Multivariate Denoising using Wavelets and Principal Component Analysis

Mina AminGhafari G.S. [a,b], Nathalie Cheze [a,c],
Jean-Michel Poggi [a,d,*]

[a] *Laboratoire de Mathématique – U.M.R. C 8628, "Probabilités, Statistique et Modélisation", Université Paris-Sud, Bât. 425, 91405 Orsay Cedex, France*

[b] *Teheran Polytechnic University*

[c] *Université Paris 10-Nanterre, Modal'X*

[d] *Université Paris 5*

Abstract

This paper proposes a multivariate extension of the well known wavelet denoising procedure widely examined for scalar valued signals. It combines a straightforward multivariate generalization of a classical one and principal component analysis. This new procedure exhibits promising behavior on classical bench signals and the associated estimator is found to be near minimax in the one-dimensional sense, for Besov balls.

Key words: Multivariate, Denoising, Wavelets, PCA

Résumé

Cet article propose une extension au cas multivarié, des procédures de débruitage par ondelettes, bien connues pour les signaux unidimensionnels. Cette nouvelle procédure combine une généralisation directe au cas multidimensionnel d'une stratégie classique, avec l'analyse en composantes principales. Cette méthode fournit des résultats prometteurs sur des signaux tests et l'estimateur associé est quasi-minimax, au sens unidimensionnel, sur des boules de Besov.

Mots clés: Multidimensionnel, Débruitage, Ondelettes, ACP

* Corresponding author.
Email addresses: `Mina.Ghafari@math.u-psud.fr` (M. AminGhafari), `Nathalie.Cheze@u-paris10.fr` (N. Cheze), `Jean-Michel.Poggi@math.u-psud.fr` (J-M. Poggi).

1 Introduction

In one hand, denoising algorithms based on wavelets become a popular method for one-dimensional statistical signal extraction and filtering. In the other hand, principal component analysis is one of the notorious data-analysis tool especially designed to simplify multidimensional data by tracking new factors capturing the main features.

Classical univariate wavelet denoising dealing both with signal processing and functional estimation in statistics, is of interest in various applied fields. Valuable references are the books [11], [13], [14], [19] and the survey paper [1]. For basics on wavelets, the reader can refer to [11] or [12] for example.
The simplest considered model is of the following form:

$$X(t) \ = \ f(t) \ + \ \varepsilon(t), \ t = 1, ..., n \tag{1}$$

where $(X(t))_{1 \leq t \leq n}$ is observed, $(\varepsilon(t))_{1 \leq t \leq n}$ is a centered Gaussian white noise of unknown variance σ^2 and $f \in L^2$ is an unknown function to be recovered through the observations.

For a given orthogonal wavelet and a suitably chosen decomposition level J, wavelet denoising proceeds in three steps:

- step 1. Compute the wavelet decomposition of the observed signal up to level J;
- step 2. Threshold conveniently the wavelet detail coefficients;
- step 3. Reconstruct a denoised version of the original signal, from the thresholded detail coefficients and the approximation coefficients, using the inverse wavelet transform.

Various strategies are available (see the survey paper [2]) to perform this task and the performance of the associated estimators is the minimax one up to a logarithmic factor, for large classes of functions simultaneously. For simplicity and since only relative performance between the proposed multivariate procedures are of interest, we restrict our attention to the so-called universal threshold (introduced by Donoho and Johnstone [8]) which is of the form $\hat{\sigma}\sqrt{2log(n)}$, where $\hat{\sigma}$ is an estimator of σ based on the detail coefficients at level 1. Such methods are effective because functions f belonging to various general classes are such that they admit a sparse wavelet representation (see [10]). So the energy of f is mainly concentrated in a few large wavelet coefficients which are adaptively selected by this procedure since the coefficients below the threshold are attributable to the additive noise.

This paper proposes a multivariate extension of wavelet denoising procedures widely examined for scalar valued signals. It combines a straightforward multivariate generalization of the classical one sketched above and principal component analysis.

The paper is divided in two main sections. Section 2 proposes a first denoising procedure which is a direct generalization of the one-dimensional strategy. This new procedure exhibits promising behavior on classical test signals and the associated estimator is found to be near minimax in the one-dimensional sense, for Besov balls. Section 3 first recalls the multiscale PCA proposed by Bakshi [3] for statistical process control purposes. This scheme is discussed and then a second denoising procedure combining wavelets and PCA is proposed and illustrated by some simulation examples.

2 Multivariate wavelet denoising

2.1 *Procedure*

Let us consider the following p-dimensional model:

$$X(t) \;=\; f(t) \;+\; \varepsilon(t), \; t = 1, ..., n \tag{2}$$

where $X(t), f(t), \varepsilon(t)$ are of size $1 \times p$ and $(\varepsilon(t))_{1 \leq t \leq n}$ is a centered Gaussian white noise with unknown covariance matrix $E(\varepsilon^T \varepsilon) = \Sigma_\varepsilon$. Each component of $X(t)$ is of the previous form (1), for $1 \leq i \leq p$:

$$X^i(t) \;=\; f^i(t) \;+\; \varepsilon^i(t), \; t = 1, ..., n$$

where $f^i \in L^2$.
The covariance matrix Σ_ε, supposed to be positive definite, captures the stochastic link between the components of $X(t)$.

