Journal of Mathematics and Statistics

Original Research Paper

Two-Stage Estimation in Inverse Problems using Combined Wavelet Thresholding and Penalized Maximum Likelihood

Robert G Aykroyd and Hassan Aljohani

Department of Statistics, University of Leeds, UK

Article history
Received: 30-06-2017
Revised: 09-08-2017
Accepted: 25-09-2017

Corresponding Author:
Robert G Aykroyd
Department of Statistics,
University of Leeds, UK
Email: r.g.aykroyd@leeds.ac.uk

Abstract: Inverse problems occur in a wide range of practical scientific investigations where the variables of interest are only observed indirectly, such as magnetic and seismic imaging in geophysics, electrical tomography in industrial process monitoring, or PET scanning in medicine. Linear inverse problems can be thought of as highly multivariate regression problems with strong multicollinearity where the aim is to interpret regression parameters-prediction is not of interest. Estimation, to give a fitted model, is known as an inverse problem which can be ill-posed and ill-conditioned, making estimation using least-squares or maximum likelihood unstable or even impossible. Instead, one approach is to introduce additional constraints through a penalty term and a penalized least-squares or penalized maximum likelihood approach taken. The major cause of numerical problems in the estimation is noise in the data and hence using a pre-processing which reduces noise may be helpful. Wavelet thresholding has proven to be highly efficient at separating useful information from noise but there has been very little work considering the use of wavelet methods for inverse problems. Hence it is of great interest to investigate the usefulness of this as an additional step in estimation for inverse problems. In particular a two stage process is proposed combining inversion and wavelet thresholding. The thresholding will be considered as either a pre-inversion or post-inversion filter and the results compared. A simulation investigation is described and reported which compares these two alternative, and also which uses a minimum mean-squared error approach to choose the penalty parameter, in the inversion, and the threshold, in the wavelet thresholding, either sequentially or jointly. The results demonstrate that a combined approach is worthwhile and that for the piecewise constant test function considered, it is better to post-process after the inversion step than it is to use the more intuitive wavelet thresholding pre-processing step for noise reduction before inversion. This new approach hence has the potential to enhance the estimation results in a wide range of applied inverse problems.

Keywords: Inverse Problems, Penalized Likelihood, Wavelet Thresholding

Introduction

Inverse problems are ubiquitous in science and engineering and have received widespread attention from scientists, including in areas such as geophysics, engineering and medicine. Many of these can be classified as function estimation or image processing problems, Aykroyd (2015). In a statistical context key challenges include dealing with the large number of unknown parameters compared to the amount of data and the highly multicollinear nature of the design matrix. In regression, a common approach would be to

perform lasso (Tibshirani, 1996) or ridge regression (Hoerl and Kennard, 1970) to stabilize estimation. These work well in standard model selection type regression problems (Zou and Hastie, 2005)-but the theme of this paper is function estimation, rather than variable selection or prediction, and such shrinkage estimators are not appropriate as they would effectively introduce a bias towards zero. Instead, some form of assumption about the smoothness of the unknown function is more appropriate and hence additional constraints in the form of local differences are widely used.

Inverse problems can be divided into two main types, linear and non-linear inverse problems. The most common being linear problems, the theme of this paper, which can be defined by the following vector-matrix model:

$$y_n = Kf_m + \epsilon_n \tag{1}$$

with data vector $y_{n\times1} = \{y_i: i = 1,...,n\}$, kernel matrix $K_{n\times m} = \{K_{ij}: i = 1,...,n, j = 1,...,m\}$, vector of unknown parameters $f_{m\times1} = \{f_j: j = 1,...,m\}$ and errors $\epsilon_{n\times1} = \{\epsilon_i: i = 1,..., n\}$. Further, the errors are often assumed to be independent and identically distributed normal random variables, that is $\epsilon \sim N_n(0, \sigma^2 I_n)$. Note that the use of notation f_m for the vector of unknowns, rather than the more usual β in regression modelling and K rather than X for what would be called the design matrix, has been chosen to be consistent with the later notation for function estimation.

As illustration and for later use in simulation experiments, consider the Blocks test functions (Donoho and Johnstone, 1994; Nason, 2010a) from the wavethresh package (Nason, 2010b) available in *R* (R Core Team, 2016). The piecewise constant nature of this function makes estimation a very challenging problem especially when tackled as an inverse problem, but it is well motivated by stratigraphy problems in archaeology (Allum *et al*., 1999). Next, consider Gaussian blurring leading to the kernel matrix, K, defined as:

$$K_{ij} = \frac{1}{\sqrt{2\pi\delta^2}}\exp\left(-\frac{\delta_{ij}^2}{2\delta^2}\right),\ i = 1,...,n, j = 1,...,m \tag{2}$$

where, $\delta_{ij} = i\text{-}j$ and δ is a positive parameter which controls the amount of blur.

Figure 1 shows three examples with $n = m = 64$ and $\sigma = 1$ but for a range of values of δ. In each the same red dashed line shows the true, but in practice unknown, function which is to be estimated, then the black solid line shows the blurred result of applying a kernel matrix and finally the points show typical data. In (a) there is no blur and hence the points are scattered equally around the true function. As the blurring increases, the edges of the true step function are rounded, as in (b) and then all detail is completely lost, as in (c). The examples in (b) and (c) correspond to moderate and large blurring of the underlying function and hence moderate and difficult inverse problems - the reciprocal condition numbers are 6×10^{-4} and 4×10^{-20} with values close to 1 indicating a well-conditioned problem (Golub and Van Loan, 1989). Estimation should be easy in (a), accurate and reliable in (b), but might be essentially useless in (c).

The rest of this paper is structured as follows. Section 2 provides key properties of inverse estimation and Section 3 an introduction to wavelet methods. Section 4 describes the proposed two-stages approach with a simulation study to investigate estimation properties in Section 5. The final summary and conclusions are presented in Section 6.

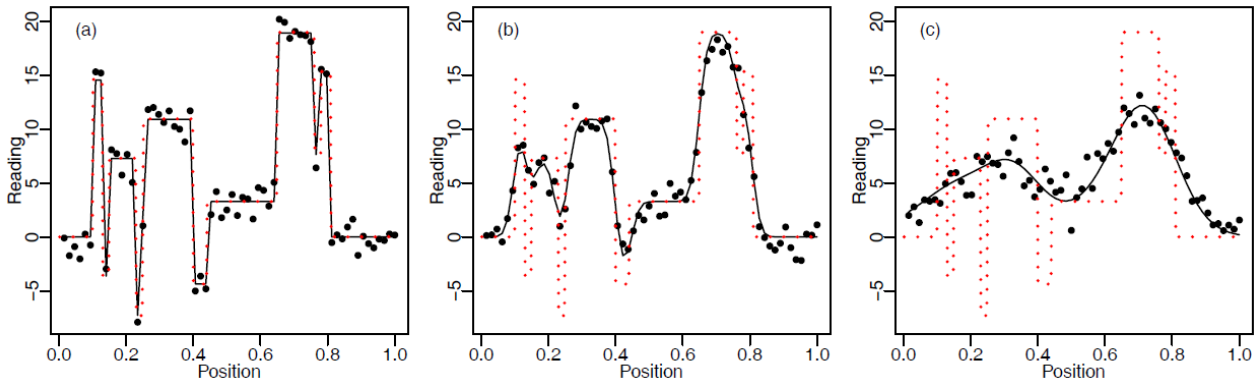


Fig. 1. Typical data (points) derived from the Blocks test function (dashed line) along with blurred test function (solid line) for different levels of blur, (a) no blur ($\delta = 0$), (b) moderate blur ($\delta = 0.02$), (c) large blur ($\delta = 0.08$)

Inverse Estimation using Penalized Likelihood

From the above mathematical statements it is now possible to define the log-likelihood:

$$\ell(f) = -\frac{1}{2\sigma^2}(y - Kf)^T (y - Kf) \tag{3}$$

with the maximum likelihood estimate of *f* given by:

$$\hat{f}_{ML} = \left(K^T K\right)^{-1} K^T y. \tag{4}$$

Note that our aim is not to fit a model to allow the prediction of *y* but to interpret the estimates of *f*. This means that stable estimation of *f* is a requirement of the procedure. In inverse problems, however, estimation of this unknown parameter vector is not straight forward as either: (i) no solution exists, (ii) there are multiple solutions or (iii) the solution does not depend smoothly on the data as small changes in the noise can lead to wildly different estimates - these properties define an inverse problem (Hadamard, 2014). Reinterpreting these conditions into statistical terminology. The first reason is that the number of parameters is larger and sometimes much larger, than the number of observations. The second reason is that even when the number of parameters is fewer than the number of data points there can still be problems due to collinearity, which is the condition where the independent variables are strongly correlated with each other.

