Unbalanced Haar technique for nonparametric function estimation

Piotr Fryzlewicz*

May 23, 2007

Abstract

The discrete Unbalanced Haar (UH) transform is a decomposition of one-dimensional data with respect to an orthonormal Haar-like basis where jumps in the basis vectors do not necessarily occur in the middle of their support. We introduce a multiscale procedure for estimation in Gaussian noise which consists of three steps: a UH transform, thresholding of the decomposition coefficients, and the inverse UH transform. We show that our estimator is mean-square consistent with near-optimal rates for a wide range of functions, uniformly over UH bases which are not "too unbalanced". A vital ingredient of our approach is basis selection. We choose each basis vector so that it best matches the data at a specific scale and location, where the latter parameters are determined by the "parent" basis vector. Our estimator is computable in $O(n \log n)$ operations.

A simulation study demonstrates the good performance of our estimator in comparison with state-of-the-art competitors. We apply our method to the estimation of the mean intensity of the time series of earthquake counts occurring in Northern California. We discuss extensions to image data, and to smoother wavelets.

Keywords: adaptive smooting, CART, binary segmentation, matching pursuit, piecewise-constant estimators, wavelets.

1 Introduction

A fundamental problem in non-parametric regression is the estimation of a one-dimensional function $f : [0,1] \mapsto \mathbb{R}$ from noisy measurements X_i observed on an equispaced grid:

$$X_i = f(i/n) + \varepsilon_i, \quad i = 1, \ldots, n, \tag{1}$$

*Department of Mathematics, University of Bristol, University Walk, Bristol BS8 1TW, UK; email: p.z.fryzlewicz@bristol.ac.uk.

where ε_i's are random variables with $\mathbb{E}(\varepsilon_i) = 0$. Various subclasses of the problem can be identified, depending on the joint distribution of $(\varepsilon_i)_{i=1}^n$ and on the smoothness of f. In particular, substantial research effort has been and is being expended on developing denoising techniques under the assumption that $(\varepsilon_i)_{i=1}^n \sim N(\underline{0}, \sigma^2\mathbf{I})$. In this paper, we also investigate the iid Gaussian case. We note that the simple 1D Gaussian iid model serves as a useful proxy for many other settings (such as various time series contexts, regression problems with other noise distributions, or regression problems in more than one dimension) in the sense that the success of an estimation methodology in the Gaussian model is often an informative test of its potential usefulness in those other, typically more challenging, contexts.

The idea of estimating f by means of piecewise constant estimators received much attention in the recent literature. The reasons for this are at least threefold. Firstly, the class of piecewise constant functions is "flexible" in the sense that it is capable of approximating a wide range of function spaces well, see e.g. DeVore (1998). Secondly, piecewise constant estimates are cheap to store as the number of jumps is typically significantly less than n. Thirdly, they are easy to interpret: jumps in the estimate can be viewed as "significant" changes in the mean level of the data, whereas constant intervals represent periods where the mean of the data remains "approximately" the same.

Of particular interest to us are non-linear estimators, which are well known to offer superior theoretical and practical performance to linear estimators when the underlying function f is spatially inhomogeneous. Two recent examples of non-linear methods which produce piecewise constant reconstructions are those by Polzehl and Spokoiny (2000) and Davies and Kovac (2001). The former one uses local averaging where the local neighbourhood is chosen in a data-driven way, while the principle behind the latter one is to control the number of local extremes.

Wavelet thresholding estimators, first proposed in a seminal work by Donoho and Johnstone (1994), perform well, and, for some threshold choices, are tractable theoretically. Their use with Haar wavelets yields piecewise constant estimates. The CART methodology of Breiman et al. (1983) is based on the idea of adaptive recursive partitioning and produces a piecewise constant reconstruction where the pieces are terminal nodes of the partition. Donoho (1997) shows the equivalence of dyadic CART and a particular type of nonlinear Haar wavelet estimation. An advantage of Haar thresholding estimators is their extremely rapid computability (provided that the threshold choice is fast). One disadvantage is that due to their construction, jumps always occur at dyadic locations, even if it is not justified by the data. By the above equivalence, the same criticism applies to the dyadic CART methodology. The full non-dyadic CART avoids this restriction but its theoretical properties are still not well understood, although Gey and Nedelec (2005) provide some partial consistency results.

Girardi and Sweldens (1997) introduce the *unbalanced Haar* wavelet basis where, unlike traditional Haar wavelets, jumps in the basis functions do not necessarily occur in the middle of their support. As such, they are potentially useful as building

blocks for piecewise constant estimators which avoid the restriction of jumps occuring at dyadic locations. Indeed, Kolaczyk and Nowak (2004) mention in passing that their complexity-penalised recursive partitioning estimator is "linked" to unbalanced Haar wavelets, as it is piecewise constant, multiscale in nature and avoids the dyadic restriction. They proceed to demonstrate its good theoretical risk properties and show that its computational complexity is of order $O(n^3)$.

In this paper, we explore more fully the possibility of using unbalanced Haar wavelets to construct piecewise constant estimators which avoid the dyadic restriction. We propose an unbalanced Haar wavelet thresholding estimator of f with respect to any choice of the unbalanced Haar basis, and show its mean-square consistency over a large class of function spaces. We show that the consistency holds in a uniform sense over the family of unbalanced Haar bases, provided that they are not "too unbalanced". We propose a computational procedure for choosing a suitable basis, and discuss similarities and differences between our method and the binary segmentation scheme of Venkatraman (1993) in the special case of piecewise constant target functions. The final estimator is mean-square consistent for a large class of functions, and its computational complexity is of order $O(n \log n)$. We demonstrate its very good finite-sample performance in a comparative simulation study. We apply our method to the estimation of the mean intensity of the time series of earthquake counts occurring in Northern California. Our study appears to confirm the previous observation that seismicity rates increase after major earthquakes in sites which are located not necessarily close to the examined area.

The paper concludes with a brief discussion of the extension of the UH technique to image data, and to other smoother unbalanced wavelet bases. As a prelude to this discussion, we introduce the so-called bottom-up UH transform, which serves as a starting point for both of these extensions.

We also note that Neumann (1996) shows how to adapt a standard "balanced" wavelet technique for iid Gaussian data to the (very general) set-up where the noise is a locally stationary time series. Although our technique is in many ways different to classical balanced wavelets, the fact that it is derived from Haar wavelets means that a similar route to that shown by Neumann (1996) is likely to be successful in extending it to this setting.

The main algorithm of the paper has been implemented in the R package `unbalhaar`, available from `http://cran.r-project.org/`.

2 Unbalanced Haar wavelets

Traditional wavelet thresholding estimation (Donoho and Johnstone, 1994) proceeds as follows: take the discrete wavelet transform of $\{X_i\}_{i=1}^n$, set to zero those coefficients which fall below a certain threshold, and then take the inverse wavelet transform of the thresholded coefficients to yield an estimate of f. Estimates of this type have

been studied by several authors: see Vidakovic (1999) for an overview.

Our estimation procedure can be summarised as follows: instead of the traditional wavelet transform, we first take a transform of the data with respect to an Unbalanced Haar (UH) basis. We then threshold the coefficients, and take the inverse transform to obtain an estimate of f. An important ingredient of our approach is basis selection. We will discuss all the ingredients in turn. This section sets the scene by describing the UH vectors and the discrete UH transform.

UH wavelets were introduced by Girardi and Sweldens (1997) and applied in a non-parametric stochastic regression context by Delouille et al. (2001). In their work, the "unbalancedness" was introduced to handle the fact that the design was stochastic and thus nonequispaced: hence the need for basis functions with different support lengths and jump locations. By contrast, we consider the case of fixed equidistant design, and introduce the unbalancedness to capture important features of the data X_i, as opposed to the design. Below, we introduce the UH vectors and the discrete UH transform for equispaced data.

2.1 Unbalanced Haar vectors

We first give a description of the construction of the UH vectors. Suppose that our domain is indexed by $i = 1, \ldots, n$, as is the case in (1), and that $n \geq 2$. We first construct a vector $\psi^{0,1}$, which is constant and positive for $i = 1, \ldots, b^{0,1}$, and constant and negative for $i = b^{0,1} + 1, \ldots, n$. The breakpoint $b^{0,1} < n$ is to be chosen by the analyst. The positive and negative values taken by $\psi^{0,1}$ are chosen in such a way that (a) the elements of $\psi^{0,1}$ sum to zero, and (b) the squared elements of $\psi^{0,1}$ sum to one.

We then recursively repeat this construction on the two parts of the domain determined by $\psi^{0,1}$: that is, provided that $b^{0,1} \geq 2$, we construct (in a similar fashion) a vector $\psi^{1,1}$ supported on $i = 1, \ldots, b^{0,1}$, with a breakpoint $b^{1,1}$. Also, provided that $n - b^{0,1} \geq 2$, we construct a vector $\psi^{1,2}$ supported on $i = b^{0,1} + 1, \ldots, n$ with a breakpoint $b^{1,2}$. The recursion then continues in the same manner for as long as feasible, with each vector $\psi^{j,k}$ having at most two "children" vectors $\psi^{j+1,2k-1}$ and $\psi^{j+1,2k}$. For each vector $\psi^{j,k}$, their start, breakpoint and end indices are denoted by $s^{j,k}$, $b^{j,k}$ and $e^{j,k}$, respectively. Additionally, we define a vector $\psi^{-1,1}$ with elements $\psi^{-1,1}(l) = n^{-1/2}\mathbb{I}(1 \leq l \leq n)$, where $\mathbb{I}(\cdot)$ is the indicator function. Note that to shorten notation, we do not explicitly emphasise the dependence of $\psi^{j,k}$ on $(s^{j,k}, b^{j,k}, e^{j,k})$. The indices j, k are scale and location parameters, respectively. Small (large) values of j can be thought of as corresponding to "coarse" ("fine") scales, like in the classical wavelet theory, see e.g. Mallat (1989b).

