Multiscale Local Polynomial Decompositions using bandwidths as scales

Maarten Jansen
Mohamed Amghar
Université libre de Bruxelles, department of Mathematics

August 2016 (final version)

Abstract

The Multiscale Local Polynomial Transform (MLPT), developped in this paper, combines the benefits from local polynomial smoothing with sparse multiscale decompositions. The contribution of the paper is twofold. First, it focusses on the bandwidths used throughout the transform. These bandwidths operate as user controlled scales in a multiscale analysis, which is explained to be of particular interest in the case of nonequispaced data. The paper presents both a likelihood based optimal bandwidth selection and a fast, heuristic approach. The second contribution of the paper is the combination of local polynomial smoothing with orthogonal prefilters, similar to Daubechies' wavelet filters, but defined on irregularly spaced covariate values.

Acknowledgement

Research support by the IAP research network grant nr. P7/06 of the Belgian government (Belgian Science Policy) is gratefully acknowledged.

1 Introduction

Multiscale local polynomial decompositions have recently been proposed [Jansen, 2013] as an alternative for wavelet transforms on irregularly spaced observations, referred to as second generation wavelets [Sweldens, 1998]. Both types of data decompositions are constructed using the lifting scheme [Sweldens, 1996]. The lifting scheme provides a sequence of linear operations defining a filterbank. This filterbank is a linear operation that maps an input vector onto a coarse scale approximative vector plus a vector representing the offset between fine scale input and coarse scale approximation. A repeated application of a filterbank takes each time the coarse scale approximation of the previous step as new input. The collection of detail coefficients at successive scales constitutes either a wavelet transform or a multiscale local polynomial decomposition.

An important difference between a wavelet transform and a multiscale local polynomial decomposition lies in the way it is designed. In particular, the central notion of the scales in the multiscale decompositions is introduced in a different way. In a wavelet transform, the calculation of a coarse scale coefficient involves a fixed number of adjacent fine scale coefficients. The distance between those points determines the working scale. In a multiscale local polynomial transform, the basic operation in the filterbanks is a local polynomial smoother. The operations are steered by a kernel function, whose bandwidth fixes the number of input points in the smoothing. Scale is thus imposed explicitly by the selections of the successive bandwidths.

Bandwidth selection has been an important topic in the literature of kernel and local polynomial smoothing [Fan and Gijbels, 1996, Chapter 3], but the context of this paper is different. In uniscale kernel methods, even with local bandwidths [Vieu, 1991], the optimal bandwidth finds the best trade-off between squared bias and variance or between goodness of fit and smoothness for use in a linear smoothing method. In this paper, the successive bandwidths are parameters in a linear multiscale decomposition that prepares for a nonlinear processing. The objective is to find bandwidths that lead to optimal multiscale decomposition, in the sense that the resulting decomposition is as easy as possible to work with. More precisely, we optimise the bandwidths with respect to the sparsity of the decomposition.

This paper is structured as follows. Section 2 presents the general concepts of the Multiscale Local Polynomial Decompositions, as proposed in [Jansen, 2013]. Section 3 has an original contribution about the design of the Multiscale Local Polynomial Decompositions, using orthogonal prefilters that are related to Daubechies's orthogonal wavelet filters. Then, Section 4 discusses the choice of the bandwidths in the Multiscale Local Polynomial Decomposition, which is the second contribution of this paper. Section 5 contains a simulation study, comparing a fast heuristic bandwidth choice with the data dependent choice in Section 4. A real data illustration with astronomical spectra is given in Section 6. The concluding section summarises the algorithm and it lists the benefits from combining local polynomial smoothing and sparse multiscale analysis.

2 Multiscale Local Polynomial Decompositions

2.1 Forward and inverse transform

Suppose we are given a signal $f(x)$, observed in n covariate values $x_i \in [0,1]$ under additive, independent and homoscedastic errors. The covariate values are assumed to be order statistics from a random design, formalised as

$$(Y_i|X_{(i)} = x_i) = f(x_i) + \sigma Z_i, \tag{1}$$

where the errors are assumed to be normal and independently distributed and the design points are assumed to fluctuate around the equidistant grid, i.e., $E(X_{(i)}) =$

i/n. The function $f(x)$ is assumed to be piecewise smooth, meaning that it is Lipschitz ν continuous, for positive ν, possibly larger than one, except in a finite subset of $[0, 1]$. The subset contains discontinuities, i.e., jumps, or other singularities, such as cusps.

The Multiscale Local Polynomial Decomposition takes the observations as fine scale input, assigning $s_{J,k} = Y_{k+1}$ for $k = 0, \ldots, n-1$. The index J refers to the finest or highest resolution level. In vector form, this is $\boldsymbol{s}_J = \boldsymbol{Y}$, where $\boldsymbol{s}_J$ are known as scaling coefficients. At the same time, we let $n_J = n$ denote the length of $\boldsymbol{s}_J$ and $\boldsymbol{x}_J = \boldsymbol{x}$ the corresponding vector of covariates. The multiscale decomposition constructs successive approximations $\boldsymbol{s}_j$ of $\boldsymbol{s}_J$, associated with a subsampled vector of $\boldsymbol{x}_J$, defined as follows.

Definition 1 *(subsampling) Suppose that $\boldsymbol{x}_{j+1}$ is a vector of length n_{j+1} at a resolution level labelled $j+1$. Let $e(j+1) \subset \{1, \ldots, n_{j+1}\}$ be a subset of indices. Then the* subsampled *vector at resolution level j is $\boldsymbol{x}_j = \boldsymbol{x}_{j+1,e(j+1)}$. The* subsampling matrix $\widetilde{\mathbf{J}}_j$ *is the $n_j \times n_{j+1}$ rectangular matrix obtained by taking the all rows r from the $n_{j+1} \times n_{j+1}$ identity matrix for which $r \in e(j+1)$, so that $\boldsymbol{x}_j = \widetilde{\mathbf{J}}_j \boldsymbol{x}_{j+1}$. If $e(j+1) = \{2k | k = 0, \ldots, n_j\}$, then the subsampling is termed* dyadic*, meaning that $x_{j,k} = x_{j+1,2k}$ and $n_j = \lceil n_{j+1}/2 \rceil$.*

The rest of the paper will adopt the short version $\boldsymbol{x}_j = \boldsymbol{x}_{j+1,e}$ to denote the subsampled vector whenever no confusion is possible. Moreover, all subsampling in this paper is dyadic, although it should be emphasised that extension towards other subsampling schemes is straightforward. A simple coarse scale approximation of the observations can be obtained by simple subsampling $\boldsymbol{s}_j = \boldsymbol{s}_{j+1,e} = \widetilde{\mathbf{J}}_j \boldsymbol{s}_{j+1}$. Alternatively, it is possible to apply a prefiltering operation [Jansen, 2013] including a subsampling. More precisely, $\widetilde{\mathbf{J}}_j$ can be replaced by a general $n_j \times n_{j+1}$ rectangular matrix $\widetilde{\mathbf{F}}_j$,

$$\boldsymbol{s}_j = \widetilde{\mathbf{F}}_j \boldsymbol{s}_{j+1} \tag{2}$$

The main objective of $\widetilde{\mathbf{F}}_j$ is to lower the variance in $\boldsymbol{s}_j$. Section 3 develops the design of orthogonal prefilters.

The information lost in the subsampling process can be recovered from a detail coefficient vector $\boldsymbol{d}_j$ defined as

$$\boldsymbol{d}_j = \mathbf{D}_j^{-1}(\boldsymbol{s}_{j+1} - \mathbf{P}_j \boldsymbol{s}_j). \tag{3}$$

The diagonal matrix $\mathbf{D}_j^{-1}$ can be used for norming or standardisation purposes. We take $\mathbf{D}_j = \mathbf{I}_{j+1}$, the $n_{j+1} \times n_{j+1}$ identity matrix, unless otherwise specified. The operation carried out by the $n_{j+1} \times n_j$ matrix $\mathbf{P}_j$ lies at the heart of the method. The idea is to use $\boldsymbol{s}_j$ in the construction of a prediction of $\boldsymbol{s}_{j+1}$. In most points of $\boldsymbol{x}_{j+1}$, the prediction performs well, which means that the offset vector $\mathbf{D}_j \boldsymbol{d}_j$ or its normalised version $\boldsymbol{d}_j$ is sparse, i.e., it contains many zeros or near-zeros.

In particular, this paper investigates the case where $\mathbf{P}_j \boldsymbol{s}_j$ is the local polynomial estimator, evaluated in the points $\boldsymbol{x}_{j+1}$ and based on the observations $(\boldsymbol{x}_j, \boldsymbol{s}_j)$. This amounts to the definition of a forward multiscale local polynomial transform.

Definition 2 *A multiscale local polynomial transform on a subsampling scheme defined by $\widetilde{\mathbf{J}}_j$, $j = J-1, J-1, \ldots, L$ is an overcomplete transform that maps a vector $\boldsymbol{s}_J$ of length n_j onto the set of vectors $\{\boldsymbol{s}_L, \boldsymbol{d}_L, \ldots, \boldsymbol{d}_{J-1}\}$, defined by (2) and (3), where $\mathbf{P}_j$ is a local polynomial smoothing matrix. This local polynomial smoothing matrix has on its kth the values $P_j(x_{j+1,k}; \boldsymbol{x}_j)$, where $P_j(x; \boldsymbol{x}_j)$ is a vector of length n_j with components depending on variable x, given by*

$$P_j(x; \boldsymbol{x}_j) = \mathrm{X}^{(\widetilde{p})}(x) \left(\mathbf{X}_j^{(\widetilde{p})^T} \mathrm{W}_j(x) \mathbf{X}_j^{(\widetilde{p})} \right)^{-1} \left(\mathbf{X}_j^{(\widetilde{p})^T} \mathrm{W}_j(x) \right). \tag{4}$$

In this expression, $\mathrm{X}^{(\widetilde{p})}(x) = [1\, x\, \ldots\, x^{\widetilde{p}-1}]$ is a row vector of $\widetilde{p}$ power functions, and the integer $\widetilde{p}$ is the order of the prediction. Moreover, the $n_j \times \widetilde{p}$ matrix $\mathbf{X}_j^{(\widetilde{p})}$ has elements

$$\left(\mathbf{X}_j^{(\widetilde{p})} \right)_{kr} = x_{j,k}^{r-1}. \tag{5}$$

Finally, $\mathrm{W}_j(x)$ is a diagonal matrix of weights with elements $(\mathrm{W}_j)_{kk}(x) = K\left(\frac{x - x_{j,k}}{h_j}\right)$. The function $K(x)$ is the kernel function and h_j is the bandwidth at resolution level j.

As in the context of wavelet transforms, the order of prediction $\widetilde{p}$ in (4) is termed the number of dual vanishing moments. The kernel function is assumed to be unimodal and symmetric around the origin. Furthermore, it should decay fast or even be zero outside the interval $[-1, 1]$. The kernel function is rescaled by the bandwidth. The bandwidth fixes for each observation $s_{j+1,k}$ the window of adjacent observations that contribute to the prediction in $x_{j+1,k}$. This window of adjacent observations replaces a multiscale triangulation in a 2D wavelet transform on scattered data [Jansen, 2014]. The lower the value of j, the more points have been taken out at previous, higher resolution levels. Therefore, h_j should increase when j decreases. The bandwidth h_j is thus the user-controlled scale used at resolution level j.

The inverse of (3) is immediate, reconstructing $\boldsymbol{s}_{j+1}$ as

$$\boldsymbol{s}_{j+1} = \mathbf{D}_j \boldsymbol{d}_j + \mathbf{P}_j \boldsymbol{s}_j. \tag{6}$$

As a result, the finest scale data $\boldsymbol{s}_J = \boldsymbol{Y}$ can be reconstructed from one lowest level vector $\boldsymbol{s}_L$ and all intermediate detail vectors $\boldsymbol{d}_j$, with $j = L, L+1, \ldots, J-1$.

2.2 Multiscale refinement and scaling basis functions

For an appropriate design of a multiscale local polynomial decomposition, it is interesting to look at the underlying basis functions that can be associated to the data transformation, just as in a wavelet transform. Let $\{\varphi_{j,k}(x); k = 1, \ldots, n_j\}$ be a set of scaling functions corresponding to the scaling coefficients $s_{j,k}$. Expressions for appropriate basis functions $\varphi_{j,k}(x)$ follow by proceeding to finer scales. Starting off at scale j, the vector of scaling coefficients $\boldsymbol{s}_j$ represents the function

$$f_j(x) = \sum_{k=1}^{n_j} s_{j,k} \varphi_{j,k}(x) = \Phi_j(x) \boldsymbol{s}_j, \tag{7}$$

where $\Phi_j(x)$ is the row vector containing the scaling functions as its elements. Imposing that $f_j(x)$ can also be decomposed at a finer scale, we can write

$$f_j(x) = \sum_{k=1}^{n_{j+1}} s_{j+1,k}\varphi_{j+1,k}(x) = \Phi_{j+1}(x)\boldsymbol{s}_{j+1},$$

where the fine scale coefficients follow from (6), taking $\boldsymbol{d}_j = \mathbf{0}_j$. In particular, if we start off with a canonical vector for $\boldsymbol{s}_j$, or replace it by the identity matrix, we find a recursive definition for the associated scaling functions

$$\Phi_j(x) = \Phi_{j+1}(x)\mathbf{P}_j. \tag{8}$$

At the finest scale J, we can take the scaling function $\varphi_{J,k}(x)$ to be characteristic functions on $[(x_{J,k-1} + x_{J,k})/2, (x_{J,k} + x_{J,k+1})/2]$. By further refinement of the grid of covariates, i.e., $J \to \infty$, all scaling functions can be defined up to an arbitrarily fine scale. This process of refinement is known as subdivision. For use in practice, Expression (8) can be thus be seen as a formal, inversely recursive definition of the basis functions associated to a given prediction operation $\mathbf{P}_j$. In general, there exists no closed form for $\varphi_{J,k}(x)$. In particular, the scaling function does not coincide with the kernel function used in the local polynomial prediction in $\mathbf{P}_j$.