Hence, in many inverse problems it is not possible to calculate the inverse, $(K^T K)^{-1}$, as the system has fewer equations than unknowns or is ill-conditioned being nearly multicollinear. To solve this problem, additional constraints are introduced leading to a penalized log-likelihood:

$$\ell_p(f, \kappa) = -\frac{1}{2\sigma^2}(y - Kf)^T (y - Kf) - \kappa R(f),\ \kappa > 0 \tag{5}$$

where, *R*(*f*) is a penalty function with small values of *R*(*f*) indicating preferred choices of *f*. The parameter κ is chosen to balance the relative weight given to the likelihood and penalty terms. Before moving on, it is worth noting that the penalized log-likelihood can be interpreted in a Bayesian setting as log-likelihood plus log-prior, but that approach will not be adopted here.

In many situations the penalty can be written in terms of a matrix, that is *R*(*f*) = *Rf* and in these cases the solution of the penalized likelihood problem produces the estimation equation:

$$\hat{f} = \left(K^T K + \kappa R^T R\right)^{-1} K^T y \tag{6}$$

Common choices of R can be derived based on assumptions about the smoothness of *f*. If it is believed that the function is not different from a constant, then this suggests considering the first derivative of *f* which can be approximated by the first difference and then $R_1(f) \propto \int \left(f'(t)\right)^2 dt \approx \sum \left(\hat{f}_i - \hat{f}_{i+1}\right)^2$. Note that this equals zero if and only if $\hat{f}(t)$ is constant. Then, the corresponding matrix representations, R_1 can be written as:

$$R_1 = \begin{pmatrix} 1 & -1 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 0 & -1 & 1 \end{pmatrix}.$$

This leads to what is called first-order smoothing. The idea can be extended to higher order smoothing, but the first-order generally works very well even when the unknown function is not a constant.

To measure the accuracy of the fitted function, $\hat{f}_{m\times 1} = \left\{\hat{f}_j : j = 1,\dots,m\right\}$, the mean squared-error can be calculated and then the best value for the penalty parameter, κ, found by minimising this mean-squared error, that is:

$$\hat{\kappa} = \arg\min_{\kappa} MSE(\kappa), \text{ where } MSE = \frac{1}{m}\sum_{j=1}^{m}\left(\hat{f}_j - f_j\right)^2. \tag{7}$$

Although, in practice, the true function is unknown it is usual to either have training data or be able to perform realistic simulations. Further, simulation also allows a comparison of different estimation approaches.

To illustrate standard function estimation using penalized likelihood inversion, consider Fig. 2 and 3 which use $\delta = 0.02$ and $\delta = 0.08$ respectively - these are the same cases as shown in Fig. 1 corresponding to moderate and large blurring of the underlying function. When $\delta = 0.02$, Fig. 2, $\hat{\kappa} = 0.011$ and $MSE(\hat{\kappa}) = 6.83$. Although in (a) it is not clear that the estimate is better than the data, noting that the MSE of the data is 9.75 reveals a substantial improvement has been achieved.

In Fig. 3, with $\delta = 0.08$ the situation is a little different. In (b) the location of the minimum is poorly defined - in contrast to the well-defined minimum in Fig. 2b – with all κ values above about 0.01 producing a similar MSE but with $\hat{\kappa} = 0.060$ and $MSE(\hat{\kappa}) = 20.76$ compared to a data MSE of 26.18. The estimate clearly follows the true function slightly better with the peaks and troughs more pronounced. These examples, however, have highlighted the main drawback of estimating piecewise constant functions using smoothing penalties - that is the estimates are smooth and are not piecewise constant.

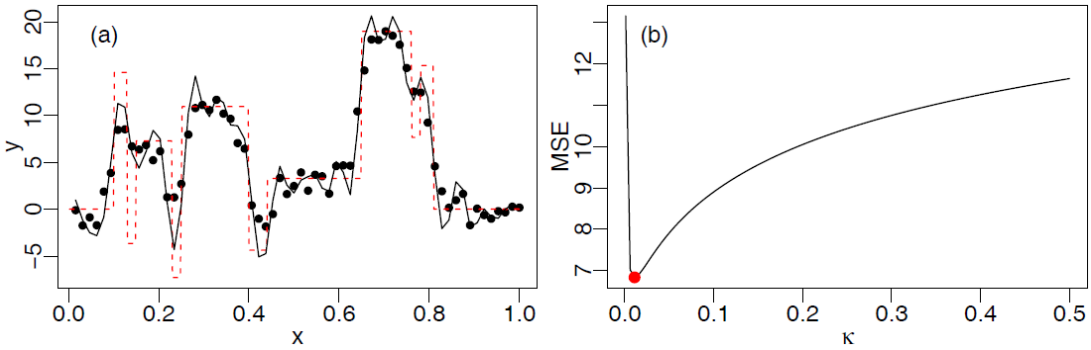


Fig. 2. Inversion, (a) estimate (solid black line) of Blocks test function (red dashed line) from n = 64 data values (points) with δ = 0.02, (b) Mean-squared error (black line) as a function of the penalty parameter κ showing the minimum MSE value (point)

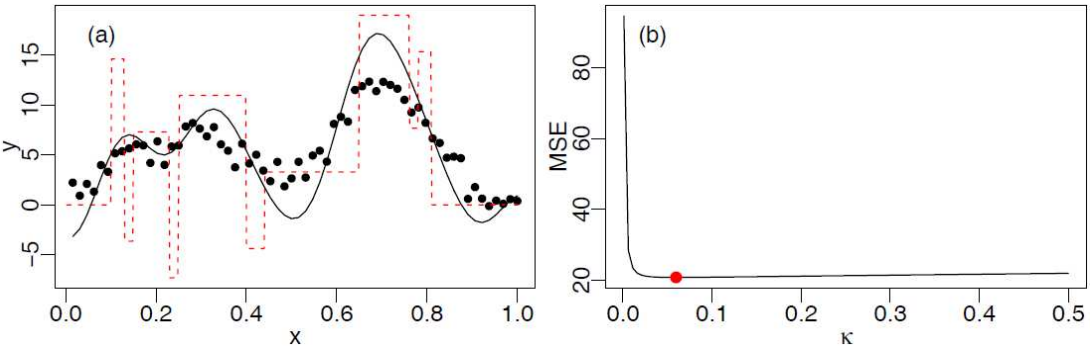


Fig. 3. Inversion, (a) estimate (solid black line) of Blocks test function (red dashed line) from n = 64 data values (points) with δ = 0.08, (b) Mean-squared error (black line) as a function of the penalty parameter κ showing the minimum MSE value (point)

Wavelet Representations and Thresholding Methods

The Discrete Wavelet Transform

Wavelet theory can be applied in many fields and applications (Young, 1993; Vidakovic, 2009) and can be explained in simple terms as describing a signal by a few wavelet coefficients, hence producing a sparse and multi-resolution representation. The most common way in which wavelets are applied is to de-noise signals which can be achieved through thresholding or shrinkage of the wavelet coefficients and then reconstruct of the signal - a straightforward introduction can be found in Vidakovic and Mueller (1994). This has the effect of both reducing the noise contribution and compressing the original data while keeping a good quality of approximation (Raimondo, 2002).

In the standard setting, consider an unknown function f at a set of equally-spaced locations which is corrupted by noise. Consider a set of noisy data $y = (y_1,..., y_n)$ that are observed values recorded at the same locations, then the model is given by:

$$y = f + \epsilon \tag{8}$$

where, ϵ is a vector of random variables such that $\epsilon \sim N_n(0, \sigma^2 I_n)$ and $n = 2^J$, for some index $J \in \mathbb{N}$. Consider the wavelet transform of the unknown function f defined by:

$$d_f = W^T f$$

where, W is an orthonormal matrix containing the wavelet basis. Hence, the unknown function f can be equivalently defined by its discrete wavelet transform $d_f = \{d_{ij}: i = 0,...,2^{j-1}, j = 0,...,J\text{-}1\}$ where $J = \log_2(n)$. The wavelet decomposition of the data y can be written as:

$$d_y = Wy = W(f + \epsilon) = Wf + W\epsilon = d_f + \eta \tag{9}$$

where, d_y and d_f are vectors of the wavelet coefficients of y and f respectively. Thus, the model in (9) can be written equivalently as:

$$d_y = d_f + \eta. \tag{10}$$

The orthogonality of matrix W and normality of the noise vector ϵ implies the noise vector η is also normal with the same structure as ϵ, that is $\eta \sim N_n(0, \sigma^2 I_n)$.