Example. We consider an example of a set of UH vectors for $n = 6$. The rows of the matrix W defined below contain (from top to bottom) vectors $\psi^{-1,1}$, $\psi^{0,1}$, $\psi^{1,2}$, $\psi^{2,3}$, $\psi^{2,4}$ and $\psi^{3,7}$ determined by the following set of breakpoints: $(b^{0,1}, b^{1,2}, b^{2,3}, b^{2,4}, b^{3,7}) =$

$(1, 3, 2, 5, 4)$.

$$W = \begin{pmatrix} 6^{-1/2} & 6^{-1/2} & 6^{-1/2} & 6^{-1/2} & 6^{-1/2} & 6^{-1/2} \\ \{5/6\}^{1/2} & -30^{-1/2} & -30^{-1/2} & -30^{-1/2} & -30^{-1/2} & -30^{-1/2} \\ 0 & \{3/10\}^{1/2} & \{3/10\}^{1/2} & -\{2/15\}^{1/2} & -\{2/15\}^{1/2} & -\{2/15\}^{1/2} \\ 0 & 2^{-1/2} & -2^{-1/2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 6^{-1/2} & 6^{-1/2} & -\{2/3\}^{1/2} \\ 0 & 0 & 0 & 2^{-1/2} & -2^{-1/2} & 0 \end{pmatrix}$$

In the above example, it is not possible to create further vectors $\psi^{j,k}$. There are $n = 6$ of them, and they are orthonormal. Thus, they form an orthonormal basis of $\mathbb{R}^6$. This is not a coincidence: the following general results holds.

Proposition 2.1 *The collection of vectors $\{\psi^{j,k}\}_{j,k}$ is an orthonormal basis of $\mathbb{R}^n$.*

As is clear from the example, a UH basis is determined by the set of breakpoints $\mathbf{b} = \{b^{j,k}\}_{j,k}$, which is a permutation of the set $\{1, \ldots, n-1\}$. On the other hand, not every permutation of $\{1, \ldots, n-1\}$ defines a UH basis. For example, by construction, if $b^{0,1} = 2$, then we must have $b^{1,1} = 1$. Thus the number of UH bases is strictly less than $(n-1)!$ for $n \geq 4$. The issue of basis selection will be covered in Sections 4 and 7.

Note that classical Haar vectors are a special case of the above construction with $b^{j,k} = (s^{j,k} + e^{j,k} - 1)/2$. This special case requires that n should be a power of two. Our general construction naturally avoids this restriction, in the sense that it is always possible to find a UH basis for any $n \geq 1$.

2.2 Discrete Unbalanced Haar transform

The Discrete UH Transform (DUHT) of an input vector $\mathbf{X} = \{X_i\}_{i=1}^n$ is simply the vector of inner products between $\mathbf{X}$ and $\psi^{j,k}$, for all j and k. We denote

$$\mathrm{DUHT}(\mathbf{X})^{j,k} = \langle \mathbf{X}, \psi^{j,k} \rangle,$$

where $\langle \cdot, \cdot \rangle$ is the inner product, and call $\mathrm{DUHT}(\mathbf{X})^{j,k}$ the UH coefficients of $\mathbf{X}$. In the classical Haar case, the discrete Haar transform is typically performed via a fast $O(n)$ "pyramid" algorithm proposed by Mallat (1989a), and a similar pyramid algorithm could be devised for the general DUHT. Such a fast algorithm would require that the finest scale UH coefficients (i.e. those with the largest value of j) be computed first.

However, our basis selection algorithm, detailed in Section 4, requires that the UH coefficients $\mathrm{DUHT}(\mathbf{X})^{j,k}$ be computed from coarsest to finest scales. In other words, we first require $\mathrm{DUHT}(\mathbf{X})^{0,1}$, then $\mathrm{DUHT}(\mathbf{X})^{1,1}$ and $\mathrm{DUHT}(\mathbf{X})^{1,2}$, and so forth. With this requirement, no fast pyramid algorithm is possible, and to obtain the cofficients, we calculate the inner products via direct computation.

As there are n vectors $\psi^{j,k}$, this could potentially be an $O(n^2)$ operation, which would be undesirable. We now describe a mild assumption on the UH basis which ensures that the computational complexity of the direct DUHT is of order $O(n \log n)$, which is acceptable.

We first note that the supports of the UH vectors $\psi^{j,k}$ at any one fixed scale j are disjoint. Therefore, the computation of *all* coefficients $\{\mathrm{DUHT}(\mathbf{X})^{j,k}\}_k$ at any single scale j is an $O(n)$ operation. If we could ensure that the total number of scales, denoted by $J(n)$, was logarithmic in n, the overall complexity of the DUHT would then be $O(n \log n)$. Note that this requirement is automatically satisfied for the classical Haar transform for which $J(n) = \log_2 n$.

The following assumption specifies a sufficient condition which ensures that the total number of scales $J(n)$ produced by a UH transform is logarithmic in n.

Assumption 2.1 *Let $|\psi^{j,k}|$, $|\psi^{j,k}|^+$ and $|\psi^{j,k}|^-$ denote the number of non-zero, positive and negative components of the vector $\psi_{j,k}$, respectively. There exists a fixed constant $p \in [1/2, 1)$ such that for all n, we have*

$$\max\left\{\frac{|\psi^{j,k}|^+}{|\psi^{j,k}|}, \frac{|\psi^{j,k}|^-}{|\psi^{j,k}|}\right\} \leq p, \tag{2}$$

uniformly over $j \geq 0$ and k.

The condition that both ratios should both be bounded away from 1 can be interpreted as the requirement that the UH basis should not be "too unbalanced".

Proposition 2.2 *Let $\mathbf{b} = \{b^{j,k}\}_{j,k}$ be a set of breakpoints which determines a UH basis defined on $\{1, \ldots, n\}$. Let the total number of scales j in $\mathbf{b}$ be denoted by $J(n)$. If Assumption 2.1 holds, then $J(n) \leq \lceil \log_{1/p} n \rceil$.*

Let the ith component of the vector $\psi^{j,k}$ be denoted by $\psi^{j,k}(i)$. The inverse DUHT is performed via direct multiplication and addition, using the Parseval identity

$$X_i = \sum_{j,k} \mathrm{DUHT}(\mathbf{X})^{j,k}\, \psi^{j,k}(i). \tag{3}$$

By the same argument as above, the single-scale operation $\sum_k \mathrm{DUHT}(\mathbf{X})^{j,k}\, \psi^{j,k}$ has computational order $O(n)$, for each j. Hence, if Assumption 2.1 holds and thus the number of scales j is logarithmic in n, then the inverse DUHT, defined by formula (3), can be computed in $O(n \log n)$ operations.

3 Function estimation via Unbalanced Haar thresholding

3.1 The estimation algorithm

Our aim in this section is to use the DUHT to estimate f from the regression problem (1). The estimation algorithm proceeds as follows.

1. Fix $p_0 \in [1/2, 1)$ independent of n. Choose a set of breakpoints $\mathbf{b}$ which determines a UH basis defined on $\{1, \ldots, n\}$ such that $\mathbf{b}$ satisfies Assumption 2.1 with $p = p_0$. (The issue of basis selection is covered in detail in Sections 4 and 7).

2. Perform the DUHT of the vector $\mathbf{X} = \{X_i\}_{i=1}^n$ with respect to the basis $\mathbf{b}$. Let $Y_{j,k} = \mathrm{DUHT}(\mathbf{X})^{j,k}$. After the transformation, the regression problem (1) can be rewritten as
$$Y_{j,k} = d_{j,k} + \varepsilon_{j,k},$$
where $d_{j,k} = \mathrm{DUHT}(\mathbf{f})^{j,k}$ with $\mathbf{f} = \{f(i/n)\}_{i=1}^n$ and $\varepsilon_{j,k} = \mathrm{DUHT}(\varepsilon)^{j,k}$ with $\varepsilon = \{\varepsilon_i\}_{i=1}^n$. The $d_{j,k}$'s are the true UH coefficients of $\mathbf{f}$ which are unknown and need to be estimated. The $\varepsilon_{j,k}$'s are again iid $N(0, \sigma^2)$ because of the orthonormality of the DUHT.

3. Estimate each $d_{j,k}$ by means of a suitable "universal" shrinkage rule
$$\hat{d}_{j,k} = h(Y_{j,k}, \lambda),$$
where the function h has the property that $h(y, \lambda) = 0$ if and only if $|y| \leq \lambda$, and the "threshold" parameter λ is set equal to $\sigma(2 \log n)^{1/2}$. Our specific choice of the shrinkage rule h is non-standard and is described and motivated in Section 3.2. The only exception is the coefficient $d_{-1,1}$, which is estimated by $\hat{d}_{-1,1} = Y_{-1,1}$. The rationale for using shrinkage estimators is as in classical wavelet shrinkage, see e.g. Donoho and Johnstone (1994). In short, for a large class of functions f, namely those that are "well approximated" by piecewise-constant functions, the sequence $d_{j,k}$ is often sparse, with most $d_{j,k}$'s close or equal to zero. Thus, the hope is that an appropriate threshold will be able to preserve those $Y_{j,k}$ that correspond to "significant" coefficients $d_{j,k}$, whilst setting to zero those that correspond to small values of $d_{j,k}$ and thus carry mostly noise $\varepsilon_{j,k}$. This operation ensures that a large proportion of noise $\varepsilon_{j,k}$ gets removed. The choice $\lambda = \sigma(2 \log n)^{1/2}$ is motivated by certain properties of the normal distribution and is discussed in detail in Donoho and Johnstone (1994). In practice the standard deviation parameter σ is unknown but can easily be estimated using the Median Absolute Deviation estimator on the sequence $2^{-1/2}|X_{i+1} - X_i|_{i=1}^{n-1}$.