The first proposal for multivariate denoising is described as follows, starting from an $n \times p$ matrix X containing p signals (the columns of X) of dyadic length n supposed to be such that $n >> p$.

- step 1. Perform the wavelet decomposition at level J of each column of X. This step produces $J+1$ matrices $D_1, ..., D_J$ containing the detail coefficients at level 1 to J of the p signals and the approximation coefficients A_J of the p signals. Matrices D_j and A_J are respectively of size $n2^{-j} \times p$ and $n2^{-J} \times p$;
- step 2. Define $\hat{\Sigma}_\varepsilon$ an estimator of the noise covariance matrix and then compute an orthogonal matrix V such that $\hat{\Sigma}_\varepsilon = V \Lambda V^T$ where $\Lambda = diag(\lambda_i, 1 \leq$

$i \leq p$).
Apply to each detail after change of basis (namely $D_j V$, $1 \leq j \leq J$), the p univariate thresholding strategies using the threshold $t_i = \sqrt{2\lambda_i log(n)}$ for the i-th column of $D_j V$;

- step 3. Reconstruct a denoised matrix $\check{X}$ (or equivalently an estimator $\hat{f}$ of f), from the simplified detail and approximation matrices, by inverting the wavelet transform.

Since the wavelet transform (in time) and the change of basis (in space) commute, it appears that this straightforward generalization of the univariate denoising is equivalent to: first, to change of basis to decorrelate the p components and second, to apply p univariate wavelet denoising. Using the orthogonal matrix V defined in step 2, let us change of basis the model (2) by defining $\tilde{X}(t) = X(t)V$ and similarly $\tilde{f}(t)$ and $\tilde{\varepsilon}(t)$, leading to

$$\tilde{X}(t) = \tilde{f}(t) + \tilde{\varepsilon}(t), \quad t = 1, ..., n \tag{3}$$

which are p near uncorrelated components as soon as $\hat{\Sigma}_\varepsilon$ is a consistent estimator of Σ_ε. Then it suffices to denoise each component and to change of basis using V^T to get back to the original one.

Surprisingly this scheme performs better than the direct one performing the denoising of each component of (2) ignoring correlation structure, this will be illustrated in section 2.4. Indeed, since $trace(F\hat{\Sigma}_\varepsilon F^T) = trace(\tilde{F}\Lambda\tilde{F}^T)$, where F and $\tilde{F}$ are defined as in (7) and (8), it turns out that the p signal to noise ratios after change of basis are not necessarily better than initial ones. It seems that the crucial point is that the change of basis merges all the original functions in each $\tilde{f}^i$ introducing a kind of redundancy leading to better estimate the main features (such as ruptures or singularities) of the original signals f^i to be recovered.

2.2 Theoretical results

In this section devoted to theoretical results, we consider the estimator, denoted by $\hat{f}^*$, introduced in section 2.1 but where the noise covariance matrix is supposed to be known and where soft thresholding is performed with the function

$$d_t(x) = sgn(x)(|x| - t)_+ \tag{4}$$

where t is the positive real threshold.
First, we compute the quadratic risk of the associated estimator which is useful

for a given sample path.
Second, we adopt the minimax viewpoint in order to examine the estimation procedure performance when the functions f^i belong to Besov balls. We prove that our estimator is near minimax in the one-dimensional sense but inherits the worst regularity rate among the p functions.

The univariate loss function, is defined by:

$$R(f^i, \hat{f}^i) = \sum_{m=1}^{n} E(f^i(m) - \hat{f}^i(m))^2 \tag{5}$$

A straightforward extension to the multivariate case is the sum over the p functions:

$$R_M(f, \hat{f}) = \sum_{i=1}^{p} R(f^i, \hat{f}^i) \tag{6}$$

2.2.1 Risk computation

Let X be the $n \times p$ matrix of the n observations from the p-dimensional vector $X(t)$, then X can be written as

$$X = F + \varepsilon \tag{7}$$

where F and ε are $n \times p$ matrices, associated with $f(t)$ and $\varepsilon(t)$ respectively. Let V be the orthogonal matrix such that $V^T \Sigma_\varepsilon V$ is diagonal. Then, we have:

$$XV = \tilde{X} = \tilde{F} + \tilde{\varepsilon} \tag{8}$$

For a given orthogonal compactly supported wavelet, the discrete wavelet transform (DWT) of the matrix $\tilde{X}$ can be written as follows:

$$W\tilde{X} = W\tilde{F} + W\tilde{\varepsilon} \tag{9}$$

where W is an $n \times n$ orthogonal matrix defined using the wavelet and scaling filters impulse responses (see [14], page 57). Let us denote by Y^i and Y_m the i-th column and the m-th row of matrix Y respectively. Then, $W_m \tilde{X}^i$ and $W_m \tilde{F}^i$ are the m-th discrete wavelet transform coefficients of $\tilde{X}^i$ and $\tilde{F}^i$ respectively.