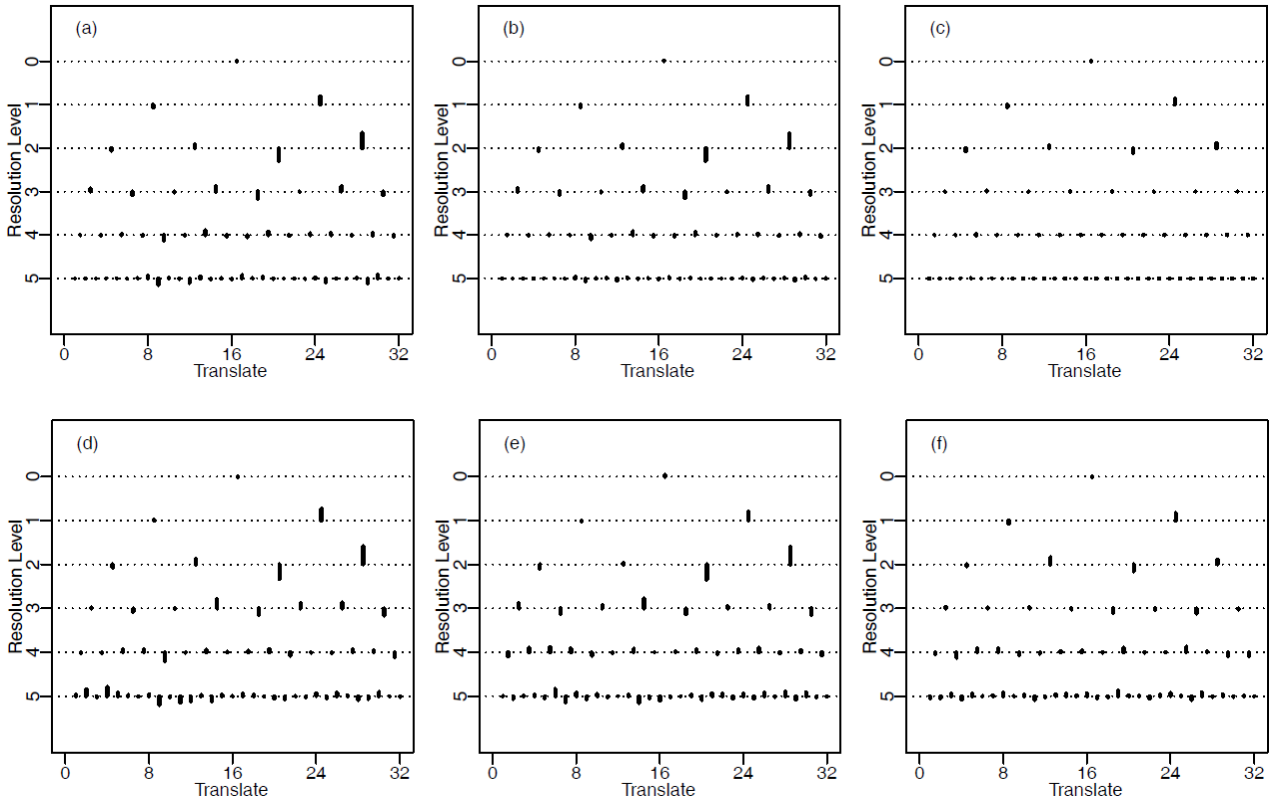


Fig. 4. Wavelet tableaux of Blocks test function, for $\delta = 0, 0.01, 0.1$ (columns) and $\sigma = 0, 5$ (rows)

Figure 4 shows empirical wavelet coefficients for the Blocks test function sampled at m = 64 equally spaced points with $\delta = 0, 0.01, 0.1$ (columns) and $\sigma = 0, 5$ (rows) - all panels have common scale to allow direct comparison. Hence, (a) shows the wavelet coefficients of the true function, then moving along a row shows the effects of increased blur and moving down a row corresponds to increased noise. As the blur increases the non-zero wavelet coefficients become closer to zero, whereas as the level of noise becomes large, the number of non-zero wavelet coefficients in the finer levels increases.

Wavelet Thresholding

Wavelet thresholding is a non-parametric and non-linear technique used in function estimation based on a concept of sparseness. Hence, thresholding of the empirical wavelet coefficients works best in problems where the underlying set of true coefficients is sparse. It is assumed that the majority of the wavelet coefficients are small, which are set to zero and the remaining few are large, which are kept. This is sometimes described as those below a threshold are "removed" while the others are "kept". The aim is that the resulting adjusted wavelet coefficients contain less noise whilst retaining important information. The simplest example is the Hard thresholding rule which is defined as:

$$\hat{d}_f = \begin{cases} 0, & if\ |d| \le \lambda \\ d_y, & if\ |d| > \lambda \end{cases} \tag{11}$$

where, λ is the threshold. The set of wavelet coefficients after thresholding $\hat{d}_f$ are then taken as estimates of the true wavelet coefficients d_f. Then, an estimate of the function f, using the estimates of d_f, is defined as:

$$\hat{f} = W^T \hat{d}_f. \tag{12}$$

In the wavelet shrinkage approach, a big challenge is to find an appropriate threshold value λ (Raimondo, 2002). Note that when $\lambda = 0$ all the coefficient are kept, while $\lambda = \infty$ means that all the coefficients are shrunk. The thresholding rule works better if the thresholding value is specified well, see for example Nason (1996). Considering again Fig. 4 emphasises that this is a difficult aim to achieve as the blurring reduces the contrast in magnitudes between coefficients and the noise hides what differences remain.

Following the approach taken above for choosing the value of the penalty parameter κ, the best value for the threshold, λ, will be found by minimising the mean-squared error, that is:

$$\hat{\lambda} = \arg\min_{\lambda} MSE(\lambda). \quad (13)$$

Again, this is appropriate when there is training data or access to realistic simulated data.

A Two-Stage Wavelet-based Inversion Method

General

The previous sections have introduced two ideas, inverse problems and wavelet methods. The aim now is to combine them together to produce a novel method to analysis linear inverse problems and to investigate the interplay between the choice of penalty parameter, κ, in the inversion method and the threshold parameter, λ, in wavelet thresholding. Two approaches are studied which depend on the order of inversion and wavelet thresholding. In the first method, wavelet thresholding is used as a noise-reduction method before inversion with an expectation that this second stage will be better defined and hence more reliable. In the second method, inversion followed by wavelet thresholding is considered in the expectation that using a Haar wavelet in the final step will promote estimation as a step function.

Method 1: Thresholding then Inversion (TI)

The first step is to perform the wavelet thresholding, based on the Haar wavelet, to remove noise and hence to estimate $g = Kf$ - the noise-free data. This can be described, by a function T, as:

$$\hat{g}(\lambda) = T(y, \lambda)$$

which depends on threshold parameter λ. The second step is to perform the inversion. Suppose that this is represented by a function I so that:

$$\hat{f}(\kappa) = I\left(\hat{g}(\lambda), \kappa\right)$$

which depends on an inversion parameter κ. This may take the form of an explicit equation, such as Equation 6, or the numerical maximization of a likelihood. This two-stage process can be written in a single equation:

$$\hat{f}(\kappa, \lambda) = I\left(T(y, \lambda), \kappa\right).$$

The use of the double argument, (κ, λ), acknowledges the fact that the wavelet-inversion estimate depends on two parameters.

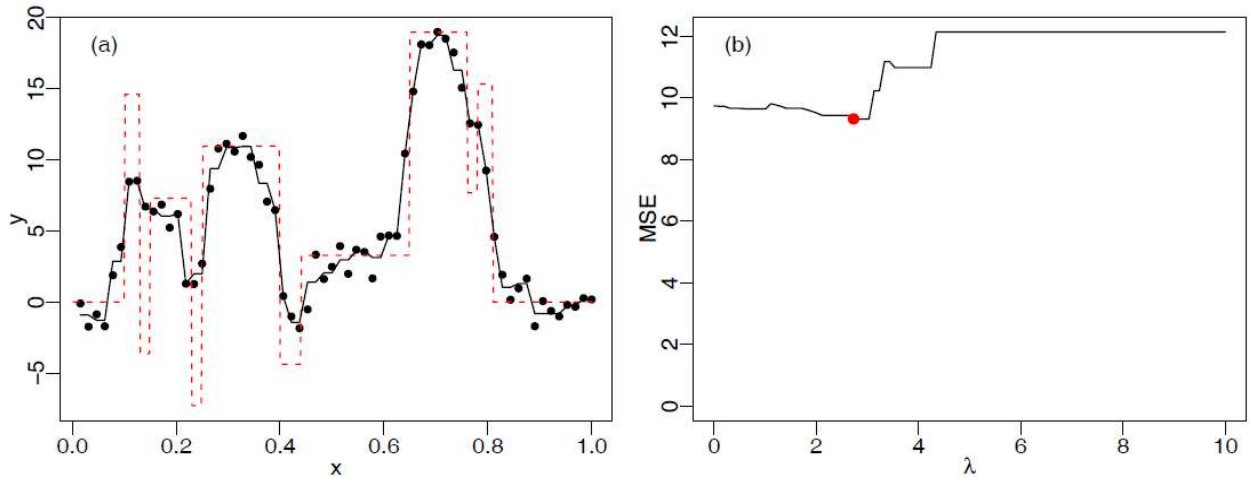
The value of the wavelet threshold, λ, is chosen as:

$$\hat{\lambda} = \arg\min_{\lambda} MSE(\lambda), \text{where } MSE(\lambda) = \frac{1}{m}\sum_{j=1}^{m}\left(\hat{g}_j(\lambda) - f_j\right)^2 \quad (14)$$

and then the value of the penalty parameter, κ, as:

$$\hat{\kappa} = \arg\min_{\kappa} MSE(\kappa), \text{ where } MSE(\kappa) = \frac{1}{m}\sum_{j=1}^{m}\left(\hat{f}_j\left(\kappa, \hat{\lambda}\right) - f_j\right)^2 \quad (15)$$

As illustration consider Figs. 5 and 6. When δ = 0.02 there are clear minimum values in the MSE allowing well-defined parameter estimates as $\hat{\lambda}$ = 2.73 and $\hat{\kappa}$ = 0.01. The corresponding mean squared errors are 9.32 and 6.65 compared to that of the data at 9.75. For the δ = 0.08 cases, in contrast, minimum values are less well defined and hence many values of the parameters will give similar function estimates. Here $\hat{\lambda}$ = 0.0 and hence the corresponding mean squared error and the estimate itself are the same as with the data with MSE of 26.18, then $\hat{\kappa}$ = 0.06 with mean squared error 20.76.



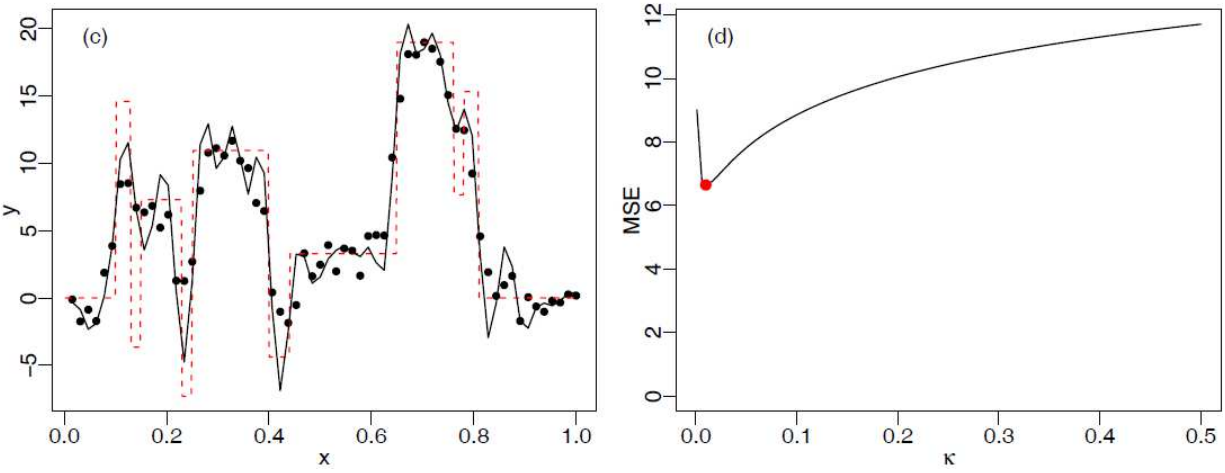


Fig. 5. Wavelet thresholding then inversion with δ = 0.02: (a) true function (dashed), data (points) and estimate (solid) after thresholding, (b) MSE(λ), (c) true function (dashed), data (points) and estimate (solid) after inversion, (d) MSE(κ)

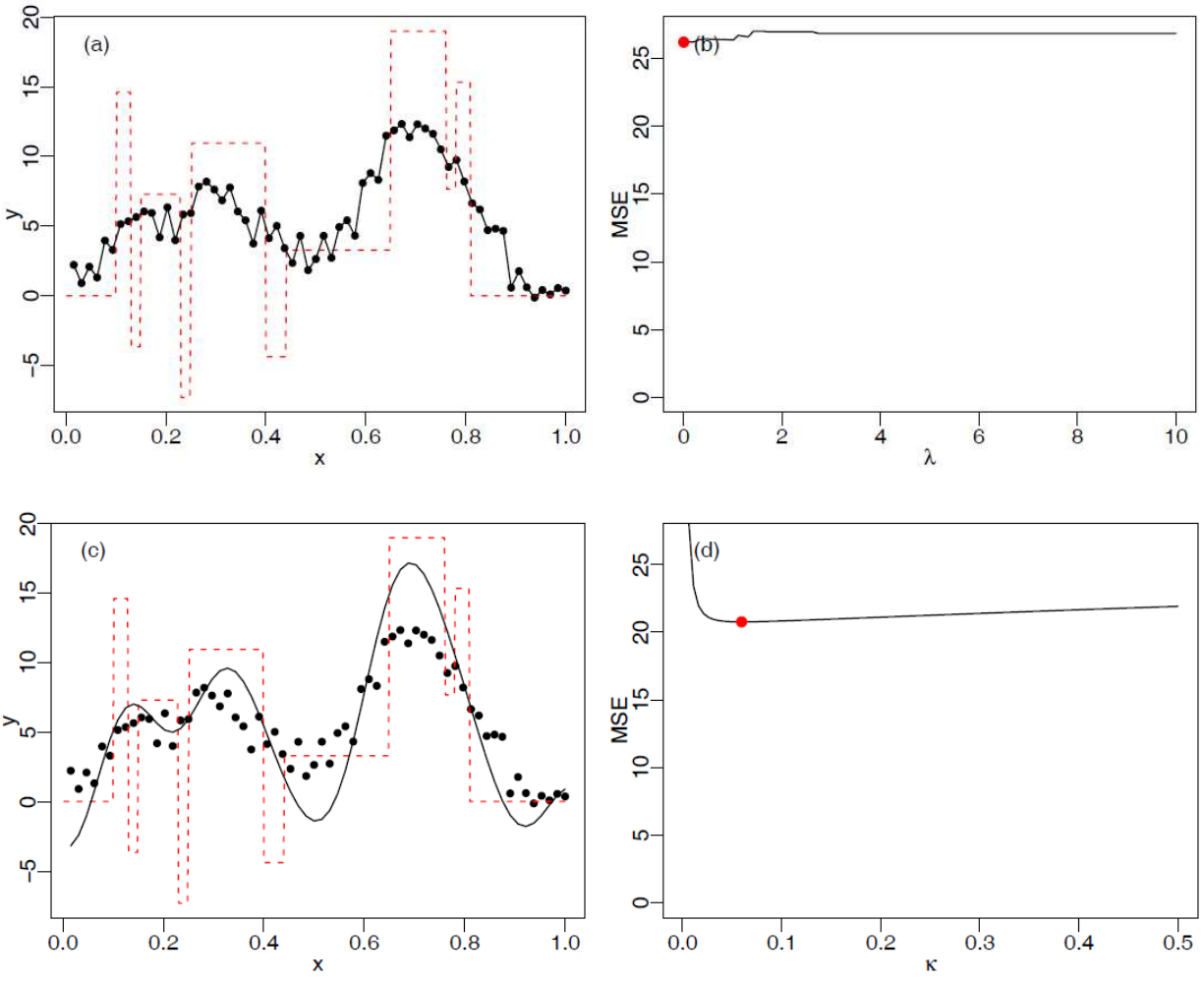


Fig. 6. Wavelet thresholding then inversion with δ = 0.08: (a) true function (dashed), data (points) and estimate (solid) after thresholding, (b) MSE(λ), (c) true function (dashed), data (points) and estimate (solid) after inversion, (d) MSE(κ)

The above approach involves sequential estimation of the penalty parameter κ and the wavelet threshold λ. Rather than this conditional approach, however, the two parameters could be found simultaneously, that is by joint minimisation of the mean squared error:

$$\left(\hat{\lambda},\hat{\kappa}\right)=\arg\min_{\lambda,\kappa} MSE\left(\lambda,\kappa\right),$$

$$MSE\left(\lambda,\kappa\right)\text{where}=\frac{1}{m}\sum_{j=1}^{m}\left(\hat{f}_j\left(\kappa,\lambda\right)-f_j\right)^2. \quad (16)$$

Although not illustrated here, this approach will be considered in the main simulation study in the next section.