4. Our estimator $\hat{f}_n^{\mathbf{b}}(z)$ of $f(z)$ arises as a result of the inverse DUHT of the coefficients $\hat{d}_{j,k}$ with respect to the basis $\mathbf{b}$.

Note that, assuming that the basis $\mathbf{b}$ is given in advance, the total computational complexity of the above algorithm is $O(n \log n)$. In Section 4, we demonstrate that our basis selection procedure is also of the same computational complexity.

3.2 Mean-square risk theory

Of interest to us in this section are three distinct smoothness classes for f. Let $S[0,1]$ be the space of piecewise constant functions on $[0,1]$ with a finite number of jumps. Let $PH^{\alpha}[0,1]$ be the space of piecewise Hölder-continuous functions on $[0,1]$ with Hölder exponent $\alpha \in (0,1]$ and with a finite number of breakpoints. Finally, let $BV[0,1]$ be the space of functions on $[0,1]$ of finite total variation.

Intuitively, we expect our algorithm to perform well if $f \in S[0,1]$, as the basis vectors $\psi^{j,k}$ in the Unbalanced Haar transform are also piecewise constant. Thus, the performance of our algorithm for functions f which are *not* in $S[0,1]$ will depend on how "well" f can be approximated by functions from $S[0,1]$. For $q \geq 1$, denote $\|f\|_q = (\int_0^1 |f|^q)^{1/q}$, with the extension $\|f\|_\infty = \sup |f|$. For a function $s \in S[0,1]$, we denote its number of breakpoints by $B(s)$. Assume that f is square-integrable and bounded. By standard results in approximation theory (see e.g. DeVore, 1998), if $f \in PH^{\alpha}[0,1]$ then an $s \in S[0,1]$ can be found such that $B(s) \leq m$ and $\|f - s\|_2^2 = O(m^{-2\alpha})$. Similarly, if $f \in BV[0,1]$ then there exists an $s \in S[0,1]$ such that such that $B(s) \leq m$ and $\|f - s\|_2^2 = O(m^{-2})$.

We now describe our specific choice of the shrinkage rules h used in this and later sections. For motivation behind this choice, see the discussion underneath Theorem 3.1.

- $h(y,\lambda) = h^H(y,\lambda) := y\,\mathbb{I}(|y| > \lambda)$, which yields the classical hard thresholding estimator.
- $$h(y,\lambda) = h_L^C(y,\lambda) := \begin{cases} 0 & |y| \leq \lambda \\ L(y+\lambda) & y \in [-\lambda L/(L-1), -\lambda) \\ L(y-\lambda) & y \in (\lambda, \lambda L/(L-1)] \\ y & |y| > \lambda L/(L-1), \end{cases}$$
  where $L > 1$.

The following theorem establishes the mean-square behaviour of our estimator.

Theorem 3.1 *Let $\hat{f}_n^{\mathbf{b}}$ be our estimator of f in the regression problem (1), computed with respect to any UH basis $\mathbf{b}$ satisfying Assumption 2.1 with $p = p_0$. Let f be square-integrable and bounded.*

1. *If $f \in PH^{\alpha}[0,1]$ or $f \in BV[0,1]$, and $h = h_L^C$, then*

$$\mathbb{E} \int_0^1 \left\{ \hat{f}_n^{\mathbf{b}}(z) - f(z) \right\}^2 dz \leq C_{f,\sigma,p_0,h}\, n^{-2\alpha/(1+2\alpha)} \log_{1/p_0}^2 n,$$

where $\alpha = 1$ *if* $f \in BV[0,1]$*, and* $C_{f,\sigma,p_0,h}$ *is a constant depending only on* f*,* σ*,* p_0 *and* h*.*

2. *If* $f \in S[0,1]$ *with* $B(f) \leq m$ *and* $h = h^H$ *or* $h = h^C_L$*, then*

$$\mathbb{E} \int_0^1 \left\{ \hat{f}_n^{\mathbf{b}}(z) - f(z) \right\}^2 dz \leq C_{f,\sigma,p_0,h}\, mn^{-1} \log^2_{1/p_0} n,$$

where the rates are uniform over all bases $\mathbf{b}$ *which satisfy Assumption 2.1 with* $p = p_0$*.*

The hard thresholding rule h^H performs well in practice (see Section 5) and, by the above theorem, is mean-square consistent when the target function f is piecewise constant. However, when f is in a richer function class, then a Lipschitz-continuous shrinkage rule, such as that defined by h^C_L, must be used. The intuitive argument for this, formalised in the proof of Theorem 3.1, is that (Unbalanced) Haar wavelets are not "smooth enough" for non-piecewise-constant functions f; however, this can be remedied by using a Lipschitz-continuous shrinkage rule. Note that as the Lipschitz constant L approaches infinity, h^C_L converges pointwise to h^H. Also, as L approaches one, h^C_L converges pointwise to the standard soft thresholding rule. We do not consider the latter in this paper, due to its inferior practical performance.

We also emphasise that the above result holds uniformly over all bases satisfying Assumption 2.1, and hence does not address the improvement of adapting the basis (see Section 4 for our basis selection algorithm) over the classical "balanced" Haar basis. However, on the other hand, this means that the result of Theorem 3.1 is flexible enough to accommodate *any* basis selection procedure, not only that described in Section 4. In particular, it can be used to demonstrate consistency of the UH estimation algorithm combined with the "bottom-up" basis selection described in Section 7.1.

4 Basis selection

The consistency result of Theorem 3.1 is uniform over all bases satisfying Assumption 2.1. Thus, our algorithm combined with *any* basis selection rule which respects this assumption, will still be consistent.

The basis selection rule which we propose is related to the *matching pursuit algorithm*. Despite a large amount of literature on matching pursuit (first proposed by Mallat and Zhang (1993); see also the more recent work of Donoho et al. (2006) and the numerous references therein), we are unaware of any existing work which combines the matching pursuit idea with Unbalanced Haar wavelets. In our view, this is an oversight: UH wavelets are naturally suited to the use of matching pursuit as their particular form permits an extremely fast algorithm for selecting a suitable basis. This is in contrast to typical matching pursuit implementations which, inevitably, suffer from slow speed.

We now describe the algorithm in detail. We first define the *Unbalanced Haar mother vector* $\psi_{s,b,e}$ (where the "s", "b" and "e" stand for "start", "breakpoint" and "end", respectively) with elements $\psi_{s,b,e}(l)$ defined by

$$\psi_{s,b,e}(l) = \left\{\frac{1}{b-s+1} - \frac{1}{e-s+1}\right\}^{1/2} \mathbb{I}(s \leq l \leq b) - \left\{\frac{1}{e-b} - \frac{1}{e-s+1}\right\}^{1/2} \mathbb{I}(b+1 \leq l \leq e).$$

Choosing a UH basis amounts to choosing breakpoints $b^{j,k}$ for each vector $\psi_{j,k}$. As before, our input vector is denoted by $\mathbf{X} = \{X_i\}_{i=1}^n$. We fix $p_0 \in [1/2, 1)$ independent of n.

- We choose the breakpoint $b^{0,1}$ such that the inner product between $\mathbf{X}$ and $\psi_{1,b^{0,1},n}$ is maximised in absolute value. More formally, $b^{0,1} = \text{argmax}_b |\langle \mathbf{X}, \psi_{1,b,n} \rangle|$, where the range of b is such that Assumption 2.1 holds with $p = p_0$.

  To effect this, we need to compute the inner products $\langle \mathbf{X}, \psi_{1,b,n} \rangle$ for all b. If done by "brute force", this could be an $O(n^2)$ operation. However, the particular form of the UH vectors $\psi_{1,b,n}$ means that the inner products can easily be computed iteratively in computational time $O(n)$ (much in the same way as cumulative means of a vector of length n can be computed in time $O(n)$).

- Similarly, we choose $b^{j+1,l} = \text{argmax}_b |\langle \mathbf{X}, \psi_{s^{j+1,l},b,e^{j+1,l}} \rangle|$, where $l = 2k-1, 2k$ and, again, the range of b is such that Assumption 2.1 holds with $p = p_0$.

For any fixed scale j, the supports of the vectors $\{\psi^{j,k}\}_k$ are disjoint and their joint length is at most n. Thus, if the above-mentioned iterative technique for computing the inner products is used, all breakpoints $\{b^{j,k}\}_k$ at scale j can be found in total computational time $O(n)$. As Assumption 2.1 ensures that the total number of scales is logarithmic in n, the overall computational cost of our basis selection procedure is $O(n \log n)$.