In a similar way we can associate basis functions with the detail offsets $\boldsymbol{d}_j$, which is

$$\Psi_j(x) = \Phi_{j+1}(x)\mathbf{D}_j, \tag{9}$$

allowing us to interpret an inverse transform (6) with nonzero $\boldsymbol{d}_j$ as the general refinement

$$\Phi_{j+1}(x)\boldsymbol{s}_{j+1} = \Phi_j(x)\boldsymbol{s}_j + \Psi_j(x)\boldsymbol{d}_j. \tag{10}$$

2.3 Properties of the Multiscale Local Polynomial Transform

Because at each scale the size of the detail coefficient vector $\boldsymbol{d}_j$ equals the size of the fine scale approximation vector $\boldsymbol{s}_{j+1}$, the total number of coefficients used in the reconstruction equals $\#\{s_{L,k}\}+\#\{d_{j,k}, j = L, \ldots, J-1\} = n_L+\sum_{j=L}^{J-1} n_{j+1} = \sum_{j=L}^{J} \left\lceil n/2^{J-j} \right\rceil = \mathcal{O}(2n)$. The decomposition (3) of $\boldsymbol{Y} = \boldsymbol{s}_J$ into $[\boldsymbol{s}_L; \boldsymbol{d}_{L,\ldots,J-1}]$ is thus expansive or overcomplete, meaning that (6) is not the only possible reconstruction. Alternatives for (6) would lead to alternative functions in $\Phi_j(x)$ and $\Psi_j(x)$.

Another effect of the overcompleteness is that the functions in $\Psi_j(x)$ and in $\Phi_j(x)$ together do not constitute a basis. Both sets $\Psi_j(x)$ and $\Phi_j(x)$ separately are bases of the function spaces they generate. Together they generate the same space as $\Phi_{j+1}(x)$, as can be seen from (10). As $\Phi_{j+1}(x)$ has less functions than $\Psi_j(x)$ and $\Phi_j(x)$ combined, $\Psi_j(x)$ and $\Phi_j(x)$ contain dependent functions.

The overcompleteness is one of the features that distinguishes the Multiscale Local Polynomial Transform from a Fast Wavelet Transform, sometimes referred to as the critically downsampled wavelet transform. In a Fast Wavelet Transform the number of coefficients at the output equals the size of the input data. Fast Wavelet Transforms can be constructed on top of a scheme that is similar to (3). The difference is that the offsets are computed only in the complementary index set $e'(j+1) = \{1, \ldots, n_{j+1}\} \backslash e(j+1)$,

$$\boldsymbol{d}_j = \mathbf{D}_j^{-1}(\boldsymbol{s}_{j+1,e'} - \mathbf{P}_j \boldsymbol{s}_j), \tag{11}$$

where $\mathbf{P}_j$ is now a matrix of size $(n_{j+1} - n_j) \times n_j$.

In some applications and with specific choices of $\mathbf{P}_j$, decompositions as in (3) are known as Laplacian pyramids [Burt and Adelson, 1983]. Laplacian pyramids are less redundant than the nondecimated version of a wavelet transform, also known as cycle spinning transform, stationary transform, translation invariant transform, à trous or maximum overlap transform. The nondecimated version of the wavelet transform has an output of size $(J - L + 1) \cdot n = \mathcal{O}(n \log_2(n))$.

A Laplacian pyramid shares some of the benefits of a nondecimated wavelet transform, in particular the smoothing effect of a reconstruction from an overcomplete representation [Jansen, 2014]. The main motivation for using the Laplacian pyramid in the context of this paper, is that critical downsampling may lead to unsmooth, fractal-like reconstructions. Indeed, for a smooth reconstruction it is necessary that the prediction in a point $x_{j+1,\ell}$ with $\ell \in e'(j+1)$ tends to the value in the adjacent point $x_{j+1,k}$ with $k \in e(j+1)$ when $x_{j+1,\ell}$ tends to $x_{j+1,k}$. This continuity condition can be formulated as [Jansen, 2013]

$$\lim_{u \to x_{j,k}} P_j(u; \boldsymbol{x}_j) \cdot \boldsymbol{s}_j = s_{j,k}, \tag{12}$$

for arbitrary $\boldsymbol{s}_j$. A matrix $P_j(u; \boldsymbol{x}_j)$ that is constructed as a local polynomial smoothing on $\boldsymbol{x}_j$ does not satisfy the condition. In wavelet decompositions, fractal-like reconstructions are avoided because the prediction $\mathbf{P}_j$ is either constructed as an interpolation between the elements in $e(j+1)$, or it is just one operation in a series of lifting steps that together define the offset $\boldsymbol{d}_j$. In that case, the wavelet decomposition takes the form of an iterated filterbank

$$\boldsymbol{s}_j = \widetilde{\mathbf{H}}_j^T \boldsymbol{s}_{j+1} \tag{13}$$

$$\boldsymbol{d}_j = \widetilde{\mathbf{G}}_j^T \boldsymbol{s}_{j+1}, \tag{14}$$

while the reconstruction is

$$\boldsymbol{s}_{j+1} = \mathbf{H}_j \boldsymbol{s}_j + \mathbf{G}_j \boldsymbol{d}_j, \tag{15}$$

where $\widetilde{\mathbf{H}}_j$ and $\mathbf{H}_j$ are $n_{j+1} \times n_j$ matrices, while $\widetilde{\mathbf{G}}_j^T$ and $\mathbf{G}_j$ have size $(n_{j+1} - n_j) \times n_j$. In the general filterbank form, the coarsening through $\widetilde{\mathbf{H}}_j^T$ and the refinement through $\mathbf{H}_j$ cannot be the same smoothing operation. As a result, the

redundant scheme of a Laplacian pyramid can also be seen as a framework to construct multiscale decompositions where both forward and inverse transforms are based on a single smoothing operation, $\mathbf{P}_j$, and where the reconstruction does not suffer from fractal effects.

Yet another effect of the redundancy in a Laplacian pyramid is that the inverse transform in (6) does not depend on the prefilter (2), which illustrates the flexibility in the design of the prefilter. The inverse transform is not unique, the reconstruction in (6) being just one of the possible solutions. Other reconstructions may depend on the prefilter [Do and Vetterli, 2003, Jansen, 2013].

An interesting comparison between the Fast Wavelet Transform, the Nondecimated Wavelet Transform and the Multiscale Local Polynomial Transform follows from the next argument, making clear that the Multiscale Local Polynomial Transform has a smoothing effect beyond the reconstruction from an overcomplete representation. Given the wavelet filters in (13), (14), and (15), consider the operation that first refines $\boldsymbol{s}_j$ without adding any details (i.e., $\boldsymbol{d}_j = \mathbf{0}_j$) and then decomposes back in to coarse scaling and detail coefficients. This operation is given by the following two expressions.

$$\begin{aligned} \boldsymbol{s}'_j &= \widetilde{\mathbf{H}}_j^T \mathbf{H}_j \boldsymbol{s}_j, \\ \boldsymbol{d}'_j &= \widetilde{\mathbf{G}}_j^T \mathbf{H}_j \boldsymbol{s}_j. \end{aligned}$$

Since the critically downsampled wavelet transform and its inverse verify the perfect reconstruction property that

$$\begin{bmatrix} \widetilde{\mathbf{H}}_j^T \\ \widetilde{\mathbf{G}}_j^T \end{bmatrix} \begin{bmatrix} \mathbf{H}_j & \mathbf{G}_j \end{bmatrix} = \mathbf{I}_{j+1} = \begin{bmatrix} \mathbf{H}_j & \mathbf{G}_j \end{bmatrix} \begin{bmatrix} \widetilde{\mathbf{H}}_j^T \\ \widetilde{\mathbf{G}}_j^T \end{bmatrix},$$

it follows that $\boldsymbol{s}'_j = \boldsymbol{s}_j$ and $\boldsymbol{d}'_j = \boldsymbol{d}_j = \mathbf{0}_j$, which implies that $\widetilde{\mathbf{H}}_j^T \mathbf{H}_j = \mathbf{I}_j$.

In the case of a nondecimated wavelet transform, $\widetilde{\mathbf{H}}_j^T \mathbf{H}_j$ becomes a — possibly non-orthogonal — projection. Although it is no longer the identity matrix, repeated application of refinement and decomposition has a one-time effect only. This is not the case in a multiscale local polynomial decomposition, where decomposition after refinement is given by $\widetilde{\mathbf{F}}_j^T \mathbf{P}_j$. This operation has a smoothing effect, even when applied repeatedly, except for inputs $\boldsymbol{s}_j$ that are exactly polynomial.

3 Orthogonal prefilters

The local polynomial prediction is at the heart of the refinement scheme (8). The choices of the bandwidths and the degree of the polynomials determine the properties of the basis functions $\Phi_j(x)$ in (7), and hence, these parameters fix the smoothness characteristics of any reconstruction. The design of the prediction operation takes the irregularity of the vector of covariates, $\boldsymbol{x}_{j+1}$ into account. The scale of the prediction operation is controlled by the explicit choice of the bandwidth.

All of this is in contrast to the design of the prefilter in (2). The prefilter plays no role in the reconstruction, and therefore it has no influence on its smoothness. Instead, its main role is to reduce the variance of the scaling coefficients when proceeding from fine to coarse scales. To this end, there is no need to include a bandwidth in the definition of the prefilter. The idea is that when n_{j+1} coefficients in $\boldsymbol{s}_{j+1}$ are approximated by n_j coefficients in $\boldsymbol{s}_j$, then each coarse scale coefficient should more or less represent an average of n_{j+1}/n_j fine scale coefficients. The ambition is thus to obtain a variance reduction so that $\mathrm{var}(s_{j,k}) = (n_j/n_{j+1})\mathrm{var}(s_{j+1,2k})$, at least when all $s_{j+1,k}, k = 1, \ldots n_{j+1}$ are uncorrelated. In the particular case of dyadic subsampling, we impose that $\mathrm{var}(s_{j,k}) = \mathrm{var}(s_{j+1,2k})/2$, except possibly for coefficients at the end points.

Moreover, we impose that the rows of the prefilter matrix $\widetilde{\mathbf{F}}_j$ are orthogonal to each other. With $\mathbf{I}_j$ the $n_j \times n_j$ identity matrix, this is

$$\widetilde{\mathbf{F}}_j \widetilde{\mathbf{F}}_j^T = \left(\frac{n_j}{n_{j+1}} \right) \mathbf{I}_j = \mathbf{I}_j/2, \tag{16}$$

the last equality holding for the special case of dyadic subsampling. The orthogonality ensures that homoscedastic uncorrelated fine scale coefficients are transformed into homoscedastic uncorrelated coarse scale coefficients. This is the easiest way to control the covariance structure at all scales. It reduces the computation time for the covariance structure of the detail coefficients $\boldsymbol{D}_j$ at scale j. This structure only depends on the prediction operation at that scale, not on any preceding prefiltering.

A second objective for the prefilter is that it should preserve polynomials up to scale $\widetilde{p} - 1$, where $\widetilde{p}$ is the polynomial order of the prediction operation, as defined in (4). With $\mathbf{X}_j^{(\widetilde{p})}$ defined in (5), this condition can be expressed as

$$\widetilde{\mathbf{F}}_j \mathbf{X}_{j+1}^{(\widetilde{p})} = \mathbf{X}_j^{(\widetilde{p})} = \widetilde{\mathbf{J}}_j \mathbf{X}_{j+1}^{(\widetilde{p})}. \tag{17}$$

Polynomial reproduction and orthogonality induce a limit on the variance reduction, as stated by the following result.

Lemma 1 *Any prefilter $\widetilde{\mathbf{F}}_j$ for which $\widetilde{\mathbf{F}}_j \widetilde{\mathbf{F}}_j^T = \widetilde{\mathbf{D}}_j$, with $\widetilde{\mathbf{D}}_j$ a diagonal matrix and for which $\widetilde{\mathbf{F}}_j \mathbf{1}_{j+1} = \mathbf{1}_j$, where $\mathbf{1}_j$ is a vector with all n_j entries equal to 1, satisfies*

$$\mathrm{Tr}(\widetilde{\mathbf{D}}_j)/n_j \geq n_j/n_{j+1}, \tag{18}$$

where $\mathrm{Tr}(\mathbf{A})$ denotes the trace of a square matrix $\mathbf{A}$. In particular, if $\widetilde{\mathbf{D}}_j = \gamma \mathbf{I}_j$, then $\gamma \geq n_j/n_{j+1}$, meaning that the variance reduction cannot exceed the subsampling rate.

Proof.
We have $\mathrm{Tr}(\widetilde{\mathbf{D}}_j) = \mathrm{Tr}(\widetilde{\mathbf{D}}_j \mathbf{1}_j \mathbf{1}_j^T) = \mathrm{Tr}(\mathbf{1}_j^T \widetilde{\mathbf{D}}_j \mathbf{1}_j) = \mathbf{1}_j^T \widetilde{\mathbf{D}}_j \mathbf{1}_j = \mathbf{1}_j^T \widetilde{\mathbf{F}}_j \widetilde{\mathbf{F}}_j^T \mathbf{1}_j = \|\widetilde{\mathbf{F}}_j^T \mathbf{1}_j\|_2^2$. So, we are minimising the 2-norm of a n_{j+1} sized vector, $\widetilde{\mathbf{F}}_j^T \mathbf{1}_j$. Using

the constant reproduction (i.e., the first vanishing moment), its components add up to $\mathbf{1}_{j+1}^T \widetilde{\mathbf{F}}_j^T \mathbf{1}_j = \mathbf{1}_j^T \mathbf{1}_j = n_j$, thereby defining a constrained optimisation problem, whose solution is obtained by taking the same value for all components $\widetilde{\mathbf{F}}_j^T \mathbf{1}_j = \mathbf{1}_{j+1}(n_j/n_{j+1})$. This vector has the minimum norm as stated in the Lemma. □

The result in Lemma 1 states that the objective in (16) could be feasible. It does, however, not guarantee the existence of a prefilter satisfying (16).