Method 2: Inversion then Thresholding (IT)

The first step is to perform the inversion which, as before, is represented by a function I so that:

$$\hat{f}\left(\kappa\right)=I\left(y,\kappa\right) \quad (17)$$

which depends on an inversion parameter κ. Note that this time, the output of stage one is also a direct estimate of the underlying function f rather than of the intermediate function g. In stage two, wavelet thresholding is used to produce a sparse representation which is in the form of a step function. This can be described as:

$$\hat{f}\left(\kappa,\lambda\right)=T\left(\hat{f}\left(\kappa\right),\lambda\right) \quad (18)$$

which depends on threshold parameter λ. This two-stage process can then be written in a single equation:

$$\hat{f}\left(\kappa,\lambda\right)=T\left(I\left(y,\kappa\right),\lambda\right). \quad (19)$$

The value of the penalty parameter, κ is chosen as:

$$\hat{\kappa}=\arg\min_{\kappa} MSE\left(\kappa\right),\text{where } MSE\left(\kappa\right)=\frac{1}{m}\sum_{j=1}^{m}\left(\hat{f}_j\left(\kappa\right)-f_j\right)^2 \quad (20)$$

and the value of the wavelet threshold as:

$$\hat{\lambda}=\arg\min_{\lambda} MSE\left(\lambda\right),\text{where } MSE\left(\lambda\right)=\frac{1}{m}\sum_{j=1}^{m}\left(\hat{f}_j\left(\hat{\kappa},\lambda\right)-f_j\right)^2 \quad (21)$$

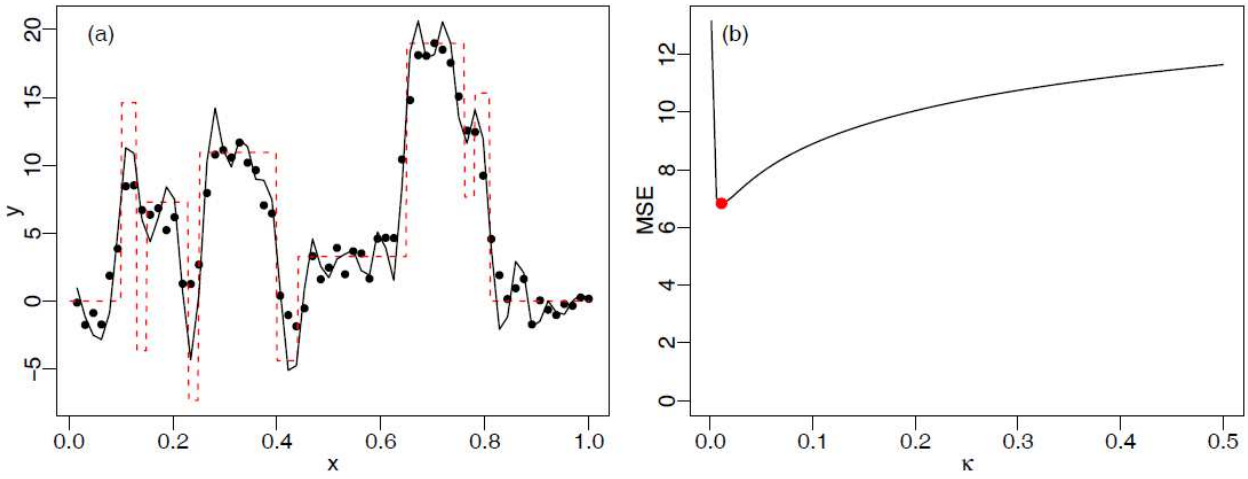
This approach is illustrated in Figs. 7 and 8 with δ = 0.02 and δ = 0.08 respectively. In each, (a) and (b) show the results of the inversions - in fact these are a repeat of Figs. 2 and 3. Then, (c) and (d) show the results of subsequently applying wavelet thresholding to the results of the inversion. For δ = 0.02, $\hat{\lambda}$ = 4.24 and $\hat{\kappa}$ = 0.01, leading to MSE values of 6.83 and 6.18 after stages one and two respectively, compared to a MSE of the data of 9.75. Corresponding values with δ = 0.08 are $\hat{\lambda}$ = 1.01 and $\hat{\kappa}$ = 0.06, leading to MSE values of 20.75 and 20.70 after stages one and two respectively, compared to a MSE of the data of 26.18. For each value of δ there is a clear visual improvement in the final estimate compared to that after only the inversion. In that a better defined step-function is produced - this is especially worthwhile in the moderate blurring case. As with the inversion penalty parameters, κ, the minimum in the MSE is better defined when the blurring is moderate compared to large.

Again, rather than sequential estimation of the parameters, estimates can be found simultaneously, that is by joint minimisation of the mean squared error:

$$\left(\hat{\lambda},\hat{\kappa}\right)=\arg\min_{\lambda,\kappa} MSE\left(\lambda,\kappa\right),$$

$$\text{where } MSE\left(\lambda,\kappa\right)=\frac{1}{m}\sum_{j=1}^{m}\left(\hat{f}_j\left(\kappa,\lambda\right)-f_j\right)^2. \quad (22)$$

This approach will also be considered in the main simulation study in the next section.



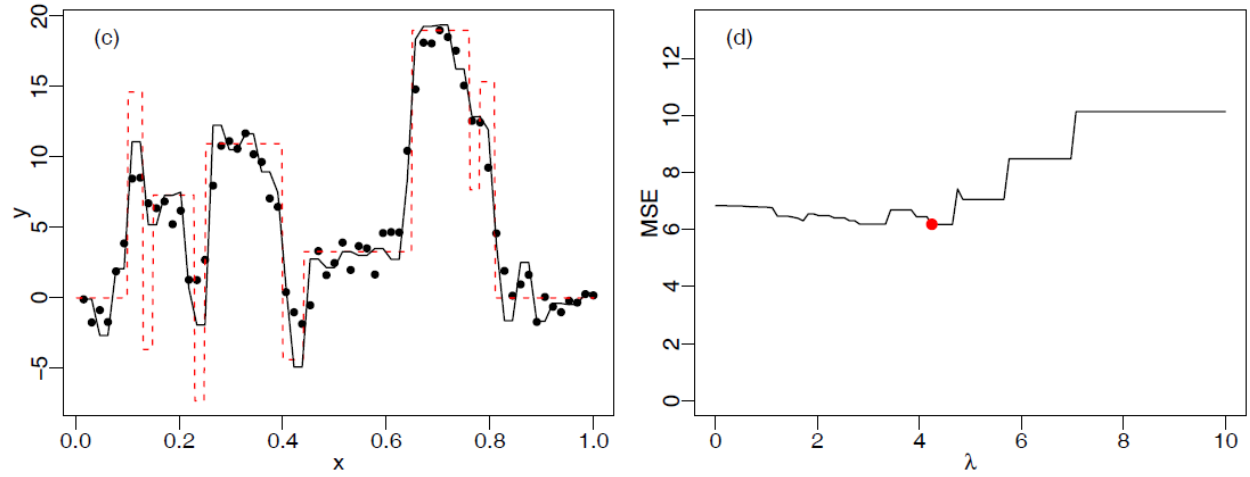


Fig. 7. Inversion then wavelet thresholding with δ = 0.02: (a) true function (dashed), data (points) and estimate (solid) after inversion, (b) MSE(κ), (c) true function (dashed), data (points) and estimate (solid) after thresholding, (d) MSE(λ)