The motivation for our basis selection procedure can be outlined as follows: it is well known that wavelet thresholding is the most successful when the representation of the signal in the wavelet domain is *sparse*, see e.g. Donoho et al. (1995). In our set-up, this would require that only a few UH coefficients $\text{DUHT}(\mathbf{X})^{j,k}$ were "large" in magnitude, whilst most were "small" and thus carried mainly noise. Typically, when performing transforms with the standard balanced Haar basis, it is often observed that large Haar coefficients are mostly concentrated at coarser scales. Our basis selection procedure renders this "concentration of power" even more extreme: it attempts to concentrate as much as possible of the signal power at coarser scales, in the hope that this would further improve the sparsity of representation.

As mentioned in the Introduction, there are links between our procedure and CART, which based on "growing" a partition tree via recursive partitioning and then "pruning" it to eliminate spurious splits, where the amount of pruning is controlled via a suitable complexity penalty. The fact that we formulate our procedure in the language of wavelets permits us to use simple universal thresholding schemes to select

the (hopefully) significant splits and eliminate spurious ones (and thus reduce complexity). Therefore, in our context, the choice of the "right" complexity penalty is not an issue. The other benefit of using wavelets is that complete consistency results are easy to obtain (partly thanks to Assumption 2.1), which is in contrast to the CART case, where Gey and Nedelec (2005) acknowledge that "obtaining an upper bound for the complete risk" is "an open question".

4.1 Link to change point detection

In the special case when $f \in S[0,1]$, our basis selection and thresholding procedure is related to the binary segmentation procedure for change point detection, proposed by Sen and Srivastava (1975), and analysed theoretically by Vostrikova (1981) and Venkatraman (1993) (see also the review paper Chen and Gupta (2001), or Braun and Müller (1998) and Olshen et al. (2004) who describe the application of the procedure to DNA data). In the version described by Vostrikova (1981) (and translated to our setting) the procedure proceeds as follows: in the first step, compare the observed value of $\max_b|\langle \mathbf{X}, \psi_{1,b,n}\rangle|$ with the "critical value" of this random variable under the null hypothesis of no jumps. If the hypothesis is rejected, choose $b^{0,1} = \operatorname{argmax}_b|\langle \mathbf{X}, \psi_{1,b,n}\rangle|$ as the estimate of a jump, and proceed recursively in the same fashion on the two parts of the data separated by the previously estimated jump. Otherwise, stop. Vostrikova demonstrates consistency in probability of the resulting jump location estimators. It is worth noting that at each stage of the procedure, the observed quantity $\max_b|\langle \mathbf{X}, \psi_{s^{j,l},b,e^{j,l}}\rangle|$ is compared to the critical value of its own distribution under the null, which is difficult to compute in practice due to the stochasticity in the (previously selected) $s^{j,l}, e^{j,l}$. In a slightly more general setting, Venkatraman (1993), Chapter 2, proposes a conservative comparison threshold of magnitude which is a power of n and thus, albeit resulting in consistent estimators, is likely to underestimate the number of jumps in practice.

By contrast, in our approach, we compare each $\max_b|\langle \mathbf{X}, \psi_{s^{j,l},b,e^{j,l}}\rangle|$ to the much lower universal threshold $\sigma(2\log n)^{1/2}$, which is essential for the procedure to work for target functions which may or may not be piecewise constant. Also, binary segmentation uses $p = 1$, which is technically not allowed by our procedure (see Assumption 2.1). Another difference is that, in our case, the transform continues right to the bottom of the UH "tree" regardless of the magnitude of the coefficients. Only after the complete transform has been performed, are the coefficients thresholded, which gives the UH method a better chance of detecting "fine-scale" features of the signal which may not apparent in coarse-scale coefficients. Finally, unlike binary segmentation, our method yields a complete invertible representation of the data with respect to the selected UH wavelet basis.

Despite these differences, the similarities between our method and binary segmentation mean that asymptotically and for p large enough, jumps estimated by the UH estimator are a superset of the set of jumps detected by the binary segmentation procedure in the version proposed by Venkatraman (1993). Thus, locally constant

stretches of our estimator asymptotically guarantee that the true signal is also locally constant, which makes our estimator easy to interpret. This is unlike most of the classical "balanced" wavelet thresholding methods, where the often "arbitrary" wavelet shapes produce spurious artefacts in the output (which look like the wavelets themselves and are thus difficult to interpret), or even methods based on the balanced Haar wavelets, where jumps tend to appear at dyadic locations, which spoils the interpretability.

5 Simulation study

The aim of this section is to compare the empirical performance of our Unbalanced Haar estimation technique to the state-of-the-art competitors mentioned in the Introduction: the Taut String (TS) method due to Davies and Kovac (2001) and the Adaptive Weight Smooting (AWS) technique of Polzehl and Spokoiny (2000). The comparison is straightforward to effect as both techniques have been implemented and thoroughly documented in the R packages `ftnonpar` and `aws`, respectively. For comparison, we also investigate the performance of the classical thresholding technique based on suitably selected "balanced" wavelet bases.

Our test functions are Donoho and Johnstone's "blocks" and "bumps", sampled at 2048 equispaced points, scaled to have a variance of 3.659 and 0.443, respectively. For the blocks function, it is of interest to detect the jumps in the signal, of which there are 11. Similarly, for the bumps function, it is of interest to detect the 11 peaks. The signals are shown in the top left plots of Figures 1 and 2, respectively.

We are particularly interested in (very) low signal-to-noise-ratio regimes (i.e. very noisy signals), where the human eye is not of much help in estimating the true signal and a reliable automatic statistical technique becomes indispensable. We contaminate both signals with iid Gaussian noise with mean zero and standard deviation σ. For the blocks signal, $\sigma = 2.5$, so that the root signal-to-noise ratio is 0.765. For the bumps signal, $\sigma = 0.6$ and the root signal-to-noise ratio is 1.109. The values of σ were chosen so that the estimation problem is challenging to the human eye, but hopefully not impossible to solve accurately by means of a good statistical technique. Two simulated sample paths for the blocks and bumps signal are shown in the top right plots of Figures 1 and 2, respectively. The standard deviation of the noise is unknown to all estimation procedures and is estimated via the Median Absolute Deviation algorithm as described in Section 3.1.

With all estimation methods, we always use default parameter values; that is, to compute the TS estimate we use the `pmreg` function of the `ftnonpar` package with default parameters, and to compute the AWS estimate we use the `awsuni` function of the `aws` package, also with default parameters. For comparison, we also use the classical "Balanced" Haar (BH) method for both signals, and a method based on the non-decimated wavelet decomposition with the (balanced) Daubechies Extremal Phase wavelet with one vanishing moment (BD2) for the bumps signal, which yields

	blocks				bumps			
	10	11	12	IQR	10	11	12	IQR
AWS	44	68	73	13–19	3	5	25	17–23
TS	3	8	11	16–21	497	277	2	10–11
UH	228	461	174	11–12	376	518	62	10–11
BH	0	0	0	21–25	18	102	288	12–14
BD2	•	•	•	•	0	0	0	38–44

Table 1: Blocks signal, 1st three columns: number of sample paths (out of the 1000 simulated ones) yielding estimates with 10, 11, 12 jumps (respectively). Fourth column: the IQR of the estimated number of jumps. Bumps function: analogous results but concerning the number of peaks. The correct number of jumps (for blocks) and peaks (for bumps) is 11.

continuous reconstructions and is thought to perform particularly well for this signal. For the two balanced wavelet methods and our UH technique we use universal hard thresholding ($h = h^H$) with all scales thresholded. For the UH method, we could also have used the Lipschitz-continuous h_L^C shrinkage rule but we noticed that the empirical results were very similar to h^H if a large L was used. The "unbalancedness" parameter p, described in Assumption 2.1, was set to the default value of $p = 0.99$.

The results in Table 1 clearly illustrate that the UH is by far the most effective detector of jumps (for the blocks signal) and peaks (for the bumps signal). For the blocks signal, sample reconstructions are shown in the two bottom rows of Figure 1. The TS estimate ignores the first "dip" and displays a spurious jump at time 1024, which is probably an artefact of the dyadic multiscale stopping criterion used by the TS algorithm (although the algorithm itself is not dyadic). The AWS estimate gets the jump locations right, but tends to estimate various disconnected parts of the signal at the same level (which is natural given how the AWS algorithm works), and also exhibits a few extra small spurious jumps (which are not clearly visible in the figure). This results in a large overall bias. The BH method is by far the least accurate. On the other hand, our new technique UH yields the best reconstruction by a wide margin.

Similar comments also apply to the bumps reconstruction (see the two bottom rows of Figure 2). The TS method ignores one of the spikes, and the AWS method estimates disconnected parts of the signal at the same level, as well as exhibiting a number of small spurious extra peaks. In the BH reconstruction, the Gibbs phenomenon is noticeable. Our UH technique yields the most convincing estimate.

Table 2 shows the Mean Integrated Squared Errors (MISE) for the above simulation set-up, averaged over 1000 simulated sample paths. For the blocks signal, UH is clearly the best method, outperforming the second best technique (AWS) by 26%. It is also the best for the bumps signal, where it is marginally better than the AWS method. We mention that the BD2 technique is by far the best for the bumps signal in

	TS	AWS	BH	UH
blocks	437	264	622	195
bumps	734	672	857	670

Table 2: MISEs for the competing methods, averaged over 1000 simulated sample paths, then multiplied by 10^3 (blocks) or 10^4 (bumps) and rounded.

terms of the MISE, which is not surprising given that it produces continuous estimates (note that the bumps signal is continuous). However, as suggested by Table 1, it is an extremely poor peak detector for this signal.

Computation was almost instantaneous for all methods tested in this section.