Moreover, a good prefilter should also satisfy a third objective, which is that the matrix $\widetilde{\mathbf{F}}_j$ should show a sort of band structure or diagonal dominance. Indeed, a coarse scale coefficient $s_{j,k}$ should get input from fine scale coefficients situated near $x_{j,k} = x_{1+1,2k}$. The matrix $\widetilde{\mathbf{F}}_j$ should therefore be "close" to simple subsampling. The following result puts a limit to the feasibility of this objective. More precisely, it provides a lower bound on the number of nonzeros at each row of $\widetilde{\mathbf{F}}_j$.

Lemma 2 *Let $\boldsymbol{x}_j = \widetilde{\mathbf{J}}_j \boldsymbol{x}_{j+1}$ be an even-odd subsampling operation and let $\widetilde{\mathbf{F}}_j$ be an orthogonal prefilter satisfying (16) and (17), so that $(\widetilde{\mathbf{F}}_j)_{km} = 0$ if $m \notin \{2k - l + 1, \ldots, 2k + r\}$, then $r - l \geq 2\widetilde{p} + 4$, unless symmetry in the covariate values allows some of the nonzeros to vanish.*

Proof. See Appendix B.

The Appendix B also explores the exceptional case of equidistant covariates, where all rows of $\widetilde{\mathbf{F}}_j$ are all translations of each other with just $2\widetilde{p}$ nonzero elements. The elements do not depend on level j, and can be found by solving the system of linear and nonlinear equations

$$\sum_{k=-\widetilde{p}+1}^{\widetilde{p}} \widetilde{F}_k^2 = 1/2 \tag{19}$$

$$\sum_{k=-\widetilde{p}+1}^{\widetilde{p}} \widetilde{F}_k \widetilde{F}_{k+2s} = 0 \text{ for } s \in \{1, \ldots \widetilde{p} - 1\} \tag{20}$$

$$\sum_{k=-\widetilde{p}+1}^{\widetilde{p}} \widetilde{F}_k = 1 \tag{21}$$

$$\sum_{k=-\widetilde{p}+1}^{\widetilde{p}} k^s \widetilde{F}_k = 0 \text{ for } s \in \{1, \ldots \widetilde{p} - 1\}. \tag{22}$$

These prefilters are closely related to the Daubechies orthogonal wavelet filters [Daubechies, 1992]. The design for these wavelet filters is, however, based on vanishing moments of the corresponding mother wavelet basis function. This is in contrast to the vanishing moment conditions in (22), which are stated in terms of polynomials evaluated in the equidistant covariate values, without any link to the basis functions. In proper wavelet analyses, it is not possible to work on covariate values without taking the basis functions into account. This could induce errors termed "the wavelet crime" [Strang and Nguyen, 1996].

We now develop conditions (16) and (17), checking the existence of an orthogonal, polynomial preserving prefilter with the best possible variance reduction. Since expression (17) can be read as $\widetilde{\mathbf{F}}_j - \widetilde{\mathbf{J}}_j$ being in the left null space of $\mathbf{X}_{j+1}^{(\widetilde{p})}$, we can write

$$\widetilde{\mathbf{F}}_j - \widetilde{\mathbf{J}}_j = \widetilde{\mathbf{U}}_j \widetilde{\mathbf{V}}_j, \tag{23}$$

where the rows of the $(n_{j+1} - \widetilde{p}) \times n_{j+1}$ matrix $\widetilde{\mathbf{V}}_j$ constitute an orthogonal basis for the left null space of $\mathbf{X}_{j+1}^{(\widetilde{p})}$, and $\widetilde{\mathbf{U}}_j$ is an unknown $n_j \times (n_{j+1} - \widetilde{p})$ matrix, taken so that $\widetilde{\mathbf{F}}_j$ has orthogonal rows as in (16). Since the maximum variance reduction in (16) is not guaranteed, we impose for the moment that $\widetilde{\mathbf{F}}_j \widetilde{\mathbf{F}}_j^T = \gamma_j \mathbf{I}_j$, hoping to find a matrix $\widetilde{\mathbf{U}}_j$ for a value γ_j as close as possible to $n_j/n_{j+1} = 1/2$. Using $\mathbf{I}_j = \widetilde{\mathbf{J}}_j \widetilde{\mathbf{J}}_j^T$, the orthogonality condition, $(\widetilde{\mathbf{U}}_j \widetilde{\mathbf{V}}_j + \widetilde{\mathbf{J}}_j)(\widetilde{\mathbf{U}}_j \widetilde{\mathbf{V}}_j + \widetilde{\mathbf{J}}_j)^T = \gamma_j \mathbf{I}_j$, can be developed as

$$(\widetilde{\mathbf{U}}_j + \widetilde{\mathbf{J}}_j \widetilde{\mathbf{V}}_j^T)(\widetilde{\mathbf{U}}_j + \widetilde{\mathbf{J}}_j \widetilde{\mathbf{V}}_j^T)^T = \widetilde{\mathbf{J}}_j \widetilde{\mathbf{V}}_j^T \widetilde{\mathbf{V}}_j \widetilde{\mathbf{J}}_j^T - (1 - \gamma_j)\mathbf{I}_j.$$

For $\gamma_j = 1$, this system has a trivial solution $\widetilde{\mathbf{U}}_j = \mathbf{0}$, i.e., $\widetilde{\mathbf{F}}_j = \widetilde{\mathbf{J}}_j$. Otherwise, for $\frac{1}{2} \leq \gamma_j \leq 1$, that is, the right hand side, being independent from the choice of the orthogonal basis $\widetilde{\mathbf{V}}_j$, can be factorised as

$$\widetilde{\mathbf{J}}_j \widetilde{\mathbf{V}}_j^T \widetilde{\mathbf{V}}_j \widetilde{\mathbf{J}}_j^T - (1 - \gamma_j)\mathbf{I}_j = \widetilde{\mathbf{E}}_j \left(\widetilde{\mathbf{\Lambda}}_j - (1 - \gamma_j)\mathbf{I}_j \right) \widetilde{\mathbf{E}}_j^T,$$

where $\widetilde{\mathbf{\Lambda}}_j$ is a diagonal matrix containing the eigenvalues of $\widetilde{\mathbf{J}}_j \widetilde{\mathbf{V}}_j^T \widetilde{\mathbf{V}}_j \widetilde{\mathbf{J}}_j^T$ while $\widetilde{\mathbf{E}}_j$ has the corresponding eigenvectors as its columns. Let $\widetilde{\lambda}_{j,\min}$ be the smallest element on the diagonal of $\widetilde{\mathbf{\Lambda}}_j$ and let $\gamma_j > 1 - \widetilde{\lambda}_{j,\min}$, then we can define the real matrix

$$\widetilde{\mathbf{S}}_j = \widetilde{\mathbf{E}}_j \left(\widetilde{\mathbf{\Lambda}}_j - (1 - \gamma_j)\mathbf{I}_j \right)^{1/2} \widetilde{\mathbf{E}}_j^T. \tag{24}$$

Then any solution $\widetilde{\mathbf{U}}_j$ in $\widetilde{\mathbf{F}}_j = \widetilde{\mathbf{U}}_j \widetilde{\mathbf{V}}_j + \widetilde{\mathbf{J}}_j$ can be written as

$$\widetilde{\mathbf{U}}_j = \widetilde{\mathbf{S}}_j \widetilde{\mathbf{Q}}_j - \widetilde{\mathbf{J}}_j \widetilde{\mathbf{V}}_j^T, \tag{25}$$

where $\widetilde{\mathbf{Q}}_j$ is a $n_j \times (n_{j+1} - \widetilde{p})$ matrix with orthogonal rows.

When $\widetilde{p} = 2$, the minimum value of γ_j is close to $1/2$, as follows from the subsequent result.

Lemma 3 *Given a vector of n_{j+1} covariates $\boldsymbol{x}_{j+1}$. Let $\boldsymbol{x}_j = \widetilde{\mathbf{J}}_j \boldsymbol{x}_{j+1}$ be a subsampled version, where $x_{j,k} = x_{j+1,2k+2}$, for $k = 0, \ldots, n_j$ and $n_{j+1} > 2n_j + 1$, meaning that the first and the last elements of $\boldsymbol{x}_{j+1}$ are not in $\boldsymbol{x}_j$. Then, with $\widetilde{p} = 2$, define*

$$\xi_j = (\overline{\boldsymbol{x}}_j - \overline{\boldsymbol{x}}_{j+1})/(\overline{\boldsymbol{x}^2}_{j+1} - \overline{\boldsymbol{x}}_{j+1}^2)^{1/2}, \tag{26}$$

and

$$\zeta_j = 1 - \overline{(\boldsymbol{x}_j - \overline{\boldsymbol{x}}_{j+1})^2}/(\overline{\boldsymbol{x}^2}_{j+1} - \overline{\boldsymbol{x}}_{j+1}^2). \tag{27}$$

Taking

$$\gamma_j \geq \frac{n_j}{n_{j+1}} \left[1 + \frac{1}{2} \left(\sqrt{\zeta_j^2 + 4\xi_j^2} - \zeta_j \right) \right], \tag{28}$$

all elements of the diagonal matrix $\widetilde{\mathbf{\Lambda}}_j$ in (24) are positive, so there exists a real matrix $\widetilde{\mathbf{S}}_j$ satisfying (24).

Proof. See Appendix C.

We thus find that if the prefilter preserves constant and linear functions, and if points on the boundary are left out from the variance reduction, then the optimal variance reduction can be nearly attained. Indeed, the values of ξ_j and ζ_j are typically close to zero, as can be verified empirically, leading to γ_j just a bit above n_j/n_{j+1}.

The result of Lemma 3 comes without any guarantee about diagonal dominance as suggested in Lemma 2. Currently, we have no algorithm that finds a bandmatrix with orthogonal rows for use as variance reducing prefilter, or even a procedure that finds out if such a bandmatrix exists for a given variance reduction γ. Instead, we propose to construct $\widetilde{\mathbf{Q}}_j$ in (25) as $\widetilde{\mathbf{Q}}_j = \widetilde{\mathbf{Q}}_j^{[0]} \cdot \widetilde{\mathbf{Q}}_j^{[1]} \widetilde{\mathbf{Q}}_j^{[2]} \widetilde{\mathbf{Q}}_j^{[3]} \ldots$, where $\widetilde{\mathbf{Q}}_j^{[0]}$ is an $n_j \times (n_{j+1} - \widetilde{p})$ matrix with orthogonal rows and $\widetilde{\mathbf{Q}}_j^{[i]}$ are elementary orthogonal matrices of size $(n_{j+1} - \widetilde{p}) \times (n_{j+1} - \widetilde{p})$. These elementary operations transforms are typically Givens rotations or Householder reflections, chosen to minimise $\|\widetilde{U}_j^{[i]}\|_F^2$, where $\widetilde{U}_j^{[i]} = \widetilde{\mathbf{S}}_j^{[i-1]} \widetilde{\mathbf{Q}}_j^{[i]} - \widetilde{\mathbf{J}}_j \widetilde{\mathbf{V}}_j^T$, and $\widetilde{\mathbf{S}}_j^{[i-1]} = \widetilde{\mathbf{S}}_j \widetilde{\mathbf{Q}}_j^{[0]} \widetilde{\mathbf{Q}}_j^{[1]} \ldots \widetilde{\mathbf{Q}}_j^{[i-1]}$. More details about the construction of $\widetilde{U}_j^{[i]}$ can be found in Appendix A. In our experiments, the resulting prefilters do not have a band structure, but most entries away from the diagonal are close to zero, while the band with large entries is even narrower than that of a true band matrix in Lemma 2. Unfortunately, the design and application of the prefilter require a bit more computations than a band matrix multiplication.

4 The choice of the bandwidth

4.1 Random model for multiscale local polynomial offsets

As mentioned in Section 1, the design of $\mathbf{P}_j$ is based on a data smoothing technique, but $\mathbf{P}_j$ itself is a mere linear data transform that does not perform any smoothing or estimation. The estimation takes place after the data transform, mostly combining linear and nonlinear operations. The nonlinear operation is often a thresholding rule or any other form of coefficient selection applied to the detail coefficients at the fine resolution levels. The linear part of the processing mostly operates on a coarse resolution level L or below. In most approaches, the linear operation is simply the identity.

The bandwidth at a fine scale, i.e., for j between L and $J-1$, should be chosen so that the detail coefficients at scale j are most ready for nonlinear processing, in particular for thresholding: the small coefficients should be as small as possible

while the large coefficients should be as sparse and as large as possible, carrying all the essential information for a good approximation or estimation of the signal $f(x)$. The framework developed in this paper holds for data both with and without errors.

More precisely, we assume that for any choice of the bandwidth h_j, the resulting noise-free detail offsets $d_{j,k}$ can be considered as possibly dependent observations from a mixture of two random variables. The mixture is denoted by D_j. The model takes the form $D_j = M_j D_{j;1} + (1 - M_j) D_{j;0}$, where $D_{j;1}$ is a random model for the large coefficients, and $D_{j;0}$ is a random model for the small coefficients. The large coefficients are described by a double exponential law, $D_{j;1} \sim \text{Laplace}(a_j)$. For the small coefficients we use a normal model $D_{j;0} \sim N(0, \tau_j^2)$. The motivation for this model is that the normal distribution is stable under linear combinations. This property corresponds to the observation that small coefficients come from the class of Lipschitz smooth functions, which is preserved under linear combinations. Furthermore, the magnitude of the small coefficients depends primarily on the bandwidth, not on the local design of covariates $\boldsymbol{X}_j$ [Fan and Gijbels, 1996]. Hence, there is no need to model any dependence of $D_{j;0}$ on $\boldsymbol{X}_j$.