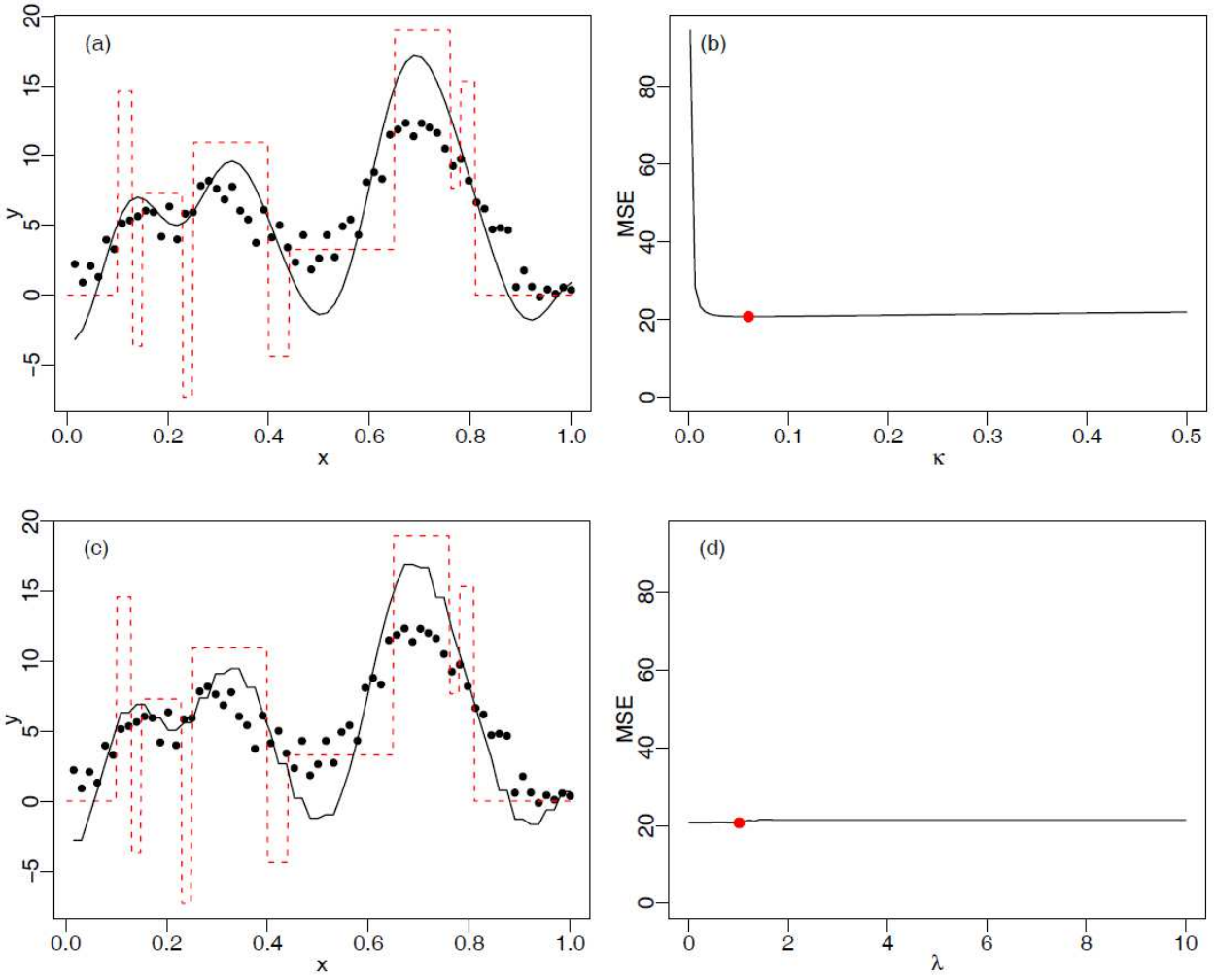


Fig. 8. Inversion then wavelet thresholding with δ = 0.08: (a) true function (dashed), data (points) and estimate (solid) after inversion, (b) MSE(κ), (c) true function (dashed), data (points) and estimate (solid) after thresholding, (d) MSE(λ)

A Simulation Study of Wavelet-Inversion Methods

The illustrative results in the previous section have given an indication of the properties of the two basic methods proposed, that is (1) wavelet Thresholding then Inversions (TI) and (2) Inversion then wavelet Thresholding (IT). To compare the estimates more precisely, however, the whole procedure will be replicated $M = 100$ times and boxplots used to compare the various examples.

Method 1: Wavelet Thresholding then Inversion (TI)

Figure 9 shows results for the two stage approach of wavelet thresholding then inversion where parameters λ and κ are chosen sequentially. In (a), the grey boxplots show the MSE after only the first stage of wavelet thresholding involving the estimation of threshold λ as shown in (b). There is a clear increase in the MSE as δ increases. Also, although there is a great spread in estimated λ values, the first few are reasonably consistent at about 2-2.5, then a substantial drop to around 1.5 for higher δ values. This reflect the effect of blurring on the true wavelet coefficients where large values get reduced as δ increases. Hence, the best threshold also reduces otherwise true coefficients are removed. In balance this also means that more noise remains. The black boxplots in (a) show the MSE after the second stage of inversion is completed and (c) shows the corresponding penalty parameter. Initially, that is for small δ, there is little improvement in the MSE due to the inversion, but as δ increases the effect if more substantial - as expected. Similarly, this is apparent in the estimates of κ where initially they are close to zero but then increase. Note that the reciprocal conditional number for $\delta = 0.02$ is 6×10^{-4}, which then jumps to 2×10^{-8} for $\delta = 0.03$ indicating a move from mildly to severely ill-conditioned.

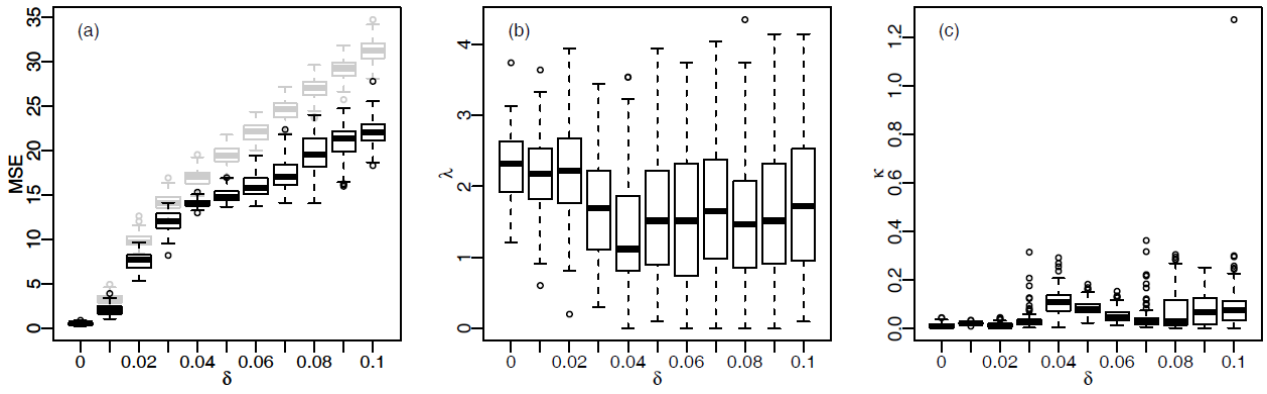


Fig. 9. Wavelet then inversion results showing boxplots: (a) MSE after thresholding (grey) and then after inversion (black) and estimated parameters (b) $\hat{\lambda}$ and (c) $\hat{\kappa}$

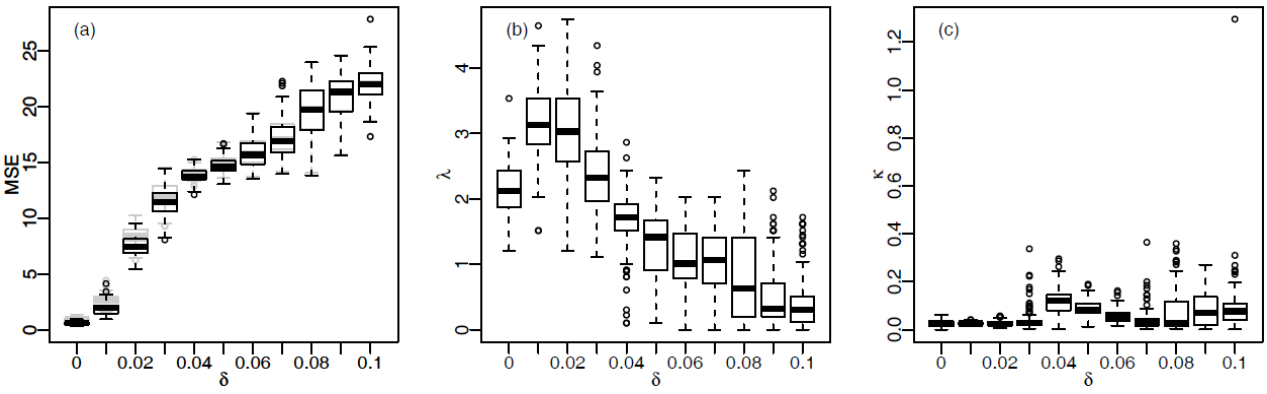


Fig. 10. Inversion then wavelet results showing boxplots: (a) MSE after inversion (grey) and then after thresholding (black) and estimated parameters (b) $\hat{\lambda}$ and (c) $\hat{\kappa}$