6 Application to earthquake data

In this section, we analyse Northern California earthquake count data, available from `http://www.ncedc.org`. We analyse the time series N_k, $k = 1, \ldots, 1024$, where N_k is the number of earthquakes of magnitude 3.0 or more which occurred in the kth week, the last week being 29 Nov – 05 Dec 2000. Since N_k is count data, we first apply the Anscombe (1948) transformation $A_k = 2\{N_k + 3/8\}^{1/2}$, which brings the distribution of Poisson data closer to Gaussianity with constant variance. The series A_k is plotted in the top left plot of Figure 3.

The top right plot of Figure 3 shows the UH estimate of the mean level of A_k. Our technique identifies 17 "spikes" in the mean intensity. It is fascinating to observe that for the majority of them (13 out of 17), the mean intensity just after the spike returns to a level which is *higher* than the mean intensity just before the spike.

Note that this feature is not immediately easy to pick up from the classical "balanced" Haar estimate plotted in the middle left plot, or from the estimate based on the non-decimated wavelet decomposition with the (balanced) Daubechies Extremal Phase wavelet with one vanishing moment, plotted in the middle right plot. This is due to a number of (possibly) spurious downward spikes, which impair the picture.

A possible explanation for the "fast-rise-slow-decay" phenomenon is the often-observed increase in seismicity rates following a large event at a possibly remote location, described in detail in Ziv (2006). For example, on 16 October 1999, the "Hector Mine" earthquake of magnitude 7.1 occurred in California, and a significant increase in seismicity rates has been observed at locations throughout the state following this event. Although the event itself is not included in the Northern California database (as it occurred too far to the south), it is interesting to note that our UH estimate displays a "spike" corresponding to weeks numbered 965 to 969, starting 13 October and ending 16 November 1999 (it is the last spike shown in the bottom plot of Figure 3). Furthermore, just after the spike, the estimated mean intensity is larger than just before the spike. One cannot help but wonder if this sudden increase and subsequent

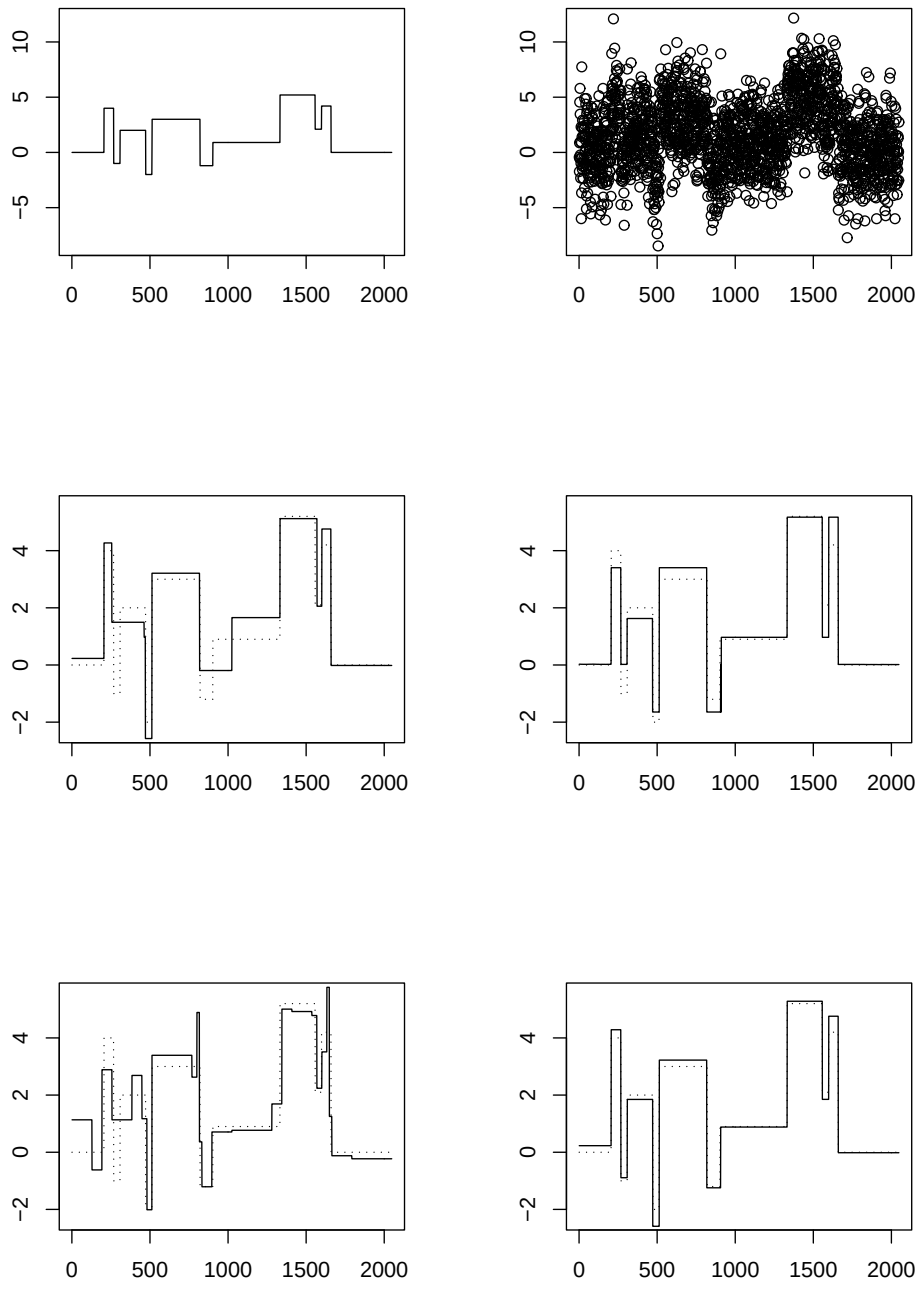

Figure 1: Top row: the blocks signal (left) and a simulated sample path (right). Middle row: reconstruction using the TS (left) and AWS methods (right). Bottom row: reconstruction using the BH (left) and UH methods (right).

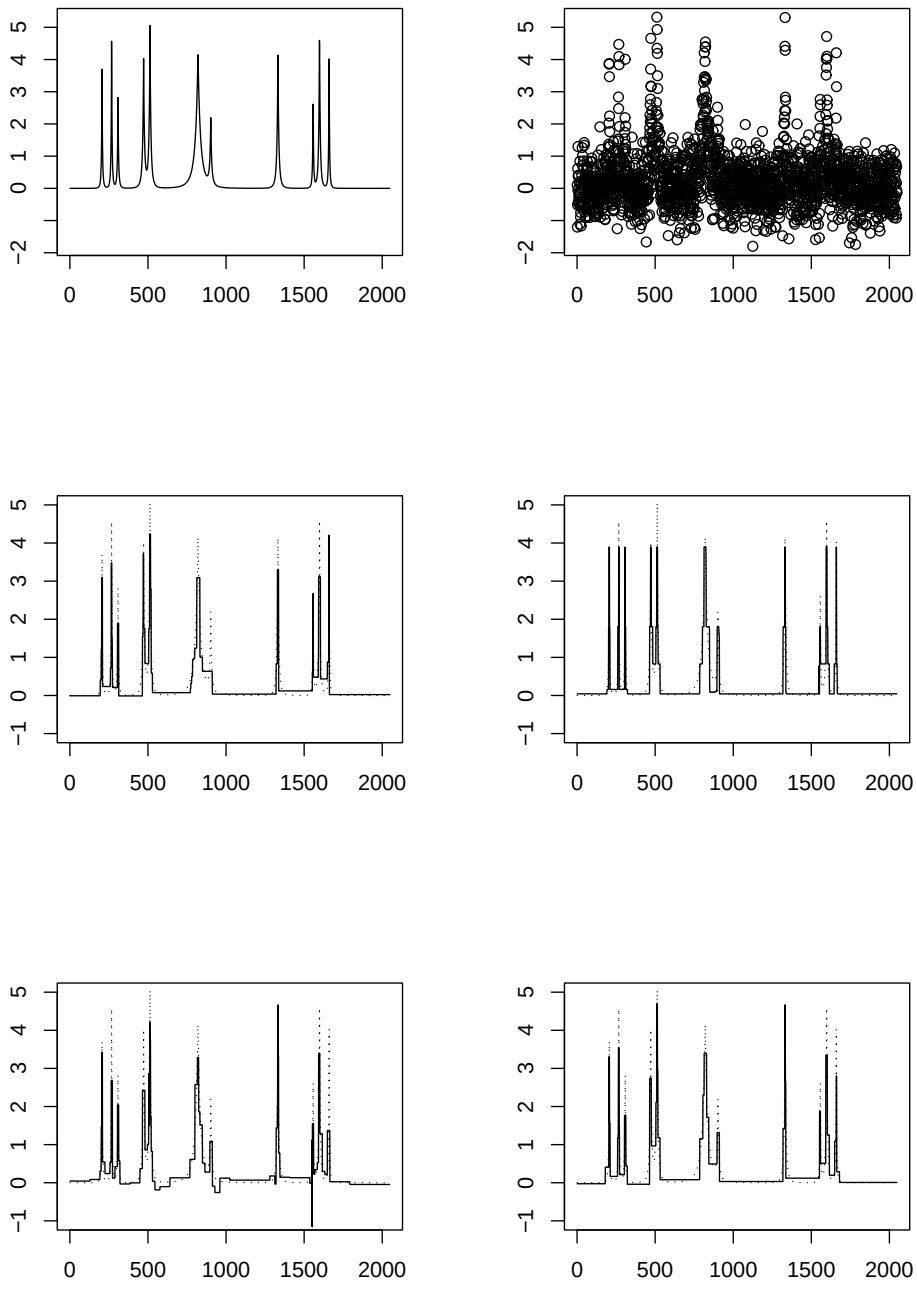

Figure 2: Top row: the bumps signal (left) and a simulated sample path (right). Middle row: reconstruction using the TS (left) and AWS methods (right). Bottom row: reconstruction using the BH (left) and UH methods (right).

slow decline in the seismicity rate in Northern California might have been triggered by the major "Hector Mine" event in the southern part of the state.