The random model also includes $p_j = P(M_j = 1)$, and we suppose that the label M_j is independent from $D_{j;1}$ and from $D_{j;0}$. As definition for the Laplace or double exponential density function, we adopt $f_{D_{j;1}}(d; a_j) = (a_j/2)e^{-a_j|d|}$, for $d \in \mathbb{R}$, so that $1/a_j = E(|D_{j;1}|)$.

Unlike the model for the small and large noise-free signal coefficients in $D_{j;0}$ and $D_{j;1}$, the model for the errors depends on the design points $\boldsymbol{X}$. Indeed, let $\boldsymbol{Z}$ be the vector of the independent, homoscedastic errors in the model (1). Furthermore, let $\mathbf{T}_j$ be the matrix that maps the vector of observations onto the detail offsets at scale j of a multiscale local polynomial transform, i.e.,

$$\mathbf{T}_j = \mathbf{D}_j^{-1}(\mathbf{I}_{j+1} - \mathbf{P}_j\widetilde{\mathbf{F}}_j)\widetilde{\mathbf{F}}_{j+1}\ldots\widetilde{\mathbf{F}}_{J-1}, \tag{29}$$

so that $\boldsymbol{d}_j = \mathbf{T}_j \boldsymbol{s}_J$. Then the covariance matrix of the transformed errors at scale j equals $\text{cov}(\mathbf{T}_j\boldsymbol{Z}|\boldsymbol{X}) = \mathbf{T}_j\mathbf{T}_j^T\sigma^2$. In this expression, $\mathbf{T}_j$ depends on the design $\boldsymbol{X}$. The model for the coefficients with errors is then

$$\widetilde{D}_{j,k}|\boldsymbol{X} = D_{j,k} + \sigma_{j,k}Z_{j,k},$$

where $Z_{j,k}$ is a standard normal random variable and $\sigma_{j,k} = \sigma\kappa_{j,k}$ with

$$\kappa_{j,k} = \sqrt{(\mathbf{T}_j\mathbf{T}_j^T)_{k,k}}. \tag{30}$$

The density of the coefficients with errors can be written as $f_{\widetilde{D}_{j,k}|\boldsymbol{X}}(d; p_j, a_j, \tau_j, \sigma_{j,k}) = p_j g(d; a_j, \sigma_{j,k}) + (1 - p_j)\phi(d/\epsilon_{j,k})/\epsilon_{j,k}$. In this expression, $\phi(x)$ is the standard normal density, $\epsilon_{j,k} = \sqrt{\tau_j^2 + \sigma_{j,k}^2}$, and $g(x; a, \sigma)$ is the convolution of the normal and Laplacian densities, which is

$$g(d; a, \sigma) \quad = \quad f_{\widetilde{D}_{j;1}}(d; a, \sigma) \tag{31}$$

$$= \frac{a}{2}\phi\left(\frac{d}{\sigma}\right)\left[\frac{1-\Phi\left(\frac{d}{\sigma}+\sigma a\right)}{\phi\left(\frac{d}{\sigma}+\sigma a\right)}+\frac{\Phi\left(\frac{d}{\sigma}-\sigma a\right)}{\phi\left(\frac{d}{\sigma}-\sigma a\right)}\right],$$

where $\Phi(x)$ is the standard normal distribution function. The corresponding cumulative distribution function equals $G(d; a_j, \sigma_{j,k})$, where

$$\begin{aligned} G(d;a,\sigma) &= F_{\widetilde{D}_{j;m}}(d;a,\sigma) \qquad (32)\\ &= \Phi\left(\frac{d}{\sigma}\right)+\frac{1}{2}\phi\left(\frac{d}{\sigma}\right)\left[\frac{1-\Phi\left(\frac{d}{\sigma}+\sigma a\right)}{\phi\left(\frac{d}{\sigma}+\sigma a\right)}-\frac{\Phi\left(\frac{d}{\sigma}-\sigma a\right)}{\phi\left(\frac{d}{\sigma}-\sigma a\right)}\right]. \end{aligned}$$

The parameters of the mixture model, p_j, a_j, τ_j and σ allow us to model a certain degree of sparsity. The forthcoming application of this model will assume that some of these parameters depend on the sample size n. Indeed, as the number of observations $f(x_i)+\sigma Z_i$ grows, while the number of singularities in $f(x)$ remains the same, it can be expected that the proportion of large offsets at scale j, p_j tends to zero. A typical behavior is

$$p_j = \mathcal{O}\left(\log(n)/n\right). \qquad (33)$$

At the same time, the few large offsets become more prominent. Indeed, as n grows larger, so does the finest resolution level J in (29). Hence the number of prefilters $\widetilde{\mathbf{F}}_i$ with $j \leq i < J$ increases, each having a variance reducing effect as in (16) on uncorrelated, homoscedastic errors. For $\boldsymbol{Z}$ i.i.d., we find that $\mathrm{var}\left((\mathbf{T}_j\boldsymbol{Z})_k\right) = (n_j/n_J)\mathrm{var}(Z_i)$, hence, using dyadic subsampling and with $\mathbf{D}_j = \mathbf{I}_{j+1}$ independently from n,

$$\mathrm{var}(\widetilde{D}_{j,k}|D_{j,k}) = \sigma_{j,k}^2 = \mathcal{O}\left(2^{j-J}\right).$$

This result states that $J-j$ should be maximised in order to reduce the variance of the errors as much as possible. In other words, the scale at which $f(x)$ is observed should be as fine as possible.

On the other hand, if $f(x)$ is Lipschitz ν continuous with $\nu \geq \widetilde{p}$, then the detail offsets at scale j from the observations $f(x_{J,k})$ have an order of magnitude $|d_{j,k}| = \mathcal{O}\left(h_j^{\widetilde{p}}\right)$, again taking $\mathbf{D}_j = \mathbf{I}_{j+1}$. This result is independent from J and independent from the grid of covariate values. It states that the scale h_j at which the data are processed should be as fine as possible in order to reduce the approximation error to a minimum.

Therefore, the asymptotic analysis will assume that the working scale h_j tends to zero and at the same time that $J-j$ grows (slowly) to infinity, so that h_J tends to zero slightly faster.

Finally, when $f(x)$ contains singularities, one can expect that the offsets near the singularities are of order $|d_{j,k}| = \mathcal{O}(1)$, again if $\mathbf{D}_j = \mathbf{I}_{j+1}$. From here on, we will assume that $\mathbf{D}_j$ includes a standardisation of order $\mathcal{O}\left(2^{(J-j)/2}\right)$, so

that $\sigma_{j,k}^2 = \mathcal{O}(1)$, $\tau_j^2 = \mathrm{var}(D_{j;0}) = \mathcal{O}\left(2^{J-j}h_j^{2\widetilde{p}}\right)$ and $1/a_j^2 = \mathrm{var}(D_{j;1}) = \mathcal{O}\left(2^{J-j}\right)$, from which we typically find

$$a_j = \mathcal{O}\left(2^{(J-j)/2}\right). \tag{34}$$

4.2 Maximum likelihood bandwidth

For any value of the bandwidth h_j, the resulting detail offsets are modelled as instances from one member of the mixture distribution (31), with bandwidth dependent values of the parameters p_j, a_j, τ_j and $\sigma_{j,k}$. This approach leads to two questions to deal with. One question is how to use the observed offsets $\boldsymbol{d}_j$ to estimate from which member of the model (31) they come from. The other question is which member of the model (31) performs best for use in nonlinear sparse processing. Both questions will be addressed by a maximum likelihood argument.

As for the issue how to choose among the possible members of the family in (31), several criteria can be established, based on the values of p_j, a_j, τ_j and $\sigma_{j,k}$. For instance, p_j should be as small as possible, because a model with small p_j concentrates all the essential information about $f(x)$ in a limited number of large coefficients. The small coefficient parameter τ_j and the error parameter $\sigma_{j,k}$, should also be as as small as possible.

As an overall criterion for sparsity of description, we use the minimum working independence entropy or, equivalently, the maximum working independence expected log-likelihood of a model. We thus maximise

$$\begin{aligned}&\ell(h_j; p_j, a_j, \tau_j, \sigma) = \\ &\frac{1}{n_{j+1}} \sum_{k=1}^{n_{j+1}} E_{\widetilde{D}_{j,k}|\boldsymbol{X}} \left[\log(f_{\widetilde{D}_{j,k}|\boldsymbol{X}}(\widetilde{D}_{j,k}|\boldsymbol{X}; p_j, a_j, \tau_j, \sigma_{j,k}))\right],\end{aligned} \tag{35}$$

where the three parameters p_j, a_j, τ_j are a function of h_j, while the fourth, $\sigma_{j,k} = \sigma\kappa_{j,k}$ depends on h_j through $\kappa_{j,k}$, but depends too on σ, the standard deviation of the errors, which is not controlled by the maximum expected log-likelihood routine. In (35) the coefficients are evaluated as if they were independent observations. This is not only for reasons of computational complexity. An assessment taking the dependence structures into account would be more tolerant of a group of large coefficients, all linked to a single singularity. The objective is, however, to keep the number of large coefficients limited. Another argument for evaluating every coefficient separately is that, like wavelet transforms, multiscale local polynomial transforms are used for decomposing signals into coefficients that can be further processed separately, for instance by thresholding.

In practice, $\ell(h_j; p_j, a_j, \tau_j, \sigma)$ is estimated by

$$\begin{aligned}&\widehat{\ell}(h_j; \widehat{p}_j, \widehat{a}_j, \widehat{\tau}_j, \widehat{\sigma}) = \\ &\frac{1}{n_{j+1}} \sum_{k=1}^{n_{j+1}} \log(f_{\widetilde{D}_{j,k}|\boldsymbol{X}}(\widetilde{d}_{j,k}(h_j); \widehat{p}_j, \widehat{a}_j, \widehat{\tau}_j, \widehat{\sigma}\kappa_{j,k})).\end{aligned} \tag{36}$$

The estimators $(\widehat{p}_j, \widehat{a}_j, \widehat{\tau}_j, \widehat{\sigma})$ follow from the observations $\boldsymbol{d}_j$, and for this, we also use a maximum likelihood approach, meaning that the estimators maximise

$$\widehat{\ell}(h_j; p_j, a_j, \tau_j, \sigma) = \\ \frac{1}{n_{j+1}} \sum_{k=1}^{n_{j+1}} \log(f_{\widetilde{D}_{j,k}|\boldsymbol{X}}(\widetilde{d}_{j,k}(h_j); p_j, a_j, \tau_j, \sigma\kappa_{j,k})).$$

Remark 1 *The problem in (35) is not a variable or model selection procedure in the classical sense, because the setting does not include a "true model" or data generating process, and no divergence from this true model, such as a Kullback-Leibler distance. Nor is there a balance between model complexity and goodness of fit. Instead, all models in (31) are equal in complexity, and all are assumed to be correct models for a given bandwidth h_j. The best model is the one that has the most "interesting" parameter values, in the sense that it generates, on average, the most structured or sparse coefficients.*

Remark 2 *Due to the double use of the sample likelihood for (1) the estimation of the expected likelihood and (2) for the estimation of the parameters by sample likelihood optimisation, the estimator $\widehat{\ell}(h_j; \widehat{p}_j, \widehat{a}_j, \widehat{\tau}_j, \widehat{\sigma})$ is biased w.r.t. $\ell(h_j; p_j, a_j, \tau_j, \sigma)$. Given the limited number of parameters in the model, four to be precise, the bias is limited and not much influenced by the bandwidth. This can be understood from the analogy with the interpretation of the penalty in Akaike's Information Criterion as a bias correction term.*

4.3 Approximation of the maximum joint likelihood estimators

The 4D maximisation is a computationally complex and ill-conditioned problem. Therefore, we approximate $f_{\widetilde{D}_{j,k}|\boldsymbol{X}}(d; p, a, \tau, \sigma)$ in a way that reduces the optimisation to a trivial task.