Proposition 1 *Let $I(i) = \{1 \leq m \leq n, |W_m \tilde{X}^i| > t_i\}$. Then, the risk of estimator $\hat{f}^*$ is given by:*

$$R_M(f,\hat{f}^*) = \sum_{i=1}^{p} \left(card(I(i))(\lambda_i + {t_i}^2) + \sum_{m \notin I(i)} (W_m \tilde{F}^i)^2 \right) \tag{10}$$

Proof The estimator $\hat{f}^*$ is defined using the p estimators $\hat{\tilde{f}}^i$ after the change of basis using V. Then using (8) and (9), $\hat{f}^*$ leads to $\hat{F}$. Matrices $\hat{F}$ and $\hat{\tilde{F}}$ are related by:

$$\hat{F} = \hat{\tilde{F}} V^T \tag{11}$$

where the estimator $\hat{\tilde{F}}$ of $\tilde{F}$ is such that $\hat{\tilde{F}} = W^T U$. The components of the n-dimensional column vector U^i are $d_{t_i}(W_m \tilde{X}^i)$ where d_{t_i} is the soft thresholding function defined in (4) with threshold t_i.
The risks in the new basis and in the initial basis are the same:

$$R_M(f,\hat{f}) = R_M(\tilde{f},\hat{\tilde{f}}) \tag{12}$$

Indeed,

$$\begin{aligned}
R_M(f,\hat{f}) &= trace\left(E\left((F-\hat{F})(F-\hat{F})^T\right)\right) \\
&= trace\left(E\left((F-\hat{F})VV^T(F-\hat{F})^T\right)\right) \\
&= trace\left(E\left((\tilde{F}-\hat{\tilde{F}})(\tilde{F}-\hat{\tilde{F}})^T\right)\right) \\
&= trace\left(E\left((\tilde{F}-\hat{\tilde{F}})^T(\tilde{F}-\hat{\tilde{F}})\right)\right) \\
&= \sum_{i=1}^{p} \left(E\left((\tilde{F}^i-\hat{\tilde{F}}^i)^T(\tilde{F}^i-\hat{\tilde{F}}^i)\right)\right) = \sum_{i=1}^{p} R(\tilde{f}^i,\hat{\tilde{f}}^i).
\end{aligned}$$

Since W is an orthogonal matrix, we can derive:

$$\begin{aligned}
R(\tilde{f}^i,\hat{\tilde{f}}^i) &= E\left((\tilde{F}^i-\hat{\tilde{F}}^i)^T(\tilde{F}^i-\hat{\tilde{F}}^i)\right) \\
&= E\left((\tilde{F}^i-\hat{\tilde{F}}^i)^T W^T W(\tilde{F}^i-\hat{\tilde{F}}^i)\right) \\
&= E\left((W\tilde{F}^i-U^i)^T(W\tilde{F}^i-U^i)\right) \\
&= \sum_{m=1}^{n} E\left(W_m\tilde{F}^i - d_{t_i}(W_m\tilde{X}^i)\right)^2
\end{aligned}$$

where $(W_m\tilde{F}^i - d_{t_i}(W_m\tilde{X}^i))^2 = \begin{cases} (W_m\tilde{\varepsilon}^i + t_i \quad \mathrm{sgn}(W_m\tilde{X}^i))^2 & \text{if } |W_m\tilde{X}^i| > t_i \\ (W_m\tilde{F}^i)^2 & \text{if } |W_m\tilde{X}^i| \leq t_i \end{cases}$
and $E\,(W_m\tilde{\varepsilon}^i)^2 = \lambda_i$ since $E\left(\tilde{\varepsilon}^T\tilde{\varepsilon}\right) = \Lambda$. Then we obtain (10). ∎

2.2.2 Near Minimaxity

The risk (10) depends on f. Then, it is traditional in nonparametric estimation to calculate its maximum over a smoothness class and to compare it to the risk of minimax estimators. We consider here functions belonging to balls of Besov spaces denoted $B^{\alpha}_{r,q}$ with norm $||.||^{\alpha}_{r,q}$, $r,q \geq 1$.

The following result is based on theorem 1.2 of Donoho [7] assuming that the considered orthogonal wavelet ψ belongs to the class C^k and has d vanishing moments, for some positive integers k and d.

Theorem 2 *Suppose that ε is a Gaussian white noise with covariance matrix Σ_ε. Let $f_i \in \mathcal{F}_C, i = 1, ..., p$ where $\mathcal{F}_C$ denotes the Besov ball of scalar valued functions $\{g, \parallel g \parallel^{\alpha}_{r,q} \leq C\}$ with $\frac{1}{r} < \alpha < min(k,d)$ and $C > 0$. Then, there is a positive constant denoted by cte depending on $\mathcal{F}_C$ and ψ, such that*

$$\sup_{f\in(\mathcal{F}_C)^p} R_M(f,\hat{f}^*) \leq cte\ p\ log(n)\ \inf_{\hat{g}} \sup_{g\in\mathcal{F}_C} R(g,\hat{g}) \tag{13}$$

where $\hat{f}^*$ is the wavelet based estimator described previously using universal soft thresholding defined in step 2 (see section 2.1).