7 Modifications and extensions

In this section, we briefly discuss how the UH wavelet methodology can be extended to other smoother unbalanced wavelet bases, and to image data. Both extensions will utilise the idea of *bottom-up* UH transforms, which we introduce next.

7.1 Bottom-up Unbalanced Haar transforms

The main idea of our basis selection algorithm of Section 4 can be summarised as follows: using a greedy algorithm, attempt to concentrate as much power of the signal as possible at coarse scales. As the algorithm proceeds from the coarsest to finest scale, it will be referred to later as the *top-down* UH basis selection algorithm.

An interesting alternative would be to proceed from the finest to coarsest scale, attempting to concentrate as *little* power as possible at fine scales, which would hopefully produce a similar effect: concentrate the bulk of the power of the signal at coarse scales (this strategy can be termed a *generous*, as opposed to "greedy", algorithm). Later in this section, we will argue that such a *bottom-up* UH basis selection algorithm is a natural starting point for the meaningful extension of the UH idea to smoother wavelet bases and to image data.

We now outline the skeleton of the bottom-up UH basis selection algorithm:

1. As in classical discrete wavelet transforms, assign the initial "smooth" coefficients to be the data: $\mathbf{s} = (s_{1,1}, s_{2,2}, \ldots, s_{n,n}) := (X_1, X_2, \ldots, X_n)$. The two subscripts in $s_{p,q}$ denote the initial (p) and final (q) index of the subset of the data which corresponds to $s_{p,q}$. For example, initially, $s_{1,1}$ corresponds to X_1.

2. Search the vector $\mathbf{s}$ for the finest-scale detail coefficient which is the lowest in magnitude. To be more precise, proceed as follows: for each pair of neighbours $(s_{p,q}, s_{q+1,r})$, construct a "detail" filter $(a_{p,q}, -b_{q+1,r})$, where $a_{p,q}, b_{q+1,r} > 0$, in the following way.

   (a) As with the classical balanced Haar transform and the top-down UH transform, we desire that the bottom-up UH transform should annihilate constants. Thus, one requirement on $(a_{p,q}, -b_{q+1,r})$ is that if $(X_p, \ldots, X_r)$ is a constant vector, the detail coefficient, defined by $d_{p,r} := a_{p,q}s_{p,q} - b_{q+1,r}s_{q+1,r}$, should be zero.

   (b) To preserve the orthonormality of the transform, another requirement on $(a_{p,q}, -b_{q+1,r})$ is $a_{p,q}^2 + b_{q+1,r}^2 = 1$.

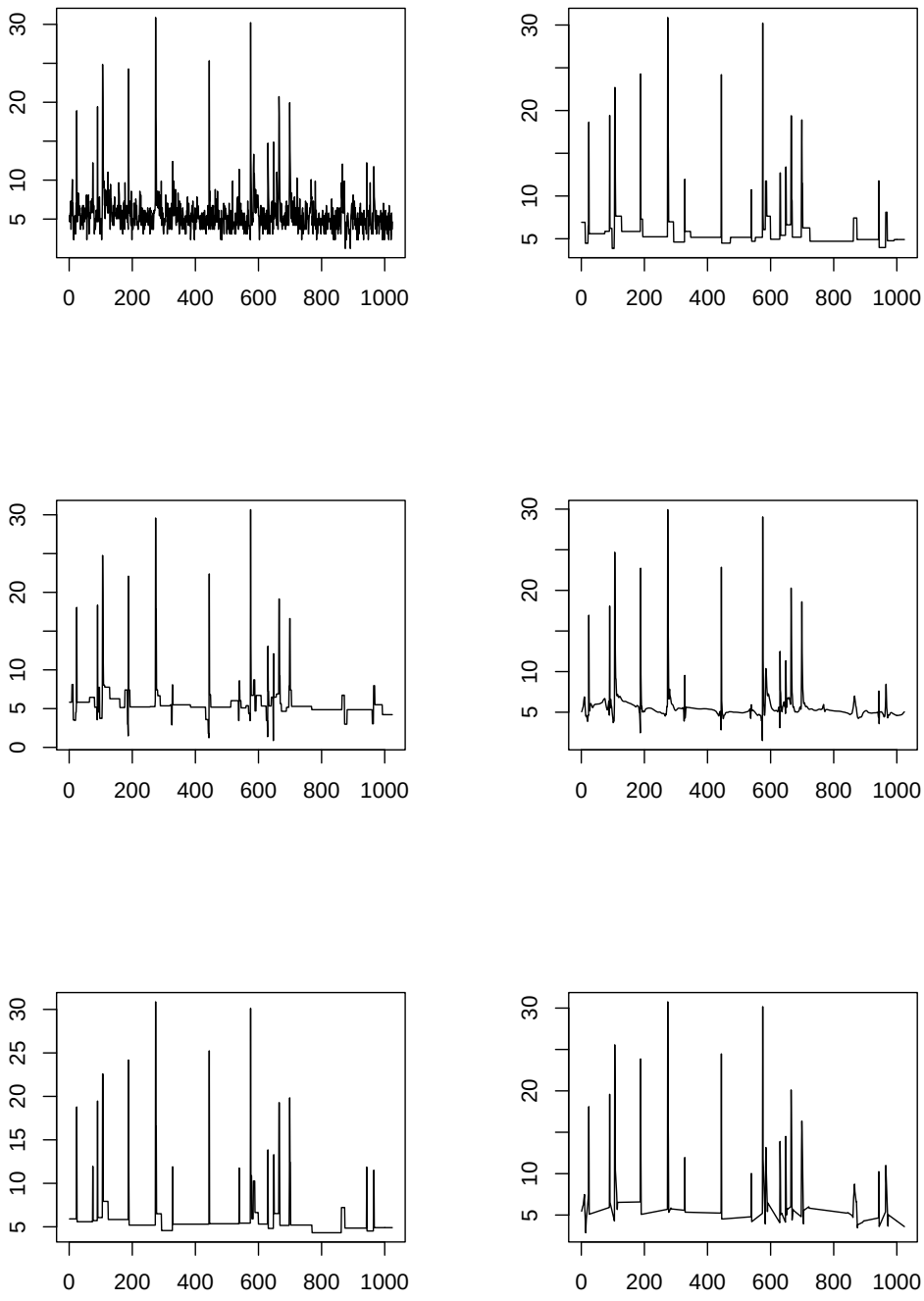

Figure 3: The A_k series (top left); its mean estimated via UH (top right), BH (middle left), BD2 (middle right), and two methods described in Section 7: BU-UH (bottom left) and UW1 (bottom right). See text for discussion.

	blocks				bumps			
	10	11	12	IQR	10	11	12	IQR
BU-UH	226	431	85	11–13	182	729	79	11–11

Table 3: Results as in Table 1, for the BU-UH method.

The two requirements (a) and (b) above determine $(a_{p,q}, -b_{q+1,r})$.

3. Select $(p_0, r_0) = \text{argmin}_{(p,r)}|d_{p,r}|$ and denote by q_0 the corresponding q. Using the filter $(b_{q_0+1,r_0}, a_{p_0,q_0})$, which is orthogonal to $(a_{p_0,q_0}, -b_{q_0+1,r_0})$, produce the corresponding "smooth" coefficient $s_{p_0,r_0} = b_{q_0+1,r_0}s_{p_0,q_0} + a_{p_0,q_0}s_{q_0+1,r_0}$.

4. Store the detail coefficient d_{p_0,r_0} in the array of wavelet coefficients. Replace the pair of neighbours $(s_{p_0,q_0}, s_{q_0+1,r_0})$ with the new smooth coefficient s_{p_0,r_0}.

5. Go to 2., unless $(p_0, r_0) = (1, n)$.

Table 3 illustrates the performance of our UH denoising algorithm where the top-down basis selection procedure of Section 4 has been replaced by the bottom-up procedure described above. The bottom-up UH algorithm (BU-UH) does marginally worse than the top-down one at detecting the jumps in the blocks signal, but does significantly better at detecting the peaks in the bumps signal. It also results in slightly inferior MISE: 220 for blocks, and 776 for bumps.

The bottom left plot of Figure 3 shows the estimate of the earthquake intensity for the BU-UH method. Note that despite the overall visual similarity to the BU reconstruction, the numbers of detected peaks differ for the two methods: BU detects 17 peaks, and BU-UH – 18.

We close this section by noting that the difference between the above bottom-up algorithm and the "lifting one coefficient at a time" paradigm of Jansen et al. (2006) is that in the latter work, the order of the detail coefficients to be computed depends only on the (irregular) *design* of the data. In our work, the design is regular, and the order of the detail coefficients to be computed depends on the *values* of the data, as highlighted in point 3. of the algorithm above.