To this end, we define an observable label $\widetilde{M}_{j,k} \in \{0, 1\}$ where $\widetilde{M}_{j,k} = 1 \Leftrightarrow |\widetilde{D}_{j,k}| > \lambda_j \epsilon_{j,k}$, with $\lambda_j = \sqrt{2\log(n_{j+1})}$ and, as in Section 4.1, $\epsilon_{j,k} = \sqrt{\tau_j^2 + \sigma_{j,k}^2}$. Furthermore, let $\widetilde{p}_{\lambda;j,k} = P(\widetilde{M}_{j,k} = 1|\boldsymbol{X})$, then

$$\begin{aligned} & f_{\widetilde{D}_{j,k}|\boldsymbol{X}}(d; p_j, a_j, \tau_j, \sigma_j) \\ = \; & \widetilde{p}_{\lambda;j,k} f_{\widetilde{D}_{j,k}|\widetilde{M}_{j,k}=1,\boldsymbol{X}}(d; p_j, a_j, \tau_j, \sigma_j) \\ & + (1 - \widetilde{p}_{\lambda;j,k}) f_{\widetilde{D}_{j,k}|\widetilde{M}_{j,k}=0,\boldsymbol{X}}(d; p_j, a_j, \tau_j, \sigma_j). \end{aligned}$$

The two conditional densities can be approximated. The small coefficients are assumed to be predominantly normal. More precisely,

$$\begin{aligned} f_{\widetilde{D}_{j,k}|\widetilde{M}_{j,k}=0,\boldsymbol{X}}(d; p_j, a_j, \tau_j, \sigma_j) \; & \approx \; \widetilde{f}_0(d; \epsilon_{j,k}, \lambda_j) \\ & = \; \frac{1}{\epsilon_{j,k}} \frac{\phi\left(\frac{d}{\epsilon_{j,k}}\right)}{2\Phi\left(\lambda_j\right) - 1}. \end{aligned}$$

The large coefficients are predominantly Laplacian, so

$$f_{\widetilde{D}_{j,k}|\widetilde{M}_{j,k}=1,\boldsymbol{X}}(d;p_j,a_j,\tau_j,\sigma_j) \approx \widetilde{f}_1(d;a_j,\lambda_j\epsilon_{j,k})$$
$$= \frac{(a_j/2)\exp(-a_j|d|)}{\exp(-a_j\lambda_j\epsilon_{j,k})} = (a_j/2)\exp(-a_j(|d|-\lambda_j\epsilon_{j,k})).$$

Writing

$$\begin{aligned}\widetilde{f}_{\widetilde{D}_{j,k}|\boldsymbol{X}}(x) &= \widetilde{p}_{\lambda;j,k}\widetilde{f}_1(d;a_j,\lambda_j\epsilon_{j,k})I\left(|d|\geq\lambda_j\epsilon_{j,k}\right)\\ &\quad +(1-\widetilde{p}_{\lambda;j,k})\widetilde{f}_0(d;\epsilon_{j,k},\lambda_j)I\left(|d|<\lambda_j\epsilon_{j,k}\right)\end{aligned}$$

we have

$$\begin{aligned}&E_{\widetilde{D}_{j,k}|\boldsymbol{X}}\left[\log(f_{\widetilde{D}_{j,k}|\boldsymbol{X}}(\widetilde{D}_{j,k}|\boldsymbol{X}))\right] = \\ &E_{\widetilde{D}_{j,k}|\boldsymbol{X}}\left[\log(\widetilde{f}_{\widetilde{D}_{j,k}|\boldsymbol{X}}(\widetilde{D}_{j,k}|\boldsymbol{X}))\right] + R(p_j,a_j,\tau_j,\sigma_{j,k}),\end{aligned} \tag{37}$$

where $R(p_j,a_j,\tau_j,\sigma_{j,k})$ is the Kullback-Leibler distance between the true and approximative model. Appendix D investigates the asymptotic behavior of the Kullback-Leibler distance. It finds that when p_j and a_j tend to zero for n growing large, and if we take λ not too large neither too small, then $R(p_j,a_j,\tau_j,\sigma_{j,k})\to 0$, while

$$E_{\widetilde{D}_{j,k}|\boldsymbol{X}}\left[\log(f_{\widetilde{D}_{j,k}|\boldsymbol{X}}(\widetilde{D}_{j,k}|\boldsymbol{X}))\right] \asymp 1.$$

The threshold λ should not be too small, meaning that for $n\to\infty$, $\exp(-\lambda^2/2)/a_j \to 0$. Otherwise, the proportion of falsely selected small coefficients is too large. The threshold should not be too large either, meaning that $a\lambda\to 0$. Otherwise, too many large coefficients are misclassified as small. As the expected magnitude of these coefficients is $1/a_j$, the condition $a\lambda\to 0$ amounts to the statement that λ should grow slower than the average large offset.

These conditions are met when p_j and a_j behave as in (33) and (34) respectively, and when we take

$$\lambda = \sqrt{2\log(n)}. \tag{38}$$

The log-likelihood in (36) can now be approximated by

$$\begin{aligned}&\widehat{\ell}_{\widetilde{f}}(h_j;\widehat{p}_j,\widehat{a}_j,\widehat{\tau}_j,\widehat{\sigma})\\ &= \frac{1}{n_{j+1}}\sum_{k=1}^{n_{j+1}}\log(\widetilde{f}_{\widetilde{D}_{j,k}|\boldsymbol{X}}(\widetilde{d}_{j,k}(h_j);\widehat{p}_j,\widehat{a}_j,\widehat{\tau}_j,\widehat{\sigma}\kappa_{j,k}))\\ &= \frac{1}{n_{j+1}}\sum_{k\in\mathcal{J}_{j;1}}\log(\widehat{p}_{\lambda,j}\widetilde{f}_1(\widetilde{d}_{j,k};\widehat{a}_j,\lambda_j\widehat{\epsilon}_{j,k}))\\ &\quad +\frac{1}{n_{j+1}}\sum_{k\in\mathcal{J}_{j;0}}\log((1-\widehat{p}_{\lambda,j})\widetilde{f}_0(\widetilde{d}_{j,k};\widehat{\epsilon}_{j,k},\lambda_j)).\end{aligned}$$

In this expression we used $\mathcal{J}_{j;m}=\{k\in\{1,\ldots,n_{j+1}\},\widetilde{M}_{j,k}=m\}$.

Writing $\widehat{n}_{j;m}$ for the cardinality of $\mathcal{J}_{j;m}$, the estimators can be substituted by their maximum likelihood values

$$\begin{aligned}
\widehat{p}_{\lambda,j} &= \widehat{n}_{j;1}/n_{j+1}, \\
\widehat{a}_j &= \widehat{n}_{j;1}\left[\sum_{k\in\mathcal{J}_{j;1}}\left(|\widetilde{d}_{j,k}| - \lambda_j\widehat{\epsilon}_{j,k}\right)\right]^{-1} \\
&= \widehat{n}_{j;1}\left[\sum_{k\in\mathcal{J}_{j;1}}\left|\mathrm{ST}(\widetilde{d}_{j,k}, \lambda_j\widehat{\epsilon}_{j,k})\right|\right]^{-1}, \\
\widehat{\epsilon}^2_{j,k} &= \widehat{\tau}^2 + \widehat{\sigma}^2\kappa^2_{j,k},
\end{aligned}$$

where $\mathrm{ST}(x,t) = I\left(|x|>t\right)\mathrm{sign}(x)\left(|x|-t\right)$ is the soft-threshold function. The values of $\widehat{\sigma}^2$ and $\widehat{\tau}$ can be found numerically by maximising

$$\begin{aligned}
&\sum_{k\in\mathcal{J}_{j;0}}\log\left[\widetilde{f}_0\left(\widetilde{d}_{j,k};\sqrt{\widehat{\tau}^2+\widehat{\sigma}^2\kappa^2_{j,k}}\right)\right] \\
&= C - \sum_{k\in\mathcal{J}_{j;0}}\left[\frac{\widetilde{d}^2_{j,k}}{2\left(\widehat{\tau}^2+\widehat{\sigma}^2\kappa^2_{j,k}\right)} + \log\left(\sqrt{\widehat{\tau}^2+\widehat{\sigma}^2\kappa^2_{j,k}}\right)\right],
\end{aligned}$$

with $C = -\left(n_{j+1}-\widehat{n}_{j;1}\right)\log\left(2\left[\Phi\left(\lambda_j\right)-1\right]\sqrt{2\pi}\right)$.

The log-likelihood expression to be optimised as a function of the bandwidth thus becomes, in terms of observed values,

$$\begin{aligned}
\widehat{\ell}_{\widetilde{f}}(h_j;\widehat{p}_j,\widehat{a}_j,\widehat{\tau}_j,\widehat{\sigma}) &= \widehat{p}_j\log\left(\widehat{p}_j\right) + \left(1-\widehat{p}_j\right)\log\left(1-\widehat{p}_j\right) \\
&\quad -\widehat{p}_j\log\left(\frac{2e}{\widehat{n}_{j;1}}\sum_{k\in\mathcal{J}_{j;1}}\left|\mathrm{ST}(\widetilde{d}_{j,k},\lambda_j\widehat{\epsilon}_{j,k})\right|\right) \\
&\quad -\frac{1}{n_{j+1}}\sum_{k\in\mathcal{J}_{j;0}}\left[\widetilde{d}^2_{j,k}/(2\widehat{\epsilon}^2_{j,k}) + \log(\widehat{\epsilon}_{j,k})\right] \\
&\quad -\left(1-\widehat{p}_j\right)\log(2\pi)/2.
\end{aligned} \tag{39}$$

5 Simulation study

5.1 Simulation setup

The random model for noise-free offsets in Section 4.1 allows us to simulate functional data according to this model. More precisely, we generate test functions that are sums of a smooth function and a random blocky function, i.e., $f(x) = f_s(x) + f_b(x)$. First, the covariate values in $\boldsymbol{x}$ are generated as the ordered sample of independent, uniform random variables on $[0,1]$. Second, the block functions are generated by picking at random a fixed number, say b, of uniformly distributed locations $\xi_q \in [0,1]$, together with jump magnitudes η_q and binary values

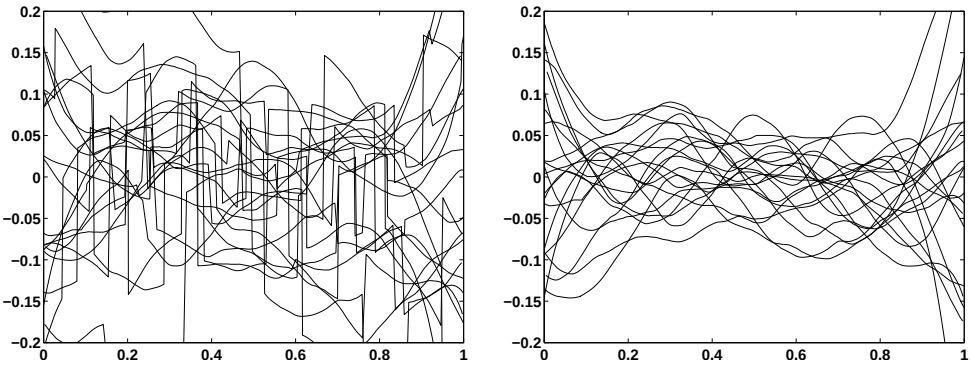

Figure 1: (a) Twenty test functions $f(x)$, generated for use in the simulation study. Each test function is constructed as the sum of a random block $f_b(x)$ and a smooth function $f_s(x)$, depicted in (b).

$T_q \in \{-1, 1\}$, so that

$$f_b(x) = \sum_{q=1}^{b} \eta_{(q)} T_q \chi_{[\xi_{(q)},1]}(x).$$

In this definition, $\chi_A(x)$ denotes the characteristic or indicator function on the set $A \subset [0, 1]$. The magnitudes are uniformly distributed on $[0, \alpha]$, where we take here $\alpha = 0.2$. The magnitudes are sampled independently from each other, but the signs T_q were taken from a binary Markov chain, so that $P(T_q = t|T_{q-1} = t) = 0.1$. This way, the blocks tend to be oscillating, rather than accumulating. Third, the smooth function is generated from a multiscale local polynomial decomposition consisting of $J - L$ levels of detail coefficients $\boldsymbol{d}'_j$, $j = L, \ldots, J - 1$ along with the coarse level scaling coefficients $\boldsymbol{s}'_L$. The latter are taken to be zero, while we set $d'_{j,k} = \tau_j Z_{j,k}$ for $j = L, \ldots, J - 2$, where all $Z_{j,k} \sim N(0, 1)$ are independent, standard normally distributed random variables. Note that we take the finest detail offsets to be zero, i.e., $\boldsymbol{d}'_{J-1} = \mathbf{0}$. This is because we want to avoid small scale, noise- or fractal-like behavior in the reconstruction

$$f_s(x) = \sum_{j=L}^{J-2} \Psi_j(x) \boldsymbol{d}'_j,$$

whose values can be found by the inverse transform, formalised as $f_x(\boldsymbol{x}) = \mathbf{P}_{J-1} \boldsymbol{s}'_{J-1}$, and $\boldsymbol{s}'_{j+1} = \mathbf{P}_j \boldsymbol{s}'_j + \boldsymbol{d}'_j$ for $j = L, \ldots, J - 2$. The detail offsets $\boldsymbol{d}'_j$ are not the eventual smooth offsets for use in the simulation, since the forward multiscale local polynomial transform of $f_s(x)$ will generate different coefficients, as explained in Section 2.3. As a result, the finest details coefficients for use in the simulation will not be zero as is the vector $\boldsymbol{d}'_{J-1}$.

The test functions in Figure 1 have $b = 4$ jumps in $f_b(x)$, while for the construction of $f_s(x)$, the detail coefficients at scale j were sampled from a normal

random variable with $\tau_j = \tau h_{j,0}^{\widetilde{p}}$. This is in accordance with the order of magnitude $|d_{j,k}| = \mathcal{O}\left(h_j^{\widetilde{p}}\right)$ discussed in Section 4.1. The parameter τ is set to one. For reasons explained in Section 5.3, the bandwidths $h_{j,0}$ used in this construction are set to a heuristic value, linking the bandwidth to the subsampling rate.

$$h_{j,0} = h_0 \cdot (x_{j,n_j} - x_{j,1}) \log(n_j)/n_j \approx h_0 \log(n_j)/n_j, \tag{40}$$

where the random smooth function generator takes $h_0 = 4$.

Figure 2 has one of the test functions from Figure 1, this time observed with additive normal errors. The standard deviation of the errors is set to $\sigma = \alpha/\mathrm{SNR}$, with a signal-to-noise ratio SNR equal to 10 throughout the simulations.

5.2 Discussion

The smoothing predictions $\mathbf{P}_j$ at each level j operate independently from each other. Indeed, applying $\mathbf{P}_j$ results in the detail offsets at scale j. These offsets play no role in the calculation of the subsequent coefficients. As a result, the bandwidths h_j can be optimised separately, taking into account the features that are present on the covariate values $\boldsymbol{x}_j$. Figure 3 depicts a plot of the log-likelihood curves (36) in black solid lines and the easy-to-evaluate approximations of (39) in solid grey lines. Both curves are plotted against the bandwidth divided by the heuristic value $h_{j,0}$ in (40), thereby taking $h_0 = 1$. All curves were generated in the framework of a multiscale local linear transform, i.e., taking $\widetilde{p} = 2$ applied to one of the test functions in Figure 1 with $n = 500$ nonequispaced covariate values. The covariate values were drawn uniformly on the interval $[0, 1]$.

From figure 3 we see that (39) performs quite well as approximative and fast evaluation of the likelihood curve. Also, the heuristic bandwidth is close to the maximum likelihood bandwidth. The good performance of the heuristic bandwidth, at least at the fine scales, is confirmed by replicating the experiment of Figure 3 a hundred times, which is summarised in Figure 4. In this figure, the heuristic bandwidths $h_{j,0}$ appear as horizontal, dashed lines, while the results of the bandwidth optimisation at each scale are plotted as line charts. As explained in Section 5.3 below, the heuristic bandwidth is based on an argument of sparsity, showing less relevance at coarse scales. Also for $\widetilde{p}$ larger than two (local quadratic or cubic smoothing), the heuristic bandwidth appears to be suboptimal.