Proof Since $\tilde{f}^i = \sum_{j=1}^{p} V^i_j f^j$, $f^j \in \mathcal{B}^{\alpha}_{r,q}$ implies that $\tilde{f}^i \in \mathcal{B}^{\alpha}_{r,q}$ (see Devore and Lorentz [6], pages 54-56). More precisely, $f^j \in \mathcal{F}_C, j = 1, ..., p$, implies that $\tilde{f}^i \in \mathcal{F}_C$. Indeed, $\parallel \tilde{f}^i \parallel^{\alpha}_{r,q} \leq \sum_{j=1}^{p} (V^i_j)^2 \parallel f^j \parallel^{\alpha}_{r,q} \leq C \sum_{j=1}^{p} (V^i_j)^2 = C$.

For each i, let $\hat{\tilde{f}}^{*i}$ be the estimator of $\tilde{f}^i$ induced by the definition of $\hat{f}^{*i}$. Then, $\hat{\tilde{f}}^{*i}$ is the estimator considered in the near minimaxity theorem of Donoho [7] with threshold $t_i = \sqrt{2\lambda_i log(n)}$.

Furthermore, using (12), the risk can be written as follows:

$$\sup_{f\in(\mathcal{F}_C)^p} R_M(f,\hat{f}^*) \;=\; \sup_{(f^j\in\mathcal{F}_C)_{1\leq j\leq p}} \sum_{i=1}^{p} R(f^i,\hat{f}^{*i}) \;=\; \sup_{(f^j\in\mathcal{F}_C)_{1\leq j\leq p}} \sum_{i=1}^{p} R(\tilde{f}^i,\hat{\tilde{f}}^{*i})$$

then, since $\tilde{f}^i \in \mathcal{F}_C$, we have

$$\sup_{f\in(\mathcal{F}_C)^p} R_M(f,\hat{f}^*) \;\leq\; \sup_{(\tilde{f}^j\in\mathcal{F}_C)_{1\leq j\leq p}} \sum_{i=1}^{p} R(\tilde{f}^i,\hat{\tilde{f}}^{*i})$$

So by the near minimaxity theorem, we get

$$\begin{aligned}\sup_{f\in(\mathcal{F}_C)^p} R_M(f,\hat{f}^*) &\leq \sum_{i=1}^{p} \sup_{\tilde{f}^i\in\mathcal{F}_C} R(\tilde{f}^i,\hat{\tilde{f}}^{*i})\\ &\leq p\ cte\ log(n)\ \inf_{\hat{\tilde{f}}^i}\ \sup_{\tilde{f}^i\in\mathcal{F}_C} R(\tilde{f}^i,\hat{\tilde{f}}^i)\end{aligned} \quad \blacksquare$$

So the multivariate estimator $\hat{f}^*$ is near minimax in the one-dimensional sense. This is expected because it estimates p scalar-valued functions.

Remark Let us mention that if $f^i \in \mathcal{B}^{\alpha_i}_{r,q}$, then $f^i \in \mathcal{B}^{\alpha}_{r,q}$ with $\alpha = \min_{1\leq i\leq p}\alpha_i$ so the estimator inherits the rate of convergence of the less regular function.

2.3 Noise covariance estimation

Now, let us turn back to the practical issue of the noise covariance estimation. In the one-dimensional case, a convenient estimation of the noise variance is needed in order to fine tune the threshold. Since typically (under very general assumptions) the finer detail coefficients are essentially a Gaussian noise possibly contaminated by a few large coefficients due to the signal, a robust estimator based on the median absolute deviation is used. Similarly for a natural extension to the multivariate case, a robust estimator of the covariance matrix must be defined.

This issue is of course crucial for factor analysis as PCA and highly robust covariance estimators with respect to outliers have been developed in this context. We use in this paper the Minimum Covariance Determinant (MCD) estimator proposed by Rousseeuw [15] and we apply it to D_1 the matrix of details at level 1. The MCD looks for that h-subset (typically $h = 0.75n$) of the data with the smallest determinant of its covariance matrix. More recently, Rousseeuw and Van Driessen [16] propose an algorithm [1] allowing quick computation of the MCD.

[1] We have used in this paper the Matlab code named *fastmcd* available from *ftp://win-ftp.uia.ac.be/pub/software/statis/newfiles/fastmcdm.gz* and we thank the authors for clear and efficient implementation.