7.2 Extension to image data

As with the classical balanced Haar transform, an image denoising technique based on the top-down UH transform (where the best-fitting 2D UH wavelets were selected from the coarsest to finest scale) would result in unwelcome "blocky" artefacts in the reconstructed image due to the particular form of the basis functions. To prevent this, a more desirable option is to use the bottom-up UH transform.

The bottom-up UH transform for images proceeds in complete analogy to the 1D case, except that it uses a different definition of two pixels, or clusters of pixels, being

"neighbours". We define two pixels to be neighbours if one adjoins the other from one of the four directions: W, N, E, S. Two clusters of pixels are said to be neighbours if one contains a pixel which is a neighbour of a pixel from the other cluster. At each stage of the algorithm, all pairs of neighbours are searched for the smallest detail coefficient as in the 1D case.

Rapid implementation of the algorithm is possible, but non-trivial, partly because it involves manipulation of graph-like data structures. It is currently the topic of our investigation, and will be reported elsewhere.

7.3 Extension to smoother unbalanced wavelet bases

The bottom-up 1D UH transform is also a convenient starting point for the extension of the methodology to "smoother" unbalanced wavelet bases, i.e. bases which annihilate not only local constants but also local polynomials of higher degrees. In this section, we briefly introduce the idea using the example of an unbalanced wavelet transform with annihilates local linear functions. We note that there is no obvious way of effecting this extension using the top-down approach.

The transform, denoted UW1, proceeds like the bottom-up UH transform for 1D data, except we now look at each triple of neighbours $(s_{p,q}, s_{q+1,r}, s_{r+1,u})$. For each triple, we construct a detail filter $(-a_{p,q}, b_{q+1,r}, -c_{r+1,u})$, where $a_{p,q}, b_{q+1,r}, c_{r+1,u} > 0$, by requiring that the filter should annihilate constants and linear functions, and sum to one in the l_2 norm. Having chosen the triple which produces the smallest detail coefficient in absolute value, we store the detail coefficient in the array of wavelet coefficients, and replace the triple by two new smooth coefficients, created by applying two filters orthonormal to the corresponding detail filter and to each other. We then proceed recursively for as long as feasible.

The corresponding smoothing method, also denoted UW1, proceeds like the UH method but replaces the UH transform with the UW1 transform. We briefly illustrate it by applying it to the earthquake data of Section 6. The reconstruction, shown in the bottom right plot of Figure 3, is continuous and seems to display fewer spurious downward spikes than the BD2 estimate.

Proofs

Proof of Proposition 2.1. We first show that there are $n-1$ vectors $\psi^{j,k}$ with $j \geq 0$. This is clearly true for $n = 1$, as in this case it is not possible to split the domain at all and thus no vectors $\psi^{j,k}$ with $j \geq 0$ are created. Let us assume that the statement holds for $n = 1, \ldots, m-1$. We will show that it holds for $n = m$ $(m \geq 2)$. In the case $n = m$, we start with a vector $\psi^{0,1}$ with a breakpoint $b^{0,1} \in \{1, \ldots, m-1\}$. By the inductive assumption, we create an extra $b^{0,1}-1$ vectors $\psi^{j,k}$ supported on the left subinterval $\{1, \ldots, b^{0,1}\}$, and an extra $m - b^{0,1} - 1$ vectors

$\psi^{j,k}$ supported on the right subinterval $\{b^{0,1}+1,\ldots,m\}$. Altogether, we have created $1+(b^{0,1}-1)+(m-b^{0,1}-1)=m-1=n-1$ vectors $\psi^{j,k}$ with $j\geq 0$, which completes the first part of the proof.

Including the extra vector $\psi^{-1,1}$, we obtain the total of n vectors $\psi^{j,k}$. It remains to be shown that they are orthonormal.

By construction, any two vectors either have disjoint supports, or the support of one vector is contained in the interval where the other one is constant. Thus $\{\psi^{j,k}\}_{j,k}$ are orthogonal. Given that the l_2 norm of each $\psi^{j,k}$ is one, they are also orthonormal.

As we have n orthonormal vectors $\psi^{j,k}$ defined on $\{1,\ldots,n\}$, they form an orthonormal basis for $\mathbb{R}^n$. □

Proof of Proposition 2.2. The total number $J(n)$ of scales is maximised when for all j and k, the ratio of the length of support of the positive part of $\psi^{j,k}$ to the length of support of its negative part is as close as possible to p. Thus, to obtain an upper bound on $J(n)$, it suffices to consider this "extremely unbalanced" case. The following recursive inequality holds:

$$J(n)\leq 1+J(\lfloor np\rfloor)\leq\ldots\leq l+J(\lfloor np^l\rfloor). \tag{4}$$

Taking $l=\lceil\log_p 1/n\rceil$, we get $\lfloor np^l\rfloor=\lfloor np^{\lceil\log_p 1/n\rceil}\rfloor\leq\lfloor np^{\log_p 1/n}\rfloor=1$. Thus, by (4), $J(n)\leq\lceil\log_p 1/n\rceil+J(1)=\lceil\log_p 1/n\rceil$, which completes the proof. □

Proof of Theorem 3.1. We first prove Case 1. C denotes a generic positive constant. Let f^m be an approximation of f in $S[0,1]$ such that $B(f^m)\leq m$ and $\|f^m-f\|_2^2=O(m^{-2\alpha})$. Note that since $\|f\|_\infty<\infty$, also $\|f^m\|<\infty$. Let f_n^m denote a function created by translating each breakpoint of f^m to the nearest multiple of $1/n$. As $\|f^m\|<\infty$, we have $\|f_n^m-f^m\|_2^2=O(m/n)$. We have

$$\mathbb{E}(\|\hat{f}_n^{\mathbf{b}}-f\|_2^2)\leq C\|f^m-f\|_2^2+C\|f_n^m-f^m\|_2^2+C\,\mathbb{E}(\|\hat{f}_n^{\mathbf{b}}-f_n^m\|_2^2). \tag{5}$$

We now focus on the last term above. Consider an (unobserved) regression problem

$$\tilde{X}_i=f_n^m(i/n)+\varepsilon_i,\quad i=1,\ldots,n. \tag{6}$$

After the DUHT with basis $\mathbf{b}$, (6) becomes $\tilde{Y}_{j,k}=\tilde{d}_{j,k}+\varepsilon_{j,k}$, and we estimate each $\tilde{d}_{j,k}$ by means of our UH estimator $\hat{\tilde{d}}_{j,k}=h_L^C(\tilde{Y}_{j,k},\sigma(2\log n)^{1/2})$, except $\tilde{d}_{-1,1}$ which we estimate by $\hat{\tilde{d}}_{-1,1}=\tilde{Y}_{-1,1}$. We then take the inverse DUHT to obtain the estimate $\hat{f}_n^{m,\mathbf{b}}$ of f_n^m. We have $\mathbb{E}(\|\hat{f}_n^{\mathbf{b}}-f_n^m\|_2^2)\leq C(\mathbb{E}(\|\hat{f}_n^{\mathbf{b}}-\hat{f}_n^{m,\mathbf{b}}\|_2^2)+\mathbb{E}(\|\hat{f}_n^{m,\mathbf{b}}-f_n^m\|_2^2))=:C(I+II)$. We first consider I. Using the fact that both $\hat{f}_n^{\mathbf{b}}$ and $\hat{f}_n^{m,\mathbf{b}}$ are piecewise constant with possible jumps only at multiples of $1/n$, the Parseval identity, and the Lipschitzness

of h_L^C, we obtain

$$\begin{aligned} I &= \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left\{\hat{f}_n^{\mathbf{b}}(i/n)-\hat{f}_n^{m,\mathbf{b}}(i/n)\right\}^2=\frac{1}{n}\sum_{j,k}\mathbb{E}\left\{\hat{d}_{j,k}-\hat{\tilde{d}}_{j,k}\right\}^2=\frac{1}{n}\mathbb{E}\{Y_{-1,1}-\tilde{Y}_{-1,1}\}^2 \\ &+ \frac{1}{n}\sum_{(j,k)\neq(-1,1)}\mathbb{E}\left\{h_L^C(Y_{j,k},\sigma(2\log n)^{1/2})-h_L^C(\tilde{Y}_{j,k},\sigma(2\log n)^{1/2})\right\}^2 \\ &\leq \frac{L^2}{n}\sum_{j,k}\mathbb{E}\left\{Y_{j,k}-\tilde{Y}_{j,k}\right\}^2=\frac{L^2}{n}\sum_{j,k}(d_{j,k}-\tilde{d}_{j,k})^2=\frac{L^2}{n}\sum_{i=1}^{n}\{f(i/n)-f_n^m(i/n)\}^2 \\ &= O(m^{-2\alpha}+m/n), \end{aligned}$$

where the last rate follows by the same arguments as in the first paragraph of the proof. We now turn to II. Take any $\tilde{d}_{j,k}$ for $(j,k)\neq(-1,1)$.

Case (a): f_n^m is constant over the support of $\psi_{j,k}$, which implies $\tilde{d}_{j,k}=0$. We have

$$\begin{aligned} \mathbb{E}\left\{\hat{\tilde{d}}_{j,k}-\tilde{d}_{j,k}\right\}^2 &= \mathbb{E}\left\{h_L^C(\tilde{Y}_{j,k},\sigma(2\log n)^{1/2})\right\}^2\leq\mathbb{E}\left\{h^H(\tilde{Y}_{j,k},\sigma(2\log n)^{1/2})\right\}^2 \\ &= \sigma^2\sqrt{2/\pi}\int_{(2\log n)^{1/2}}^{\infty}x^2\exp(-x^2/2)dx \\ &\leq \sqrt{2/\pi}\frac{\sigma^2}{n}\left\{(2\log n)^{1/2}+(2\log n)^{-1/2}\right\}=O(n^{-1}(\log n)^{1/2}), \end{aligned}$$

where the last inequality follows by noting that $1-\Phi(x)\leq\phi(x)/x$, where $\phi(x)$ ($\Phi(x)$) denotes the pdf (cdf) of standard normal.