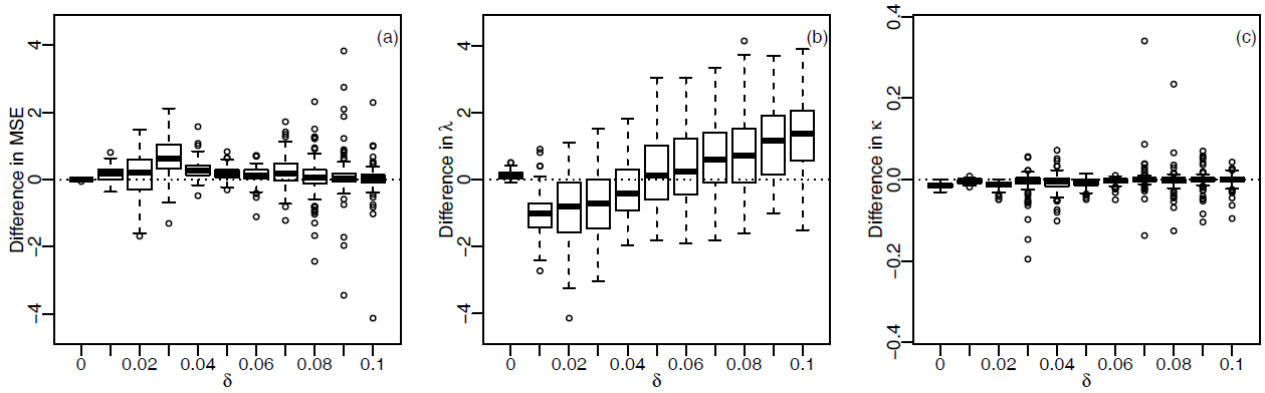


Fig. 11. Comparison of Method 1 and Method 2 with sequential estimation of parameters - a negative value indicates that Method 1 has a larger value.

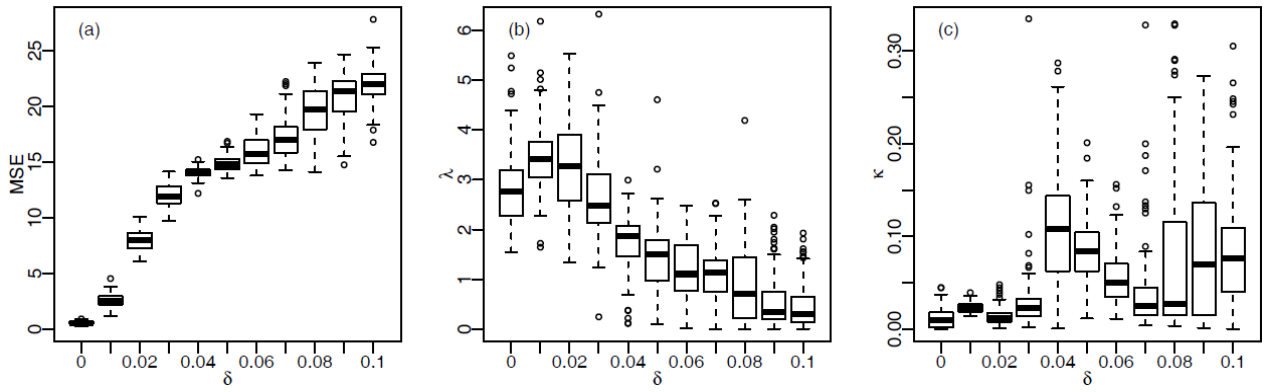


Fig. 12. Wavelet then inversion results, with joint estimation of parameters, showing boxplots: (a) MSE and estimated parameters (b) λ and (c) κ

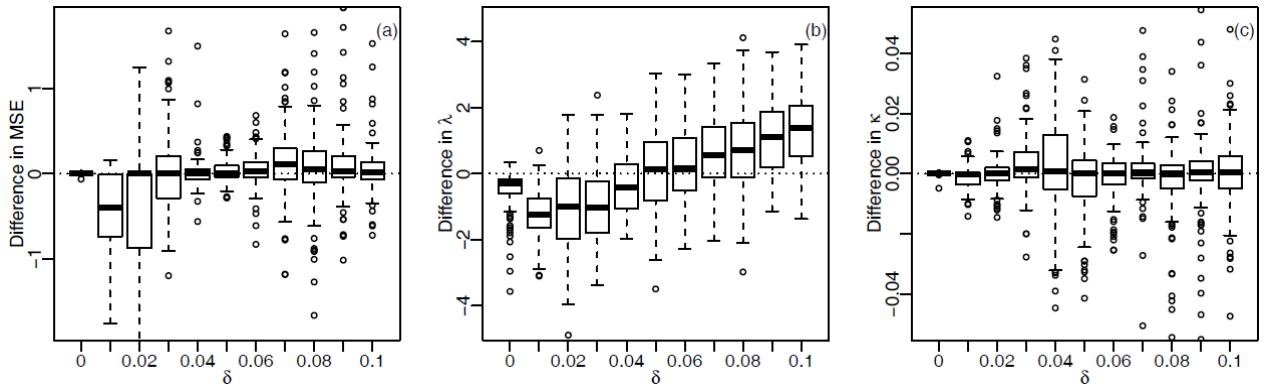


Fig. 13. Wavelet then inversion results, with joint estimation of parameters, showing boxplots improvement due to simultaneous estimation: (a) MSE and estimated parameters (b) λ and (c) κ - a negative value indicates a higher value for sequential estimation

Method 2: Inversion then Wavelet Thresholding (IT)

Similar results for Method 2 of inversion then wavelet thresholding are shown in Fig. 10. This time there is very little difference in the MSE values at the end of Stage 1 and Stage 2. The greatest benefit in terms of MSE is for small to moderate δ values, for example up to about 0.03 or 0.04. There is a very noticeable pattern in the estimated λ values used in the wavelet thresholding which is much more pronounced than in Fig. 10. Finally, Fig. 11 shows a comparison of the two

methods. In (a) there is a clear improvement in terms of MSE for δ in the range 0.0-0.03 by performing Inversion then wavelet Thresholding (IT) over vice versa (TI). There is also a clear pattern in the estimated λ values but nothing noticeable in the κ values. This indicates that inversion then wavelet thresholding is the best method in terms of MSE, but more importantly in terms of producing a function estimate resembling a step function.

Comparison of Joint Estimation of λ and κ

Before making final conclusions, in this section simultaneous estimation for the parameters λ and κ is considered. Figure 12 shows the results using wavelet thresholding then inversion with joint estimation of the wavelet threshold λ and the penalty parameter κ. Fig. 13 compares the results with those from the separate sequential estimation of λ and κ. Given the very wide variability it is difficult to conclude more than that there is general agreement between the methods, but there are some consistent patterns which are worthy of comment. From Fig. 13(a) the median MSE is slightly better for joint estimation for small δ but very slightly worse for large δ. In (b) the joint estimate of λ is smaller for small values of δ and larger for larger values of δ in the joint estimation compared to the sequential. Finally, the values of $\hat{\kappa}$, compared in (c), are much more similar, though there is less variability for small δ.

Figures 14 and 15 show similar comparisons for Method 2, that is inverse then wavelet thresholding, using joint estimation of λ and κ and compared to sequential estimation. In Fig. 14 there are very similar MSE values in (a) and estimates of κ in (b), but a different pattern in the λ values in (c). In Fig. 15, the improvements in MSE due to simultaneous estimation are clearly seen in (a) as in almost all cases there is a reduction in MSE. This appears to be mainly due to a change in the estimated wavelet threshold λ with smaller values for small δ and larger values for larger δ. There is, perhaps, an indication of smaller κ values in the joint estimation case. Hence, for this method there is a worthwhile improvement performing joint estimation of λ and κ compared to the sequential approach.