5.3 The heuristic bandwidth

If the number of singularities, b, is small as compared to the size of the covariate vector, n_j, then the prefiltered observations at level j are dominated by long smooth sections. In that case, the optimal bandwidth h_j can be expected to be close to the heuristic value $h_{j,0}$ in (40), where $h_0 = 1$. The heuristic value follows from the requirement that all predictions in $\mathbf{P}_j$ have a sufficient number of evens for the construction of the local polynomials. Let $x_{j+1,2k+d}$, $d \in \{0, 1\}$ be an even

($d = 0$) or odd ($d = 1$) point in which we compute a prediction based on the $N_{j+1,2k+d}$ values of $x_{j,l} = x_{j+1,2l}$ for which $|x_{j,l} - x_{j+1,2k+d}| < h_{j,0}$. Note that if $d = 0$, then $N_{j+1,2k+d} = N_{j+1,2k}$ must be at least one, because $x_{j+1,2k}$ itself is an even point. Furthermore, assume that $x_{j+1,k}$ are ordered observations from independent uniformly random variables, i.e., $x_{j+1,k} = u_{(k:n_{j+1})}$. Then the number of neighbours within distance $h_{j,0}$ is binomial, and thus approximately Poisson with expected value $2n_j h_{j,0}$, from which the probability function of the number of even neighbours follows, for both $d = 0$ and $d = 1$. Writing $\mu_j = n_j h_{j,0}$, straightforward calculations prove that

$$\begin{aligned} P(N_{j+1,2k+1} = 0) &= e^{-2\mu_j}, \\ P(N_{j+1,2k+1} = 1) &= e^{-2\mu_j}(2\mu_j + \mu_j^2), \\ P(N_{j+1,2k} = 1) &= P(N_{j+1,2k+1} = 0) \\ &\quad + P(N_{j+1,2k+1} = 1), \end{aligned}$$

and for any value $r > 1$, $d \in \{0, 1\}$,

$$\begin{aligned} P(N_{j+1,2k+d} = r) = e^{-2\mu_j} \sum_{m=0}^{r-1} &\left(\frac{\mu_j^{2m}}{(2m)!} + \frac{\mu_j^{2m+1}}{(2m+1)!} \right) \\ &\times \left(\frac{\mu_j^{2r-2m-2}}{(2r-2m-2)!} + \frac{\mu_j^{2r-2m-1}}{(2r-2m-1)!} \right). \end{aligned}$$

As a result, the expected proportion of deficient predictions, $p_{j,d}$, is of the order $p_{j,d} = \mathcal{O}\left(e^{-2\mu_j}\right)$. As deficient predictions may result in relatively large offsets, the proportion $p_{j,d}$ must not dominate the proportion of large offsets p_j carrying information about singularities. From (33), we have that $p_j = \mathcal{O}\left(\log(n_j)/n_j\right)$, so we impose that $n_j p_{j,d} = o(\log(n_j))$, which implies that $n_j e^{-2\mu_j} = o(\log(n_j))$, from which the heuristic bandwidth (40) follows.

Because of the logarithmic factor in (40), the bandwidth is a little bit larger than the mean distance between the covariate values. Since the bandwidth operates as the scale in the multiscale decomposition, this means that the scale used in the decomposition is a bit larger than the average scale of the data. This is necessary to deal with the irregularly spaced nature of the data. For equidistant data, the logarithmic factor can be replaced by a constant.

The random function generator in Section 5.1 takes $h_0 = 4$, which is larger than the optimal value in a multiscale local polynomial analysis. The larger bandwidth for generating smooth data is necessary because the model itself does not account for any correlation between the detail coefficients in $d_{j,k}$. The correlation plays no role in the analysis, in a similar way that Besov spaces do not take correlations between wavelet coefficients into account. For the generation of realistic functions, adjacent coefficients should be correlated. Alternatively, the scale at which they are generated, can be taken to be larger than the scale of the analysis.

6 A real data example from the Sloan Digital Sky Survey

Since 2000, the Sloan Digital Sky Survey (SDSS) has been observing photometric data and spectra from astronomical objects, typically galaxies, in the context of a redshift survey and using an optical telescope at the Apache Point Observatory in New Mexico, United States. The project has found nearly 50 million galaxies so far. The observed spectra range from near-infrared to ultraviolet frequencies, more precisely covering wavelengths from 380 to 920 nm (i.e.,from 3800 to 9200 Ångström). As the data are noisy, its analysis may include denoising as a preprocessing step. Data are available through `www.sdss.org`, more specifically from `http://skyserver.sdss.org/dr12/en/tools/chart/navi.aspx`.

In the equatorial coordinate system, the astronomical object whose spectrum is studied in Figure 5 can be found at coordinates $(132.86, 11.619)$ (the first coordinate being the right ascension, the second being the declination). The object at this location is a starburst galaxy.

The aim is to identify the typical emission and absorption lines in the observed spectrum. These spectral lines have well known eigen wavelengths λ_e at emission. Comparing the theoretical, emitted wavelength with the observed wavelengths λ_o, allows us to estimate the redshift z defined by $z = (\lambda_o - \lambda_e)/\lambda_e$. As the object under consideration has been found to have $z = 0.1667298$, we can use this value to locate the expected emission and absorption lines on the observed spectrum. We check whether these features are better preserved by a denoising based on multiscale local polynomial decomposition than with the "best-fit" preprocessing step proposed in numerous publications on the SDSS project. The best-fit procedure includes a uniscale cubic spline smoothing, see for instance, Anderson, e.a. [2012, 2014], Percival, e.a. [2010].

Figure 5 depicts the data together with the expected locations of the emission and absorption lines.

The samples size here is $n = 3858$. The multiscale local polynomial decomposition used in this analysis has two resolution levels only, using the heuristic bandwidths, a cosine kernel function, and two vanishing moments, i.e., local linear smoothing. As the data appear to be heteroscedastic, the variance $\sigma^2_{j,k}$ of the detail coefficient $d_{j,k}$ at scale j, location k is estimated locally using a median filter on the absolute values of the coefficients. The coefficients are then thresholded using a scale and location dependent universal threshold $\lambda_{j,k} = \sqrt{2\log(n_j)}\widehat{\sigma}_{j,k}$. Reconstruction from the denoising routine can be found in Figure 6, for comparison with the uniscale "best-fit" method in 7.

From the comparison, we see that the multiscale local polynomial reconstruction is smoother. As smoothness is balanced with goodness of fit, this can be a matter of tuning the parameters, including the choice of the threshold values and the number of scales involved in the processing step. Nevertheless, we also see that the multiscale polynomial reconstruction seems to better capture the local mean value, especially for wavelengths around 870 nm and 910 nm. At the same time, but not entirely visible from these zoomed figures, the small and large peaks are

better reconstructed by the multiscale local polynomial approach. The latter observation is confirmed by plotting the residuals for each of the two procedure, giving more outliers for the "best-fit" approach.

7 Conclusions

7.1 Summary of the algorithm

Given the signal-plus-noise model in (1), the forward multiscale local polynomial transform proceeds as follows

- Assign the observations to the finest scaling coefficients, i.e., for $J = \lceil \log(n) \rceil$, define $\boldsymbol{s}_J = \boldsymbol{Y}$ and $n_J = n$, where n is the sample size, i.e., the length of vector $\boldsymbol{Y}$. This step also defines the finest covariate grid as $\boldsymbol{x}_J = \boldsymbol{x}$.

- Fix the coarsest resolution leven $L < J$. Typically L is a few levels below J. Depending on the application, the coarsest level can be optimised in a heuristic or an adaptive way. The choice of the coarsest resolution level may also be postponed till after the calculations in each iteration step below, thereby turning the for loop into a while loop.

- **For** decreasing value of $j = J-1, J-2, \ldots, L$, indicating the resolution level, **do**

  - **Subsample**, i.e., choose $n_j < n_{j+1}$ and fix the subsampling matrix $\widetilde{\mathbf{J}}_j$ as in Definition 1. Set $\boldsymbol{x}_j = \widetilde{\mathbf{J}}_j \boldsymbol{x}_{j+1}$. In this paper, we adopt dyadic subsampling, i.e., $n_j = \lceil n_{j+1}/2 \rceil$.
  - **Design** the **local polynomial prediction** $\mathbf{P}_j$. Choose the **degree of the polynomial**, $\widetilde{p} - 1$ to be one or higher, so that the function $f(x) = x$ is reconstructed exactly. This avoids the unequispaced covariate locations to be reflected in the reconstruction of processed coefficients.
    Also fix the **bandwidth** at level j, h_j. At this point, take the heuristic value of (40) and Section 5.3.
  - **Design** the **prefilter** $\widetilde{\mathbf{F}}_j$ for given $\mathbf{P}_j$. In particular, make sure that the prefilter preserves at least all polynomials of degree $\widetilde{p} - 1$. This paper concentrates on orthogonal variance reducing prefilters for use in statistical applications, using the iterative procedure in (25). Other prefilters have been proposed in [Jansen, 2013].
  - **Apply** the **prefilter**, as in (2).
  - **Apply** the **local polynomial prediction**, as in (3). At this moment, the bandwidth can be finetuned in a data-adaptive way, by optimisation of the likelihood as in (36).

- Next to the coarse scale coefficients $\boldsymbol{s}_L$ and the details at all intermediate levels $\boldsymbol{d}_j$, $j = L, \ldots, J_1$, the forward transform may deliver also the values of the bandwidths used at each scale.

The implementation of this algorithm, as well as all others used in this paper can be found in the latest version of `ThreshLab`, a Matlab®software package available for download from
`http://homepages.ulb.ac.be/~majansen/software/threshlab.html`.

The forward and inverse Multiscale Local Polynomial Transforms are carried out by the routines `FMLPT1D.m` and `IMLPT1D.m`. The prefilter is implemented in
`prefilterorthnotsparseSTCO.m` and dependent routines.

7.2 Discussion

The multiscale local polynomial transform is an alternative for the wavelet transform. It combines the benefits of smoothness offered by a uniscale local polynomial smoothing with those of sparsity in a multiscale decomposition. It has several specific advantages, especially for the analysis of nonequispaced data. First, its design uses the same operation in the forward and inverse transforms, making it easier and more intuitive than the filter banks in a fast wavelet transform. Second, the design can be based on smoothing, rather than on interpolation. Interpolation may induce fluctuations and therefore it may lead to unpleasant numerical effects. Third, the bandwidth operates as an explicit scale in the multiscale decomposition, thereby allowing the user to adapt the choice of the successive scales in the transform to the application at hand. Fourth, the transformation operates directly on the observations. There is no need for some preprocessing step to avoid the "wavelet crime" or to map the nonequidistant data onto an equidistant grid. Fifth, the transformation may include an additional prefilter step for fine-tuning and variance reduction. As this step has no repercussions on the central smoothing step, both steps can be designed separately.

This paper has investigated the optimal choice of the bandwidth. It has come to the conclusion that the optimal bandwidth is typically a logarithmic factor larger than the average scale of the data, to account for the intermittent space between the covariate values.

The extension of the proposed transform towards multivariate data and to the design of application specific data decompositions are themes of ongoing research.

References

L. Anderson, e.a. The clustering of galaxies in the sdss-iii baryon oscillation spectroscopic survey: Baryon acoustic oscillations in the data release 9 spectroscopic

galaxy sample. *Monthly Notices of the Royal Astronomical Society*, 427(4): 3435–3467, 2012.

L. Anderson, e.a. The clustering of galaxies in the sdss-iii baryon oscillation spectroscopic survey: Baryon acoustic oscillations in the data release 10 and 11 galaxy samples. *Monthly Notices of the Royal Astronomical Society*, 441(1): 24–62, 2014.

P. J. Burt and E. H. Adelson. Laplacian pyramid as a compact image code. *IEEE Trans. Commun.*, 31(4):532–540, 1983.

I. Daubechies. *Ten Lectures on Wavelets*. CBMS-NSF Regional Conf. Series in Appl. Math., Vol. 61. SIAM, Philadelphia, PA, 1992.

M. N. Do and M. Vetterli. Framing pyramids. *IEEE Transactions on Signal Processing*, 51(9):2329–2342, 2003.

J. Fan and I. Gijbels. *Local Polynomial Modelling and its Applications*. Chapman and Hall, London, 1996.

M. Jansen. Multiscale local polynomial smoothing in a lifted pyramid for non-equispaced data. *IEEE Transactions on Signal Processing*, 61(3):545–555, 2013.

M. Jansen. Multiscale local polynomial models for estimation and testing. In M. Akritas, S. N. Lahiri, and D. Politis, editors, *Topics in NonParametric Statistics*, volume 74 of *Springer Proceedings in Mathematics & Statistics*, chapter 14, pages 155–166. Springer, 2014. Proceedings of the First Conference of the International Society for Nonparametric Statistics.

W. J. Percival, e.a. Baryon acoustic oscillations in the sloan digital sky survey data release 7 galaxy sample. *Monthly Notices of the Royal Astronomical Society*, 401(4):2148–2168, 2010.

G. Strang and T. Nguyen. *Wavelets and Filter Banks*. Wellesley-Cambridge Press, Box 812060, Wellesley MA 02181, fax 617-253-4358, 1996.

W. Sweldens. The lifting scheme: A custom-design construction of biorthogonal wavelets. *Appl. Comp. Harmon. Anal.*, 3(2):186–200, 1996.

W. Sweldens. The lifting scheme: a construction of second generation wavelets. *SIAM J. Math. Anal.*, 29(2):511–546, 1998.

P. Vieu. Nonparametric regression: Optimal local bandwidth choice. *Journal of the Royal Statistical Society, Series B*, 53(2):453–464, 1991.