Case (b): f_n^m is not constant over the support of $\psi_{j,k}$, which implies that, possibly, $\tilde{d}_{j,k}\neq 0$. W.l.o.g., assume $\tilde{d}_{j,k}>0$. We have

$$\mathbb{E}\{\hat{\tilde{d}}_{j,k}-\tilde{d}_{j,k}\}^2\leq 2\mathbb{E}\{\hat{\tilde{d}}_{j,k}-h^H(\tilde{Y}_{j,k},\sigma(2\log n)^{1/2})\}^2+2\mathbb{E}\{h^H(\tilde{Y}_{j,k},\sigma(2\log n)^{1/2})-\tilde{d}_{j,k}\}^2. \tag{7}$$

Since $|\hat{\tilde{d}}_{j,k}-h^H(\tilde{Y}_{j,k},\sigma(2\log n)^{1/2})|\leq\sigma(2\log n)^{1/2}$, the first term above is bounded by $4\sigma^2\log n$. We bound the second term as follows.

$$\begin{aligned} 2\mathbb{E}\{h^H(\tilde{Y}_{j,k},\sigma(2\log n)^{1/2})-\tilde{d}_{j,k}\}^2 &\leq 4\mathbb{E}\{(\tilde{Y}_{j,k}-\tilde{d}_{j,k})\mathbb{I}(|\tilde{Y}_{j,k}|>\sigma(2\log n)^{1/2})\}^2 \\ +4\mathbb{E}\{\tilde{d}_{j,k}\mathbb{I}(|\tilde{Y}_{j,k}|<\sigma(2\log n)^{1/2})\}^2 &\leq 4\sigma^2+4\tilde{d}_{j,k}^2\mathbb{P}(|\tilde{Y}_{j,k}|<\sigma(2\log n)^{1/2}). \end{aligned}$$

Using Markov's inequality, we bound

$$\begin{aligned} \mathbb{P}(|\tilde{Y}_{j,k}|<\sigma(2\log n)^{1/2}) &\leq \mathbb{P}(\tilde{Y}_{j,k}<\sigma(2\log n)^{1/2})=\mathbb{P}(\sigma(2\log n)^{1/2}+\tilde{d}_{j,k}-\tilde{Y}_{j,k}\geq\tilde{d}_{j,k}) \\ &\leq \mathbb{E}\{\sigma(2\log n)^{1/2}+\tilde{d}_{j,k}-\tilde{Y}_{j,k}\}^2\tilde{d}_{j,k}^{-2}\leq(4\sigma^2\log n+2\sigma^2)\tilde{d}_{j,k}^{-2}. \end{aligned}$$

Putting together the above, we bound (7) by $4\sigma^2(5\log n+3)$.

We are now ready to evaluate II. The function f_n^m has at most m jumps. Since Assumption 2.1 holds, the total number $J(n)$ of scales is bounded by $\lceil\log_{1/p_0}n\rceil$ by

Proposition 2.2. As the vectors $\psi_{j,k}$ within each scale j have non-overlapping supports, the maximum number of non-zero coefficients $\tilde{d}_{j,k}$ within each scale j is m. Thus, the total number of coefficients $\tilde{d}_{j,k} \neq 0$ is bounded by $m\lceil \log_{1/p_0} n \rceil$.

Using the fact that both $\hat{f}_n^{m,\mathbf{b}}$ and f_n^m are are piecewise constant with possible jumps only at multiples of $1/n$, and the Parseval identity, we have

$$\begin{aligned} II &= \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}\left\{\hat{f}_n^{m,\mathbf{b}}(i/n) - f_n^m(i/n)\right\}^2 = \frac{1}{n}\sum_{j,k} \mathbb{E}\left\{\hat{\tilde{d}}_{j,k} - \tilde{d}_{j,k}\right\}^2 \\ &= \frac{\sigma^2}{n} + \frac{1}{n}\sum_{(j,k)\neq(-1,1),\tilde{d}_{j,k}\neq 0} \mathbb{E}\left\{\hat{\tilde{d}}_{j,k} - \tilde{d}_{j,k}\right\}^2 + \frac{1}{n}\sum_{(j,k)\neq(-1,1),\tilde{d}_{j,k}=0} \mathbb{E}\left\{\hat{\tilde{d}}_{j,k} - \tilde{d}_{j,k}\right\}^2 \\ &\leq \frac{\sigma^2}{n} + \frac{m}{n}\lceil \log_{1/p_0} n \rceil 4\sigma^2(5\log\, n + 3) + \frac{1}{n} n n^{-1} (\log\, n)^{1/2} = O(mn^{-1}\log^2_{1/p_0} n). \end{aligned}$$

Putting these results together, we obtain the final rate for (5) as $O(m^{-2\alpha} + mn^{-1}\log^2_{1/p_0} n)$. Equating $m^{-2\alpha}$ and m/n, we get the "optimal" $m = n^{1/(1+2\alpha)}$, which yields the rate of $O(n^{-2\alpha/(1+2\alpha)}\log^2_{1/p_0} n)$ as advertised. This completes the proof of Case 1.

The proof for Case 2 proceeds almost exactly like the proof of the bound for the quantity II above, yielding the rate of $O(mn^{-1}\log^2_{1/p_0} n)$ as required. □

References

F.J. Anscombe. The transformation of Poisson, binomial and negative-binomial data. *Biometrika*, 35:246–254, 1948.

J.V. Braun and H.G. Müller. Statistical methods for DNA sequence segmentation. *Stat. Science*, 13:142–162, 1998.

L. Breiman, J. Friedman, R. Olhsen, and C.J. Stone. *Classification and Regression Trees*. Wadsworth, 1983.

J. Chen and A.K. Gupta. On change point detection and estimation. *Comm. in Statistics – Simulation and Computation*, 30:665–697, 2001.

P.L. Davies and A. Kovac. Local extremes, runs, strings and multiresolution (with discussion). *Ann. Statist.*, 29:1–65, 2001.

V. Delouille, J. Franke, and R. von Sachs. Nonparametric stochastic regression with design-adapted wavelets. *Sankhya Ser. A*, 63:328–366, 2001.

R. DeVore. Nonlinear approximation. *Acta Numerica*, 7:51–150, 1998.

D. Donoho, I. Johnstone, G. Kerkyacharian, and D. Picard. Wavelet shrinkage: asymptopia? (with discussion). *J. Roy. Stat. Soc. B*, 57:301–369, 1995.

D. L. Donoho and I. M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81:425–455, 1994.

D.L. Donoho. CART and best-ortho-basis: a connection. *Ann. Statist.*, 25:1870–1911, 1997.

D.L. Donoho, M. Elad, and V.N. Temlyakov. Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Trans. Inf. Theory*, 52:6–18, 2006.

S. Gey and E. Nedelec. Model selection for CART regression trees. *IEEE Trans. Inf. Th.*, 51:658–670, 2005.

M. Girardi and W. Sweldens. A new class of unbalanced Haar wavelets that form an unconditional basis for L_p on general measure spaces. *J. Fourier Anal. Appl.*, 3: 457–474, 1997.

M. Jansen, G.P. Nason, and B.W. Silverman. Multiscale methods for data on graphs and irregular multidimensional situations. *Preprint*, 2006.

E.D. Kolaczyk and R.D. Nowak. Multiscale likelihood analysis and complexity penalized estimation. *Ann. Statist.*, 32:500–527, 2004.

S. Mallat. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Trans. Pattn Anal. Mach. Intell.*, 11:674–693, 1989a.

S. Mallat. Multiresolution approximations and wavelet orthonormal bases of $\mathbb{L}_2(\mathbb{R})$. *Trans. Amer. Math. Soc*, 315:69–87, 1989b.

S. Mallat and Z. Zhang. Matching pursuit with time-frequency dictionaries. *IEEE Trans. Sig. Proces.*, 41:3397–3415, 1993.

M. H. Neumann. Spectral density estimation via nonlinear wavelet methods for stationary non-Gaussian time series. *Journal of Time Series Analysis*, 17:601–633, 1996.

A.B. Olshen, E.S. Venkatraman, R. Lucito, and M. Wigler. Circular binary segmentation for the analysis of array-based DNA copy number data. *Biostatistics*, 5: 557–572, 2004.

J. Polzehl and V. Spokoiny. Adaptive weights smoothing with applications to image restoration. *J. R. Stat. Soc. B*, 62:335–354, 2000.

A. Sen and M.S. Srivastava. On tests for detecting change in mean. *Ann. Statist.*, 3: 98–108, 1975.

E.S. Venkatraman. *Consistency results in multiple change-point problems*. PhD thesis, Stanford University, 1993.

B. Vidakovic. *Statistical Modeling by Wavelets*. Wiley, New York, 1999.

L. J. Vostrikova. Detecting 'disorder' in multidimensional random processes. *Soviet Math. Dokl.*, 24:55–59, 1981.

A. Ziv. On the role of multiple interactions in remote aftershock triggering: the Landers and the Hector Mine case studies. *Bull. Seismological Soc. of Amer.*, 96: 80–98, 2006.