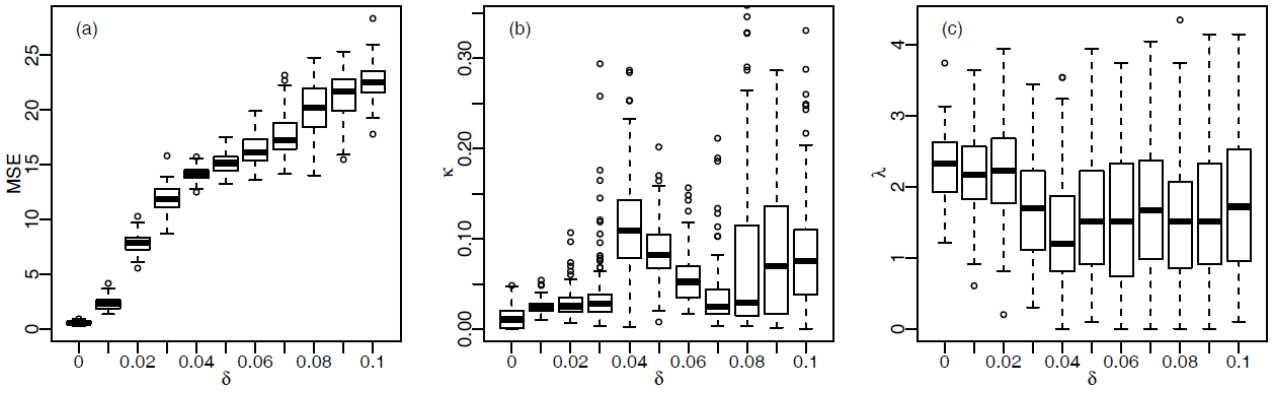


Fig. 14. Inversion then wavelet results, with joint estimation of parameters, showing boxplots: (a) MSE and estimated parameters (b) λ and (c) κ

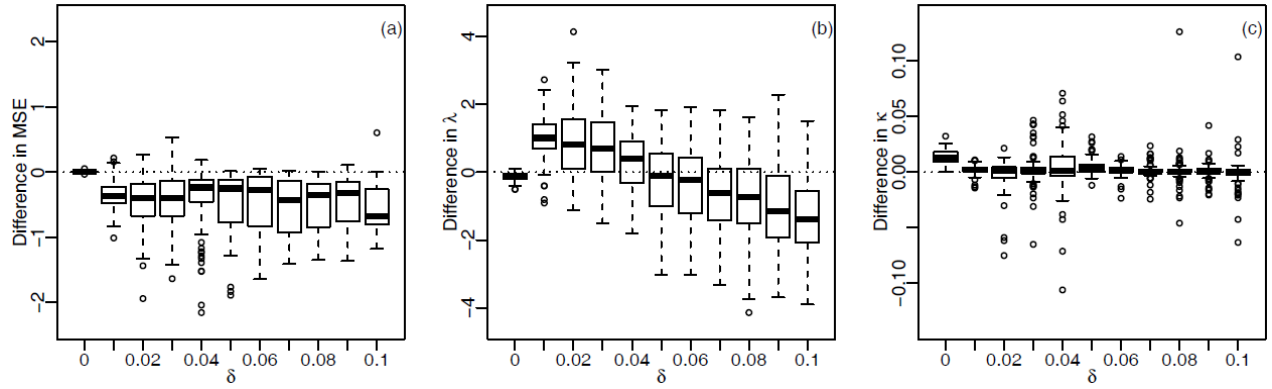


Fig. 15. Inversion then wavelet results, with joint estimation of parameters, showing boxplots improvement due to simultaneous estimation: (a) MSE and estimated parameters (b) λ and (c) κ - a negative value indicates a higher value for sequential estimation

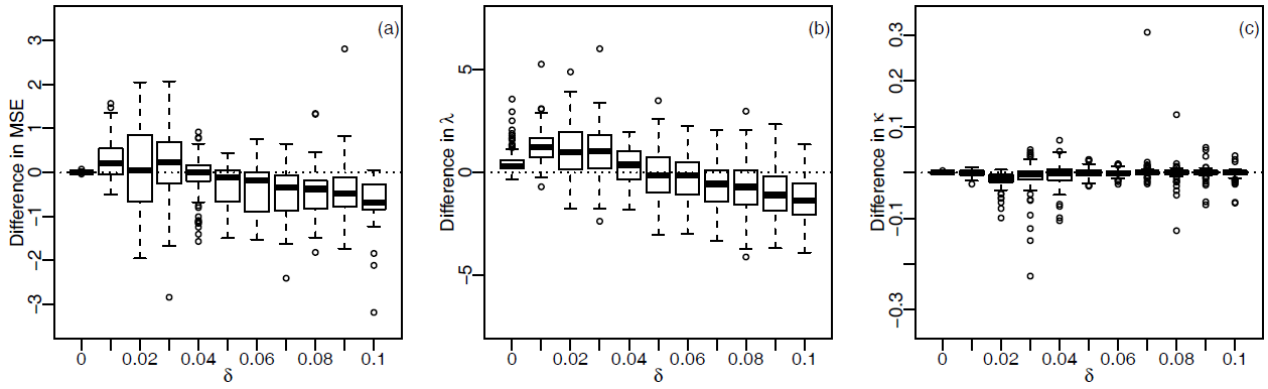


Fig. 16. Comparison of simultaneous estimation of λ and κ in terms of (a) MSE, (b) $\hat{\lambda}$ and (c) $\hat{\kappa}$ - a negative value indicates a higher value for IT estimation

Figure 16 shows the final results which compare the MSE and the two sets of joint parameter estimates. In (a) the MSE is initially better for wavelet thresholding then inversion but for larger δ values inverse then wavelet thresholding is better. For smaller δ the estimated threshold $\hat{\lambda}$ in (b) is larger for wavelet thresholding then inversion but smaller for larger δ values. There is no substantial pattern visible amongst the variability in (c) for the estimated κ.

Discussion

The aim of this work was to investigate the use of wavelet-based models for the estimation of piecewise constant functions in inverse problems. The nature of inverse problems means that some of the attractive computational properties of wavelets are lost, but they still present a useful modelling tool. Inverse problems are widely encountered in the applied sciences and assumptions of piecewise constant, or at least piecewise smooth functions, are appropriate. It is common, however, to use prior distributions on the function values themselves which usually lead to poor reconstruction-shrinkage type models move in the estimates towards zero whilst smoothing priors destroy sharp discontinuities. Hence, the approach proposed here has the potential to have significant impact on a wide range of practical problems.

Conclusion

From the results it is clear that for this type of function the best method is to use inversion then wavelet thresholding. This leads to a function estimate which more closely resembles a step function and generally has a smaller mean squared error. Although sequential estimation of the two parameters, λ in the thresholding and κ in the likelihood penalty function, is satisfactory there is still a further benefit from estimating them together. It is worth saying that for larger problems, then the sequential estimation is quicker and potentially more reliable than joint estimation. The comparisons here have estimated parameters by minimum mean squared error which is feasible when training data are available or for when realistic simulations can be performed, but in other situations other estimation approaches would be preferable. This is the theme of further work in this area. Also, it is our intention to evaluate the procedures on real data problems, in particular application to archaeological stratigraphy where data is 1D and a segmentation into occupation layers is required.

Acknowledgment

The authors thank the Editor and referees for their helpful comments.

Author's Contributions

Both authors participated in all aspects of the research and the writing of the article.

Ethics

The authors confirm that this article is original work and contains unpublished material. The authors declare no conflict of interest and there are no ethical issues involved.

References

Allum, G., R. Aykroyd and J. Haigh, 1999. Empirical Bayes estimation for archaeological stratigraphy. J. R. Stat. Society, 48: 1-14.
DOI: 10.1111/1467-9876.00135

Aykroyd, R.G., 2015. Industrial Tomography: Systems and Applications. 1st Edn., Woodhead Publishing, ISBN-10: 1782421181, pp: 772.

Donoho, D.L. and J.M. Johnstone, 1994. Ideal spatial adaptation by wavelet shrinkage. Biometrika, 81: 425-455. DOI: 10.1093/biomet/81.3.425

Golub, G. and C.F. Van Loan, 1989. Matrix Computations. 2nd Edn., Johns Hopkins, Baltimore.

Hadamard, J., 2014. Lectures on Cauchy's problem in linear partial differential equations. Courier Corporation.

Hoerl, A.E. and R.W. Kennard, 1970. Ridge regression: Biased estimation for nonorthogonal problems. Technometrics, 12: 55-67. DOI: 10.1080/00401706.1970.10488634

Nason, G., 2010a. Wavelet methods in statistics with R. Springer.

Nason, G., 2010b. Wavethresh 4.5. Wavelets, statistics and transforms. R package.

Nason, G.P., 1996. Wavelet shrinkage using cross-validation. J. R Stat. Society, 58: 463-479.

R Core Team, 2016. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. www.R-project.org

Raimondo, M., 2002. Wavelet shrinkage via peaks over threshold. Inter. Stat.

Tibshirani, R., 1996. Regression shrinkage and selection via the lasso. J. R Stat. Society, 58: 267-288.

Vidakovic, B., 2009. Statistical modeling by wavelets. John Wiley and Sons.

Vidakovic, B. and P. Mueller, 1994. Wavelets for kids: A tutorial introduction. Technical report, Duke University.

Young, R.K., 1993. Wavelet theory and its applications. Springer Science and Business Media.

Zou, H. and T. Hastie, 2005. Regularization and variable selection via the elastic net. J. R Stat. Society, 67: 301-320. https://www.jstor.org/stable/3647580