2.4 Simulated examples

In this section, we present some selected and typical examples among numerous numerical trials. We apply the proposed procedure on simulated examples built from the well known signals introduced by Donoho and Johnstone [8] and widely considered as bench examples. Two methods are compared:

- p direct univariate denoising in the initial basis which is considered as the reference method;
- its multivariate alternative: p univariate denoising in the new basis defined by diagonalizing the robust estimate of the noise covariance matrix.

Let us introduce some preliminary material, useful for the simulation examples all along the paper. We consider a 4-dimensional model of the form (2) with $n = 1024$. We consider noisy versions of the well known functions called "Blocks", "Bumps", "HeaviSine", "Doppler" (see [8] for the definitions) suitably normalized in order to avoid huge scale effect. The actual model is then defined by f^1 = Blocks, f^2 = Bumps, f^3 = HeaviSine and $f^4 = 10\times$Doppler and the covariance matrix given by:

$$\Sigma_\varepsilon = \Sigma_1 = \begin{pmatrix} 1 & 0.8 & 0.6 & 0.7 \\ 0.8 & 1 & 0.5 & 0.6 \\ 0.6 & 0.5 & 1 & 0.7 \\ 0.7 & 0.6 & 0.7 & 1 \end{pmatrix} \tag{14}$$

The noise covariance matrix Σ_ε is estimated using the MCD estimator: $\hat{\Sigma}_\varepsilon = MCD(D_1)$. Let us mention that the estimations carried out for all simulated examples are accurate.

For each model, we simulate 50 realizations of $(X^1(t), ..., X^4(t))_{t=1,...,1024}$ and then obtain 50 realizations of each denoised signal $(\check{X}^i_k)_{k=1,...,50}$.
To evaluate the performance of each method, we introduce the following error criterion, which is an unbiased estimate of a normalized version of the individual risk (5), calculated for each $(\check{X}^i)$:

$$R(\check{X}^i, f^i) = \frac{1}{50}\sum_{k=1}^{50}\frac{1}{1024}\sum_{t=1}^{1024}(\check{X}^i_k(t) - f^i(t))^2 \tag{15}$$

The smaller R, the better the denoising. In addition, we compute the global error performance indicator:

$$R_{glob}(\check{X}) = \sum_{i=1}^{4} R(\check{X}^i, f^i)$$

We perform the wavelet decomposition at level $J = 5$ (a usual choice when $n = 2^{10}$) and at level $J = 3$ (this last value is provided by an automatic choice discussed in 3.2) using the near symmetric wavelet of order 4 (with four vanishing moments) due to Daubechies (see [5]).

The performance of the two methods (and for the two values of J) are given in Table 1. The relative performance is defined by the ratio of the risk of the reference method and the risk of our method.

Table 1
Multivariate denoising of noisy versions of "Blocks", "Bumps", "HeaviSine" and "10× Doppler" with $\Sigma_\varepsilon = \Sigma_1$: performance of our method compared to the reference one

level	method	$R(., f^1)$	$R(., f^2)$	$R(., f^3)$	$R(., f^4)$	R_{glob}
$J = 5$	reference	0.635	0.585	0.239	0.589	2.048
	our method	0.528	0.494	0.327	0.549	1.898
	relative performance	120%	118%	73%	107%	108%
$J = 3$	reference	0.521	0.407	0.360	0.512	1.800
	our method	0.485	0.399	0.378	0.493	1.755
	relative performance	107%	102%	95%	104%	103%

For $J = 5$, as it can be seen the global performance is improved after change of basis. The same occurs for each individual function except for the more regular one HeaviSine. The reason is that the first method slightly oversmoothes while the second slightly undersmoothes and captures in a better way the discontinuities.
When $J = 3$, the comments are qualitatively the same but the performance is better and the difference between the two methods is smaller.

Now let us consider the same model but with the following covariance structure allowing positive and negative covariance terms:

$$\Sigma_\varepsilon = \Sigma_2 = \begin{pmatrix} 1 & -0.8 & 0.6 & -0.7 \\ -0.8 & 1 & -0.5 & 0.6 \\ 0.6 & -0.5 & 1 & -0.7 \\ -0.7 & 0.6 & -0.7 & 1 \end{pmatrix} \tag{16}$$

The results are given in Table 2.

Table 2
Multivariate denoising of noisy versions of "Blocks", "Bumps", "HeaviSine" and "10× Doppler" with $\Sigma_\varepsilon = \Sigma_2$: performance of our method compared to the reference one

level	method	$R(., f^1)$	$R(., f^2)$	$R(., f^3)$	$R(., f^4)$	R_{glob}
	reference	0.633	0.587	0.238	0.587	2.045
$J = 5$	our method	0.557	0.520	0.329	0.518	1.924
	relative performance	113%	110%	72%	113%	106%
	reference	0.519	0.406	0.364	0.510	1.799
$J = 3$	our method	0.497	0.407	0.380	0.488	1.772
	relative performance	104%	100%	74%	104%	101%

The conclusions are similar, except that the differences between the methods become smaller.

3 Multivariate wavelet denoising using PCA

The multivariate procedure previously examined can be generalized by looking at the deterministic relationships between the p signals.