A Construction of prefilters using elementary orthogonal matrices

This sections develops more details on the construction of prefilters using the form in (25). We assume that appropriate choices of γ_j and $\widetilde{\mathbf{J}}_j$ allow us to define a real matrix $\widetilde{\mathbf{S}}_j$ in (24). In particular, as in Lemma 3, the subsampling operation $\widetilde{\mathbf{J}}_j$ leaves out not only the odd indices from $\boldsymbol{s}_{j+1}$, but also $\widetilde{p} - (2n_j - n_{j+1})$ even indices near the boundary. Near the boundary, the filtering is replaced by simple subsampling, meaning for instance that $s_{j,0} =_{j+1,0}$, thus allowing us to focus on the interior. Obviously, alternative solutions may filter near the boundary, at the price of slightly less variance reduction in the interior.

The initial step of the iteration creates the matrix $\widetilde{\mathbf{U}}_j^{[0]} = \widetilde{\mathbf{S}}_j\widetilde{\mathbf{Q}}_j^{[0]} - \widetilde{\mathbf{J}}_j\widetilde{\mathbf{V}}_j^T$. As the primary goal of this step is to map the $n_j \times n_j$ matrix $\widetilde{\mathbf{S}}_j$ onto the $n_j \times (n_{j+1} - \widetilde{p})$ matrix $\widetilde{\mathbf{S}}_j^{[0]} = \widetilde{\mathbf{S}}_j\widetilde{\mathbf{Q}}_j^{[0]}$, the matrix $\widetilde{\mathbf{Q}}_j^{[0]}$ is chosen to be a possibly row-permuted submatrix of the $(n_{j+1} - \widetilde{p}) \times (n_{j+1} - \widetilde{p})$ identity matrix. The resulting matrix $\widetilde{\mathbf{S}}_j^{[0]}$ contains the columns of $\widetilde{\mathbf{S}}_j$ completed with zero columns. Since $\|\mathbf{A} - \mathbf{B}\|_F^2 = \|\mathbf{A}\|_F^2 + \|\mathbf{B}\|_F^2 - 2\mathrm{Tr}(A^T B)$, the norm of $\widetilde{U}_j^{[0]}$ can be minimised by looking at the maximum values in each row of $\widetilde{\mathbf{S}}_j^T \left(\widetilde{\mathbf{J}}_j\widetilde{\mathbf{V}}_j^T\right)$.

The next matrices $\widetilde{\mathbf{Q}}_j^{[i]}$ consist of products of elementary Givens rotations or reflections. More precisely,

$$\widetilde{\mathbf{Q}}_j^{[i]} = \widetilde{\mathbf{Q}}_j^{[i,1]}\widetilde{\mathbf{Q}}_j^{[i,2]}\widetilde{\mathbf{Q}}_j^{[i,3]}\ldots,$$

where the elements of $\widetilde{\mathbf{Q}}_j^{[i,\ell]}$ coincides with those of $\mathbf{I}_j$, except in the 2×2 submatrix at rows (r_1, r_2) and columns (r_1, r_2), where the couple (r_1, r_2) depends on ℓ. This submatrix is equal to

$$\begin{bmatrix} b_\ell & 1 - b_\ell \\ 1 - b_\ell & b_\ell \end{bmatrix} \begin{bmatrix} \cos(\alpha_\ell) & \sin(\alpha_\ell) \\ -\sin(\alpha_\ell) & \cos(\alpha_\ell) \end{bmatrix},$$

where the binary b_ℓ and the real $\alpha_\ell \in [0, 2\pi]$ are chosen to minimize the Frobenius norm of the outcome $\widetilde{\mathbf{U}}_j^{[i,\ell]} = \widetilde{\mathbf{S}}_j^{[i,\ell-1]}\widetilde{\mathbf{Q}}_j^{[i,\ell]} - \widetilde{\mathbf{J}}_j\widetilde{\mathbf{V}}_j^T$. Since right multiplication with $\widetilde{\mathbf{Q}}_j^{[i,\ell]}$ affects rows r_1 and r_2 only, the values of b_ℓ and α_ℓ are easy to find. A slight modification consists in weighting the elements of the matrix in the calculation of the Frobenius norm. More precisely, by putting zero weights on the elements near the diagonal of $\widetilde{\mathbf{U}}_j^{[i,\ell]}$, this matrix and also $\widetilde{\mathbf{F}}_j^{[i,\ell]}$ are pushed towards diagonal dominance.

B Proof of Lemma 2

First, the polynomial reproducing condition (17) represents $n_j\widetilde{p}$ linear equations, independently from the number of zeros elements in $\widetilde{\mathbf{F}}_j$. Second, the diagonal

of (16) adds n_j non-linear non-homogeneous equations. The homogeneous equations corresponding to the off-diagonals express orthogonality between rows of $\widetilde{\mathbf{F}}_j$. These equations are inactive if the nonzeros of the rows under consideration have no overlap. Ignoring boundary effects, this amounts to $\lceil (r-l)/2 \rceil - 1$ active equations on each row of (16). Third, from the proof of Lemma 1, it turns out that (16) being the optimal variance reduction implies that all columns of $\widetilde{\mathbf{F}}_j$ add up to the same value, more precisely $\widetilde{\mathbf{F}}_j^T \mathbf{1}_j = \mathbf{1}_{j+1}(n_j/n_{j+1})$. These n_{j+1} equations for column sums are linearly independent from the row sum equations in (17), except for the last one. In total, we have at least $\mathcal{O}(n_j\widetilde{p}+n_j+n_j(\lceil (r-l)/2 \rceil -1)+n_{j+1})$ conditions, which is at least $\widetilde{p}+2+\lceil (r-l)/2 \rceil$ at each row. Therefore, the number of free parameters at each row, $r-l$, must satisfy $(r-l) \geq \widetilde{p}+2+\lceil (r-l)/2 \rceil$, which leads to the stated result. □

In the case of equidistant covariate values, the prefilter $\widetilde{\mathbf{F}}_j$ has only $2\widetilde{p}$ nonzeros on each row. This can be understood from the fact that for equidistant covariates, $\widetilde{\mathbf{F}}_j = \widetilde{\mathbf{J}}_j \overline{\mathbf{F}}_j$, where is $\overline{\mathbf{F}}_j$ a Toeplitz matrix. The proof of Lemma 1 reveals that in the general case $n_{j+1}-1$ linear equations follow from minimising the output variance, namely that all column sums $\widetilde{\mathbf{F}}_j^T \mathbf{1}_j$ have the same value. For $\widetilde{\mathbf{F}}_j = \widetilde{\mathbf{J}}_j \overline{\mathbf{F}}_j$, with $\overline{\mathbf{F}}_j$ Toeplitz, all even column sums are automatically equal to each other, and the same holds for the odd column sums, thereby reducing $n_{j+1}-1$ linear equations to a single equation. This subtracts the value two from the number of nonzeros on each row. But because of that, there is less overlap between rows, giving us a bonus reduction of two elements.

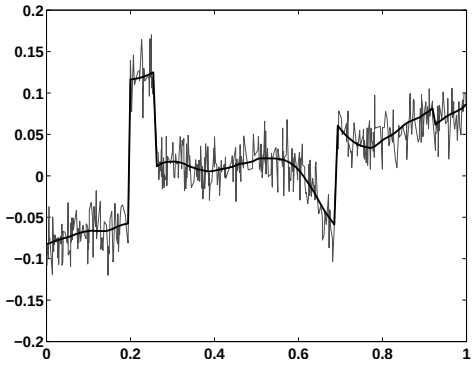

Figure 2: One of the test functions $f(x)$ in Figure 1, along with additive normal errors.

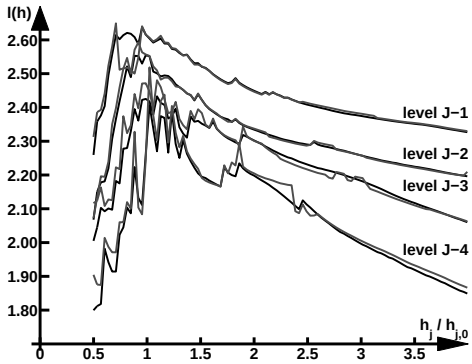

Figure 3: Black lines: likelihood curves (36) for one of the test functions in Figure 1, grey lines: easy-to-evaluate approximations (39) at four resolution levels as a function of the bandwidth, normalised with the heuristic values given by (40). Level $J-1$ is the finest, level $J-4$ is the coarsest.

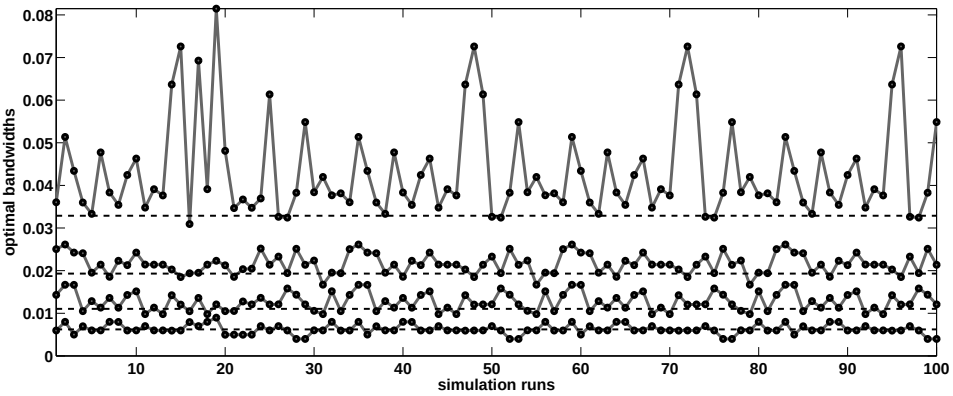

Figure 4: Solid line charts: maximum likelihood bandwidths at 4 scales for 100 replicates of the experiment in Figure 3. Dashed lines: heuristic values of (40).

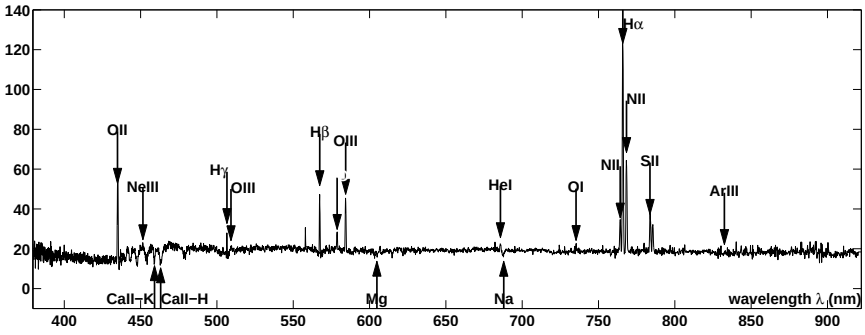

Figure 5: Spectrum for astronomical object (starburst galaxy) at equatorial coordinates $(132.86, 11.619)$. Raw data, with indicators to a selection of emission lines (downwards arrows) and absorption lines (upwards arrows). The spectral lines are computed from the emitted values and the redshift.

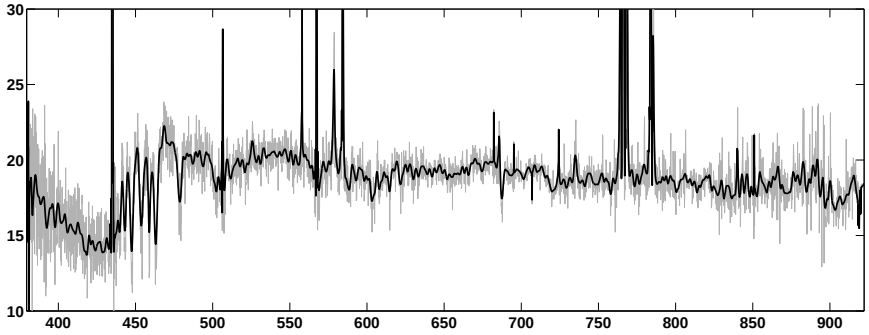

Figure 6: Multiscale local polynomial based estimated spectrum from Figure 5, together with the raw data in background grey. The vertical scale has been zoomed in compared to Figure 5.

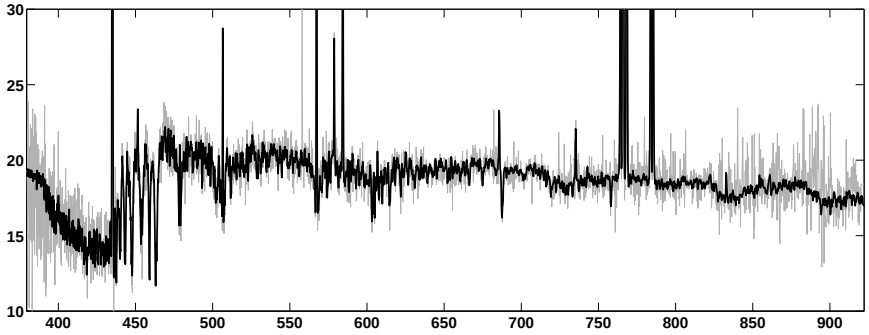

Figure 7: "Best-fit" Anderson, e.a. [2012, 2014], Percival, e.a. [2010] estimation of the spectrum from Figure 5, based on, among others, a uniscale cubic spline smoothing.

C Proof of Lemma 3

In this proof, $\widetilde{\mathbf{J}}_j$ denotes a partitioning operation, which can be more general than the even-odd subsampling.

