & Romagnoli, J.A. 1997. Wavelet-Based Density Estimation and Application to Process Monitoring. *AIChE Journal*, **43**(5), 1227–1241.

Sardy, S., Percival, D.B., Bruce, A.G., Gao, H-Y., & Stuetzle, W. 1999. Wavelet Shrinkage for Unequally Spaced Data. *Statistics and Computing*, **9**(1), 65–75.

Schomer, D.F., Elekes, A.A., Hazle, J.D., Huffman, J.C., Thompson, S.K., Chui, C.K., & Murphy, W.A. 1998. Introduction to Wavelet-Based Compression of Medical Images. *Radiographics*, **18**(2), 469–481.

Schwartz, S.C. 1967. Estimation of Probability Density by an Orthogonal Series. *The Annals of Mathematical Statistics*, **38**, 1261–1265.

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

Sheather, S.J., & Jones, M.C. 1991. A Reliable Data-Based Bandwidth Selection Method for Kernel Density Estimation. *Journal of The Royal Statistical Society Series B*, **53**(3), 683–690.

Sibson, R. 1978. Locally Equiangular Triangulations. *The Computer Journal*, **21**(3), 243–245.

Sibson, R. 1980a. A Brief Description of Natural Neighbour Interpolation. *Pages 21–36 of:* Barnett, V. (ed), *Interpreting Multivariate Data*. Wiley.

Sibson, R. 1980b. A Vector Identity for the Dirichlet Tessellation. *Mathematical Proceedings of the Cambridge Philosophical Society*, **87**, 151–155.

Silverman, B.W. 1986. *Density Estimation for Statistics and Data Analysis*. Monographs on Statistics and Applied Probability. London: Chapman and Hall.

Simonoff, J.S. 1996. *Smoothing Methods in Statistics*. Springer Series in Statistics. New York: Springer-Verlag.

Smith, M.J.T., & Barnwell, T.P. 1986. Exact Reconstruction Techniques for Tree-Structured Subband Coders. *IEEE Transactions on Acoustics, Speech and Signal Processing*, **34**(3), 434–441.

Smith, M.J.T., & Eddins, S.L. 1990. Analysis/Synthesis Techniques for Subband Image Coding. *IEEE Transactions on Acoustics, Speech and Signal Processing*, **38**(8), 1446–1456.

Stein, C.M. 1981. Estimation of the Mean of a Multivariate Normal Distribution. *The Annals of Statistics*, **9**(6), 1135–1151.

Sweldens, W. 1996. Wavelets and the Lifting Scheme: A 5 Minute Tour. *Zeitschrift für Angewandte Mathematik und Mechanik*, **76**(Supplement 2), 41–44.

Sweldens, W. 1998. The Lifting Scheme: A Construction of Second Generation Wavelets. *SIAM Journal on Mathematical Analysis*, **29**(2), 511–546.

Tarter, M.E., & Kronmal, R.A. 1976. An Introduction to the Implementation and Theory of Nonparametric Density Estimation. *The American Statistician*, **30**(3), 105–112.

Tarter, M.E., & Lock, M.D. 1993. *Model-Free Curve Estimation*. Monographs on Statistics and Applied Probability. New York: Chapman and Hall.

Terrell, G.R., & Scott, D.W. 1992. Variable Kernel Density Estimation. *The Annals of Statistics*, **20**(3), 1236–1265.

Triebel, H. 1992. *Theory of Function Spaces II.* Basel: Birkhäuser.

Vaidyanathan, P.P. 1990. Multirate Digital Filters, Filter Banks, Polyphase Networks, and Applications: a Tutorial. *Proceedings of the IEEE*, **78**, 56–93.

Vannucci, M. 1995. *Nonparametric Density Estimation using Wavelets.* Discussion Paper 95-26. ISDS, Duke University.

Vannucci, M., & Vidakovic, B. 1995. *Preventing the Dirac Disaster: Wavelet Based Density Estimation.* Discussion Paper 95-27. ISDS, Duke University.

Venables, W.N., & Ripley, B.D. 1994. *Modern Applied Statistics with S-Plus.* Statistics and Computing. New York: Springer-Verlag.

Vidakovic, B. 1999. *Statistical Modeling by Wavelets.* New York: Wiley.

Wand, M.P., & Jones, M.C. 1995. *Kernel Smoothing.* Monographs on Statistics and Applied Probability. London: Chapman and Hall.

Watson, D.F., & Philip, G.M. 1984. Systematic Triangulations. *Computer Vision, Graphics, and Image Processing*, **26**, 217–223.

Watson, G.S. 1969. Density Estimation by Orthogonal Series. *The Annals of Mathematical Statistics*, **40**(4), 1496–1498.