In this section, we first recall the multiscale PCA proposed by Bakshi ([3], [4]) in another context and we discuss it from the denoising perspective. Next, a second denoising procedure combining wavelets and PCA is proposed and illustrated by some simulated examples.

3.1 Multiscale principal component analysis

The idea comes from Bakshi which proposes a multiscale version of principal component analysis (called MSPCA) in order to fine tune the limits in statistical process control (SPC). The MSPCA scheme is as follows:

- step 1. Perform the wavelet transform at level J of each column of X;
- step 2. For $1 \leq j \leq J$, perform the PCA of the matrix D_j and select an appropriate number p_j of useful principal components or suppress the detail D_j;
- step 3. Similarly, perform the PCA of the matrix A_J and select p_{J+1} principal components;

- step 4. From the simplified detail and approximation matrices reconstruct a new matrix $\check{X}$ containing the main features of the original matrix X, by inverting the wavelet transform;
- step 5. Finally perform the PCA of the matrix $\check{X}$ and build a convenient statistic for SPC.

Let us remark that to reduce dimensionality, the final PCA is necessary since, in general, $rank(\check{X}) = rank(X)$. Indeed, a PCA is performed at each level leading to an optimal basis depending on the level and then no dimensionality reduction is reached in the initial common basis.
Nevertheless, steps 2 and 3 provide a compact representation of the p original signals of length n under the form of $n\left(\sum_{j=1}^{J} p_j 2^{-j} + p_{J+1}2^{-J}\right)$ wavelet coefficients.
This procedure is attractive since it allows, roughly speaking, to decorrelate in time each signal by wavelet transform and decorrelate in space the p signals by PCA.

Let us interpret the MSPCA scheme in terms of model (2) from the denoising perspective. The step 2 for $j = 1$ reduces to compute the matrix V diagonalizing D_1. If $p_j = 0$, step 2 performs a crude denoising killing all the coefficients, even those attributable to signals. In fact, when j is small, step 2 focuses on noise components and when j is large, it focuses on the dimensionality reduction of the deterministic function f. In this last case, suppressing the components associated to lower eigenvalues leads to capture the significant ones as well as to denoise signals. For intermediate scales, the MSPCA scheme is difficult to interpret.

3.2 General denoising procedure

Our procedure for extending denoising to multivariate signals, sketched in section 2, does not take into account the deterministic relations between the p signals. A natural way to handle it, is:

- to apply the previously considered thresholding strategy involving change of basis using V for the details;
- to perform a PCA and select the convenient number of components for the approximations.

More precisely, the general procedure for multivariate denoising is as follows:

- step 1. Perform the wavelet transform at level J of each column of X;
- step 2. Define $\hat{\Sigma}_\varepsilon$ the estimator of the noise covariance matrix as $\hat{\Sigma}_\varepsilon =$

$MCD(D_1)$ and then compute V such that $\hat{\Sigma}_\varepsilon = V\Lambda V^T$ where $\Lambda = diag(\lambda_i, 1 \leq i \leq p)$. Apply to each detail after change of basis (namely D_jV, $1 \leq j \leq J$), the p univariate thresholding strategies using the threshold $t_i = \sqrt{2\lambda_i log(n)}$ for the i-th column of D_jV;
- step 3. Perform the PCA of the matrix A_J and select the appropriate number p_{J+1} of useful principal components;
- step 4. Reconstruct a denoised matrix $\check{X}$, from the simplified detail and approximation matrices, by inverting the wavelet transform;

Let us remark that a useful alternative scheme could be to cancel step 3 (which is natural when compression purposes are also of interest) by selecting $p_{J+1} = p$ and to introduce step 5:

- step 5. Perform a final PCA of the matrix $\check{X}$ obtained at step 4 and select $\check{p}$ principal components.

The proposed procedure both generalizes univariate denoising and captures the attractive property of dimensionality reduction of the MSPCA scheme leading to an additional denoising effect.

Remark For steps 3 and 5, one can use the well known and widely used Kaiser criterion selecting the components corresponding to eigenvalues greater than the mean of all the eigenvalues (see also [9]). Nevertheless such choices lead to retain a too small number of components in the toy examples examined in the next section. The following rule of thumb is used: retain the components associated to eigenvalues greater than 5% of the sum of all the eigenvalues.

Remark Another crucial issue from a practical viewpoint is the choice of the decomposition level J, which can be considered as a low frequency cut-off. Of course model selection type rule can be used to choose it, but here the comparison between the MSPCA scheme and our procedure suggests the entirely different following rule: select J corresponding to the first jump of $trace(\Lambda^{(j)} - \Lambda^{(1)})(\Lambda^{(j)} - \Lambda^{(1)})^T$ where $\Lambda^{(j)}$ is the diagonal matrix obtained from the PCA of D_j. The underlying idea is the following. For high frequency detail D_1 the matrix $V\Lambda^{(1)}V^T$ is close to $\hat{\Sigma}_\varepsilon = MCD(D_1)$ since the coefficients due to functions f are generally rare and of little influence. When j increases, D_j captures more and more components due to f and $\Lambda^{(j)} - \Lambda^{(1)}$ becomes larger and larger.