Let $\widetilde{\mathbf{X}}_{j+1}^{(2)}$ be the orthonormalised columns of $\mathbf{X}_{j+1}^{(2)}$, i.e., $(\widetilde{\mathbf{X}}_{j+1}^{(2)})_{i,1} = n_{j+1}^{-1/2}$ and $(\widetilde{\mathbf{X}}_{j+1}^{(2)})_{i,2} = n_{j+1}^{-1/2}(x_{j+1,i} - \overline{\boldsymbol{x}}_{j+1})/(\overline{\boldsymbol{x}^2_{j+1}} - \overline{\boldsymbol{x}}^2_{j+1})^{1/2}$. Then for any vector $\boldsymbol{c}$ of length n_j, we have

$$\boldsymbol{c}^T\widetilde{\mathbf{J}}_j(\widetilde{\mathbf{V}}_j^T\widetilde{\mathbf{V}}_j - (1-\gamma_j)\mathbf{I}_{j+1})\widetilde{\mathbf{J}}_j^T\boldsymbol{c} = \boldsymbol{c}^T\widetilde{\mathbf{J}}_j\widetilde{\mathbf{V}}_j^T\widetilde{\mathbf{V}}_j\widetilde{\mathbf{J}}_j^T\boldsymbol{c} - (1-\gamma_j)\boldsymbol{c}^T\widetilde{\mathbf{J}}_j\widetilde{\mathbf{J}}_j^T\boldsymbol{c} = \|\widetilde{\mathbf{V}}_j\widetilde{\mathbf{J}}_j^T\boldsymbol{c}\|_2^2 - (1-\gamma_j)\|\widetilde{\mathbf{J}}_j^T\boldsymbol{c}\|_2^2.$$

As $[\widetilde{\mathbf{X}}_{j+1}^{(2)}\widetilde{\mathbf{V}}_j^T]$ constitutes an orthogonal matrix, we have $\|\widetilde{\mathbf{X}}_{j+1}^{(2)T}\widetilde{\mathbf{J}}_j^T\boldsymbol{c}\|_2^2 + \|\widetilde{\mathbf{V}}_j\widetilde{\mathbf{J}}_j^T\boldsymbol{c}\|_2^2 = \|\widetilde{\mathbf{J}}_j^T\boldsymbol{c}\|_2^2 = \|\boldsymbol{c}\|_2^2$. From this it follows that

$$\boldsymbol{c}^T\widetilde{\mathbf{J}}_j(\widetilde{\mathbf{V}}_j^T\widetilde{\mathbf{V}}_j - (1-\gamma_j)\mathbf{I}_{j+1})\widetilde{\mathbf{J}}_j^T\boldsymbol{c} \geq 0 \Leftrightarrow \|\widetilde{\mathbf{X}}_{j+1}^{(2)T}\widetilde{\mathbf{J}}_j^T\boldsymbol{c}\|_2^2 \leq \gamma_j\|\boldsymbol{c}\|_2^2.$$

This holds for any vector $\boldsymbol{c}$ if $\|\widetilde{\mathbf{X}}_{j+1}^{(2)T}\widetilde{\mathbf{J}}_j^T\|_2^2 = \|\widetilde{\mathbf{J}}_j\widetilde{\mathbf{X}}_{j+1}^{(2)}\|_2^2 \leq \gamma_j$. By writing $\rho(\mathbf{A})$ for the spectral radius of matrix $\mathbf{A}$, the squared matrix norm is

$$\|\widetilde{\mathbf{J}}_j\widetilde{\mathbf{X}}_{j+1}\|_2^2 = \rho(\widetilde{\mathbf{X}}_{j+1}^{(2)T}\widetilde{\mathbf{J}}_j^T\widetilde{\mathbf{J}}_j\widetilde{\mathbf{X}}_{j+1}) = \frac{n_j}{n_{j+1}}\rho\left(\begin{bmatrix} 1 & \xi_j \\ \xi_j & 1-\zeta_j \end{bmatrix}\right),$$

with ξ_j and ζ_j as defined in (26) and (27). The squared matrix norm is equal to the minimum value of γ_j proposed in (28). □

D Kullback-Leibler divergence of the approximative mixture model in Section 4.3

The error term in (37) consists of two terms $R(p_j, a_j, \tau_j, \sigma_{j,k}) = R_0(p_j, a_j, \tau_j, \sigma_{j,k}, \lambda_j) + R_1(p_j, a_j, \tau_j, \sigma_{j,k}, \lambda_j)$. These terms are defined as

$$\begin{aligned} R_0(p_j, a_j, \tau_j, \sigma_{j,k}, \lambda_j) &= E_{\widetilde{D}_{j,k}|\boldsymbol{X}}\left[\log\left(\frac{f_{\widetilde{D}_{j,k}|\boldsymbol{X}}(\widetilde{D}_{j,k}|\boldsymbol{X}))}{(1-\widetilde{p}_{\lambda;j,k})\widetilde{f}_0(\widetilde{D}_{j,k};\epsilon_{j,k},\lambda_j)}\right) I(\widetilde{M}_{j,k}=0|\boldsymbol{X})\right], \\ R_1(p_j, a_j, \tau_j, \sigma_{j,k}, \lambda_j) &= E_{\widetilde{D}_{j,k}|\boldsymbol{X}}\left[\log\left(\frac{f_{\widetilde{D}_{j,k}|\boldsymbol{X}}(\widetilde{D}_{j,k}|\boldsymbol{X}))}{\widetilde{p}_{\lambda;j,k}\widetilde{f}_1(\widetilde{D}_{j,k};a_j,\lambda_j\epsilon_{j,k})}\right) I(\widetilde{M}_{j,k}=1|\boldsymbol{X})\right]. \end{aligned}$$

Since

$$\log\left(\frac{f_{\widetilde{D}_{j,k}|\boldsymbol{X}}(u)}{(1-\widetilde{p}_{\lambda;j,k})\widetilde{f}_0(u)}\right) = \log(2\Phi(\lambda_j)-1) - \log(1-\widetilde{p}_{\lambda;j,k}) + \log\left[1 + p_j\left(\frac{g(u;a_j,\sigma_{j,k})}{\phi(u/\epsilon_{j,k})/\epsilon_{j,k}} - 1\right)\right],$$

we can rewrite

$$R_0(p_j, a_j, \tau_j, \sigma_{j,k}, \lambda_j) = (1-\widetilde{p}_{\lambda;j,k})\left[\log(2\Phi(\lambda_j)-1) - \log(1-\widetilde{p}_{\lambda;j,k})\right] + R_{00}(p_j; a_j, \tau_j, \sigma_{j,k}, \lambda_j),$$

where the following definition of R_{00} uses the short notation $\epsilon = \sqrt{\sigma^2 + \tau^2}$,

$$\begin{aligned} R_{00}(p; a, \tau, \sigma, \lambda) &= E_{\widetilde{D}}\left(\log\left[1 + p\left(\tfrac{g(\widetilde{D};a,\sigma)}{\phi_\epsilon(\widetilde{D})} - 1\right)\right]\middle|\widetilde{M} = 0\right) P(\widetilde{M} = 0) \\ &= E_{\widetilde{D}}\left(\log\left[1 + p\left(\tfrac{g(\widetilde{D};a,\sigma)}{\phi_\epsilon(\widetilde{D})} - 1\right)\right]\middle|\widetilde{M} = 0, M = 0\right) P(M = 0|\widetilde{M} = 0)P(\widetilde{M} = 0) \\ &+ E_{\widetilde{D}}\left(\log\left[1 + p\left(\tfrac{g(\widetilde{D};a,\sigma)}{\phi_\epsilon(\widetilde{D})} - 1\right)\right]\middle|\widetilde{M} = 0, M = 1\right) P(M = 1|\widetilde{M} = 0)P(\widetilde{M} = 0 \end{aligned}$$

In this expression, we can bound the second factor of the second term by

$$P(M = 1|\widetilde{M} = 0) = \frac{pP(|\widetilde{D}| \leq \lambda\epsilon|M = 1)}{pP(|\widetilde{D}| \leq \lambda\epsilon|M = 1) + (1 - p)P(|\widetilde{D}| \leq \lambda\epsilon|M = 0)} \leq p,$$

provided that $P(|\widetilde{D}| \leq d|M = 1) \leq P(|\widetilde{D}| \leq d|M = 0)$ for any positive d, which is true for $a\epsilon$ sufficiently close to zero. In the first factor of the second term, we use that $g(u; a, \sigma)/\phi_\epsilon(u)$ is symmetric on $[-\lambda\epsilon, \lambda\epsilon]$ and non-decreasing on $[0, \lambda\epsilon]$,

$$E_{\widetilde{D}}\left(\log\left[1 + p\left(\frac{g(\widetilde{D}; a, \sigma)}{\phi_\epsilon(\widetilde{D})} - 1\right)\right]\middle|\widetilde{M} = 0, M = 1\right) \leq \log\left[1 + p\left(\frac{g(\lambda\epsilon; a, \sigma)}{\phi_\epsilon(\lambda\epsilon)} - 1\right)\right].$$

This factor is bounded below by $-p/(1-p)$ and above by $\mathcal{O}(\lambda^2)$, while the values of a, σ, and ϵ have no impact on these bounds.

In the first term we have, for p small,

$$E_{\widetilde{D}}\left(\log\left[1 + p\left(\frac{g(\widetilde{D}; a, \sigma)}{\phi_\epsilon(\widetilde{D})} - 1\right)\right]\middle|\widetilde{M} = 0, M = 0\right) = pE_{\widetilde{D}}\left(\frac{g(\widetilde{D}; a, \sigma)}{\phi_\epsilon(\widetilde{D})} - 1\middle|\widetilde{M} = 0, M = 0\right) + o(p).$$

This is further rewritten and bounded as

$$E_{\widetilde{D}}\left(\frac{g(\widetilde{D}; a, \sigma)}{\phi_\epsilon(\widetilde{D})} - 1\middle|\widetilde{M} = 0, M = 0\right) = \int_{-\lambda\epsilon}^{\lambda\epsilon} [g(u; a, \sigma) - \phi_\epsilon(d)]du \in [-1, 1].$$

All together, we have, for $\lambda \to \infty$ and $p \to 0$,

$$|R_{00}(p; a, \tau, \sigma, \lambda)| = \mathcal{O}\left(p \cdot 1 \cdot 1 + \lambda^2 \cdot p \cdot 1\right) = \mathcal{O}\left(p\lambda^2\right).$$

For $R_1(p_j, a_j, \tau_j, \sigma_{j,k}, \lambda_j)$, we use that $\widetilde{p}_\lambda = P(\widetilde{M} = 1|\boldsymbol{X}) = p\,2\,[1 - G(\lambda\epsilon; a, \sigma)] + (1 - p)\,2\,[1 - \Phi(\lambda)]$. For $\lambda\epsilon \to \infty$ and for $a \to 0$, so that $a\lambda\epsilon \to 0$, it can be verified that $G(\lambda\epsilon; a, \sigma) \to 0$, and thus $\widetilde{p}_\lambda/p \to 1$. The condition that $a\lambda\epsilon \to 0$ means that $\lambda \to \infty$, but not too fast, as otherwise, it would take away too many large coefficients. Furthermore, as

$$R_1(p, a, \tau, \sigma, \lambda) = E\left[\log\left(\frac{pg(D; a, \sigma) + (1 - p)\phi(D/\epsilon)/\epsilon}{\widetilde{p}_\lambda \widetilde{f}_1(D; a, \lambda\epsilon)}\right)\middle||D| > \lambda\epsilon\right] P\left(|D| > \lambda\epsilon\right),$$

this term is bounded from below and from above by

$$\begin{aligned}\widetilde{p}_\lambda \cdot \log\left(\frac{p}{\widetilde{p}_\lambda}\inf_{|d|>\lambda\epsilon} r_1(d) + \frac{(1-p)}{\widetilde{p}_\lambda}\inf_{|d|>\lambda\epsilon} r_0(d)\right) &\leq R_1(p,a,\tau,\sigma,\lambda) \\ &\leq \widetilde{p}_\lambda \cdot \log\left(\frac{p}{\widetilde{p}_\lambda}\sup_{|d|>\lambda\epsilon} r_1(d) + \frac{(1-p)}{\widetilde{p}_\lambda}\sup_{|d|>\lambda\epsilon} r_0(d)\right),\end{aligned}$$

where $r_1(d) = g(d;a,\sigma)/\widetilde{f}_1(d;a,\lambda\epsilon)$ and $r_0(d) = \phi(d/\epsilon)/\epsilon\widetilde{f}_1(d;a,\lambda\epsilon)$. The function $r_1(d)$ reaches a global minimum at $d=0$, from where it increases monotonically from $r_1(0) = 2[1-\Phi(a\sigma)]\exp(\sigma^2a^2/2)$ towards $r_1(\pm\infty) = \exp(\sigma^2a^2/2)$. On $]-\infty,-\lambda\epsilon]\cup[\lambda\epsilon,\infty[$, and for $\lambda > a\epsilon$, the function $r_0(d) = \phi(d/\epsilon)/\epsilon\widetilde{f}_1(d;a,\lambda\epsilon)$ decreases monotonically and rapidly from $r_0(\lambda\epsilon) = \exp(-\lambda^2/2)/\sqrt{\pi/2}a\epsilon$ to 0. The bounds for $R_1(p,a,\tau,\sigma,\lambda)$ are then

$$\begin{aligned}\widetilde{p}_\lambda \cdot \log\left(\frac{p}{\widetilde{p}_\lambda}2[1-\Phi(a\sigma)]\exp(\sigma^2a^2/2)\right) &\leq R_1(p,a,\tau,\sigma,\lambda) \\ &\leq \widetilde{p}_\lambda \cdot \log\left(\frac{p}{\widetilde{p}_\lambda}\exp(\sigma^2a^2/2) + \frac{(1-p)}{\widetilde{p}_\lambda}\frac{\exp(-\lambda^2/2)}{\sqrt{\pi/2}a\epsilon}\right),\end{aligned}$$

from which we conclude that $R_1(p,a,\tau,\sigma,\lambda) \sim \widetilde{p}_\lambda \sim p$, when $a \to 0$ and $\lambda \to \infty$, so that $\exp(-\lambda^2/)/a \to 0$. This means that λ should grow sufficiently fast in order to prevent false positives from dominating the error term R_1. □