3.3 Simulated examples

Let us consider an example which addresses multivariate denoising including dimensionality reduction tools. We start with two previously considered initial

functions: "Blocks" the more irregular one and "HeaviSine" the more regular, and we consider the sum and the difference of these two functions, leading to the actual model defined by f^1 = Blocks, f^2 = HeaviSine, $f^3 = f^1 + f^2$ and $f^4 = f^1 - f^2$ and the covariance matrix $\Sigma_\varepsilon = \Sigma_2$ defined in (16).
Three methods are compared:

- the first multivariate procedure which is considered here as the reference method: p univariate denoising in the new basis diagonalizing covariance matrix, corresponding to steps 1, 2 and 4 of the general procedure;
- two variants corresponding to steps 1 to 4 for the first one and steps 1, 2, 4 and 5 for the second one.

The decomposition level J, the number of principal components p_{J+1} or $\check{p}$ are selected according to the rules introduced in the previous section. They lead to select $J = 3$ and p_{J+1} or $\check{p} = 2$ for all but a few realizations.

The performance of the three methods are given in Table 3.

Table 3
Multivariate denoising of noisy versions of "Blocks", "HeaviSine" and their sum and difference with $\Sigma_\varepsilon = \Sigma_2$: performance of the three methods

	$R(.,f^1)$	$R(.,f^2)$	$R(.,f^3)$	$R(.,f^4)$	R_{glob}
steps 1,2,4	0.469	0.369	0.457	0.427	1.722
steps 1,2,3,4 $p_{J+1} = 2$	0.343	0.213	0.390	0.310	1.256
relative performance	136%	173%	117%	138%	138%
steps 1,2,4,5 $\check{p} = 2$	0.301	0.200	0.391	0.325	1.217
relative performance	156%	185%	117%	131%	141%

Let us focus on the two last methods including PCA. First, they outperform tremendously the first method. Second, they perform well, even on the hard function "HeaviSine" (see the first experiment in 2.4). Third, they exhibit close global performance and seem to be equally interesting. Fourth, the last one performing PCA after the inverse wavelet transformation improves the performance on "HeaviSine". This suggests that final PCA has an additional denoising effect with respect to the PCA of the approximation coefficients leading to slightly oversmooth irregular functions (like "Blocks") and to better denoise more regular ones (like "HeaviSine").

Let us mention that, on these examples, direct univariate denoising is slightly improved by a final PCA but not sufficiently to outperform the two last methods.

The results obtained from the same model with the covariance matrix $\Sigma_\varepsilon = \Sigma_1$ are qualitatively similar but the results are globally less impressive (the global relative performance is 114% instead of 141%).

To synthesize, let us turn back to the effect of the choice of the decomposition level J. We give an example of the obtained results using the last considered method and the univariate method considered as the reference method in 2.4, for $J = 3$ and $J = 5$, for one realization focusing on the functions of interest f^1 and f^2. The results are given in Figure 1. The figure is composed of four rows

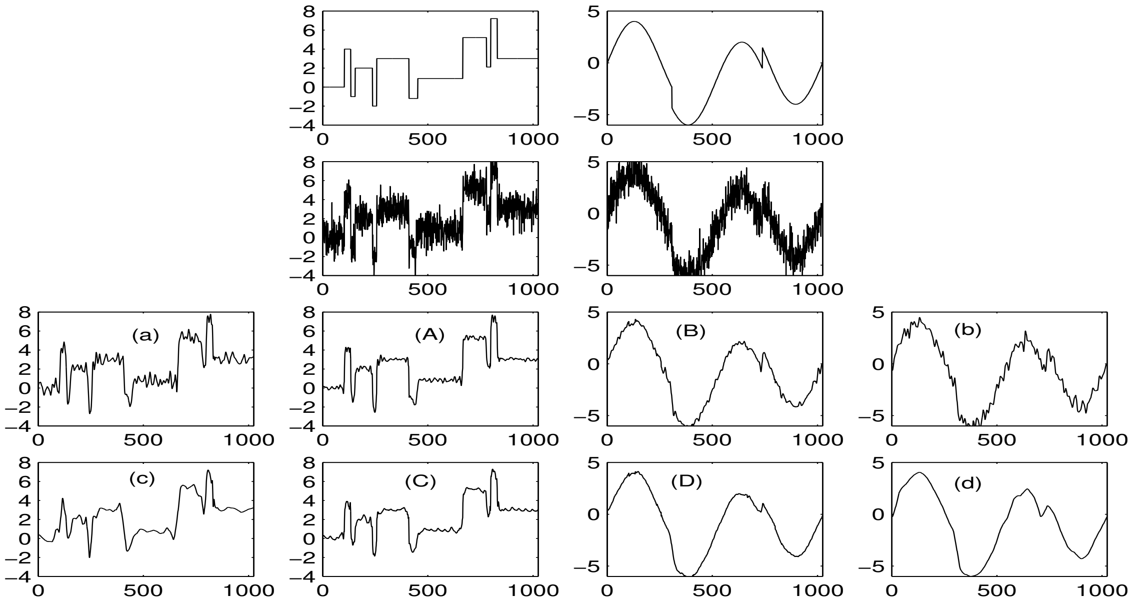

Fig. 1. Multivariate denoising of noisy versions of "Blocks", "HeaviSine" and their sum and difference with $\Sigma_\varepsilon = \Sigma_2$: denoising signals obtained from the two methods. First row: original signals, second row: noisy signals. The two last rows correspond to $J = 3$ and $J = 5$ respectively. Denoised signals using steps $1, 2, 4, 5$ with $\check{p} = 2$: plots A,B,C,D and univariate denoising: plots a,b,c,d.

of plots. The first one contain the original signals and the second the noised versions. The two last rows correspond to $J = 3$ and $J = 5$ respectively. The denoised signals using steps $1, 2, 4, 5$ with $\check{p} = 2$ can be found in the four plots labelled A,B,C and D. Those obtained using univariate denoising are drawed in the plots labelled a,b,c and d.

A visual comparison leads to the superiority of the procedure using $J = 3$: the two signals of interest are better recovered, especially around the discontinuities. The price to pay is to locally slightly undersmooth the signal where the underlying function is regular. In addition this figure illustrates the satisfactory behavior of our procedure.

To end, let us mention that a Matlab implementation of the various procedures considered in this paper is available from the authors upon request.

4 Concluding remarks

The strategy introduced in this paper seems to be efficient when it is examined from the denoising perspective. Of course it could be useful in some narrow contexts like functional data analysis in statistics or multivariate calibration in chemometrics, with some necessary adaptations.

The combination of wavelet strategies and data analytic tools is recent and is a field of cooperation between practitioners from various areas and the statistical community. Let us mention for example some references in chemometrics (see [18], [17]) or in data mining (see [20]). A lot of tools for multivariate signal processing is of interest in a statistical perspective both from theoretical and practical viewpoints.

References

[1] A. Antoniadis. Wavelet in statistics: a review. *J. Ital. Statist. Soc.*, 6:97–144, 1997.

[2] A. Antoniadis, J. Bigot, and T. Sapatinas. Wavelet estimators in nonparametric regression: a comparative simulation study. *Journal of Statistical Software*, 6:1–83, 2001.

[3] B. R. Bakshi. Multiscale PCA with application to MSPC monitoring. *AIChE Journal*, 44:1596–1610, 1998.

[4] B. R. Bakshi. Multiscale analysis and modeling using wavelets. *Journal of Chemometrics*, 13:415–434, 1999.

[5] I. Daubechies. *Ten lectures on wavelets.* SIAM, 1992.

[6] R. A. DeVore and G. G. Lorentz. *Constructive approximation.* Springer-Verlag, 1993.

[7] D. L. Donoho. De-noising by soft-thresholding. *IEEE Trans. Inform. Theory*, 41(3):613–627, 1995.

[8] D. L. Donoho and I. M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455, 1994.

[9] D. Karlis, G. Saporta, and A. Spinakis. A simple rule for the selection of principal components. *Comm. Statist. Theory Methods*, 32:643–666, 2003.

[10] G. Kerkyacharian and D. Picard. Thresholding algorithms, maxisets and well-concentrated bases. *Test*, 9(2):183–244, 2000.

[11] S. Mallat. *A wavelet tour of signal processing.* Academic Press Inc., 1998.

[12] M. Misiti, Y. Misiti, G. Oppenheim, and J-M. Poggi. *Les ondelettes et leurs applications.* Hermes, 2003.

[13] R. T. Ogden. *Essential wavelets for statistical applications and data analysis.* Birkhäuser, 1997.

[14] D. B. Percival and A. T. Walden. *Wavelet methods for time series analysis.* Cambridge University Press, 2000.

[15] P. J. Rousseeuw. Least median of squares regression. *J. Amer. Statist. Assoc.*, 79:871–880, 1984.

[16] P. J. Rousseeuw and V. K. Driessen. A fast algorithm for the minimum covariance determinant estimator. *Technometrics*, 41:212–223, 1999.

[17] P. Teppola and P. Minkkinen. Wavelet-PLS regression models for both exploratory data analysis and process monitoring. *Journal of chemometrics*, 14:383–399, 2000.

[18] P. Teppola and P. Minkkinen. Wavelet for scrutizing multivariate exploratory models-interpreting. *Journal of chemometrics*, 15:1–18, 2001.

[19] B. Vidakovic. *Statistical modeling by wavelets.* Wiley, 1999.

[20] M. Vlachos, J. Lin, E. Keogh, and D. Gunopulos. A wavelet-based anytime algorithm for k-means clustering of time series. *Workshop on Clustering High Dimensionality Data and Its Applications, 3rd SIAM International Conference on Data Mining*, 2003.