TrendTM: A R Package for the Trend of High-Dimensional Time Series Estimation

Emilie Lebarbier, Nicolas Marie, Amélie Rosier

TRENDTM: A R PACKAGE FOR THE TREND OF HIGH-DIMENSIONAL TIME SERIES ESTIMATION

EMILIE LEBARBIER[†], NICOLAS MARIE[†], AND AMÉLIE ROSIER[†,◇]

ABSTRACT. In this paper, we present the R package `TrendTM` dedicated to the trend estimation of high dimensional time series matrices. The main features of this package is the possibility to take into account different types of temporal structures on the data (none, smooth or periodic) and to select, according to the user's wishes, either the rank k of the matrix, the parameter τ linked to the temporal structure or both. Then, a two-stage heuristic is provided for the joint selection of the rank k and of the tuning parameter τ. Moreover, the `TrendTM` function is applied to time series clustering and principal component analysis on real datasets. The package is available on the CRAN.

1. INTRODUCTION

Since the 1970's, it is usual to model a one-dimensional time series by a process $(X_t)_{t\in\mathbb{Z}}$ satisfying

$$F_\theta(X_{t+q},\dots,X_{t-q},\eta_{t+p},\dots,\eta_{t-p}) = 0 \ ; \ t\in\mathbb{Z}, \tag{1}$$

where $p,q\in\mathbb{N}$, $\eta=(\eta_t)_{t\in\mathbb{Z}}$ is a second order stationary process, often a white noise, and $\mathcal{F}=(F_\theta)_{\theta\in\Theta}$ is a family of continuous maps from $\mathbb{R}^{2(p+q+1)}$ into $\mathbb{R}$ indexed in a set Θ. For instance, ARMA models, GARCH models, and all their extensions are defined this way. An advantage of Model (1) is that $\mathcal{F}$ can be chosen in order to take into account properties known on the dynamics of the modeled phenomenon regardless to the datas. However, except in simple cases, Model (1) is difficult to extend to the high dimensional framework. It is also difficult to bypass the stationary condition on η. For a good reference on the classic time series models, see Gouriéroux and Monfort [12]. For an introduction to GARCH models, the reader can refer to Brockwell and Davis [7], Chapter 7. Finally, for an introduction to time series analysis with R, see Cowpertwait and Metcalfe [10].
Independently, for almost two decades, in particular thanks to the Netflix challenge on movies recommendations, the matrix factorization for the completion and denoising of high dimensional matrices with i.i.d. entries has been deeply investigated on the theoretical side (see [8, 16, 17, 18, 20]). Recently, Alquier and Marie [1] (resp. Alquier, Marie and Rosier [2]) has extended this method in order to take into account time series trends properties as periodicity in the denoising (resp. completion) process of time series matrices. On the theoretical side, to take into account trends properties in the definition of the denoising and completion estimators allowed the authors to improve existing risk bounds.

The `TrendTM` package, for "Trend of High-Dimensional Time Series Matrix Estimation", implements the denoising method investigated on the theoretical side in Alquier and Marie [1]. The selection of both the rank k of the trends matrix and the parameter τ controlling the temporal structure (the period for seasonal time series) of the dataset is a model selection issue, that can be difficult to perform in practice when some unknown constants exist in the associated criterion and in the particular context of two parameters to be selected. This issue is well-managed by the `TrendTM` package by using the strategy of Devijver et al. [11].
Moreover, since the denoising method implemented in the `TrendTM` package is based on matrix factorization algorithms, natural applications to time series clustering and principal component analysis (PCA) are investigated in this paper. To illustrate the efficiency of the package in these application fields, we present an application to ECG profiles and children growth profiles clustering.

The paper is organized as follows. Section 2 recalls the estimation procedure proposed by Alquier and Marie [1]. Section 3 discusses the model selection issue in practice. More precisely, data-driven existing

heuristics are recalled and presented for our joint selection on the rank and the trend parameter whose performances are studied in Section 4 on simulated data. Section 5 gives some precisions and guidelines of the proposed method in the package `TrendTM` and shows an application on real data. In Section 6, we show that this method can be used for two classical statistical problems which are the time series clustering and PCA. As mentioned above, illustrations on real datasets are given.

2. The trend estimation procedure

In this section, we recall the model and the estimation procedure proposed by Alquier and Marie [1].

2.1. A high dimensional matrix based time series model.

Consider the model

$$\mathbf{X} = \mathbf{M} + \varepsilon, \tag{2}$$

where $\mathbf{X}$ is an observed $d \times n$ matrix which rows are time series, $\mathbf{M}$ is a deterministic matrix of low rank $k \in \mathbb{N}^*$ (i.e. $k \ll d \wedge n$), and ε is a random matrix which rows are i.i.d. samples of a centered Gaussian stochastic process with a covariance matrix $\mathbf{\Sigma}_\varepsilon$.

When we want to take into account for a possible trend property in the time series, we assume that $\mathbf{M} = \underline{\mathbf{M}}\mathbf{\Lambda}$, where $\underline{\mathbf{M}}$ is a $d \times \tau$ matrix of low rank k (thus with $\tau > k$) and $\mathbf{\Lambda}$ is a known $\tau \times n$ full-rank matrix. Two trend's property cases are considered:

- **periodic series:** if the trend of $\mathbf{X}$ is τ-periodic then $\mathbf{\Lambda} = (\mathbf{I}_\tau | \cdots | \mathbf{I}_\tau)$ where $\mathbf{I}_\tau$ is the identity matrix in $\mathcal{M}_{\tau,\tau}(\mathbb{R})$,
- **smooth series:** if the form of the trend is $t \in \{1, \dots, n\} \mapsto f(t/n)$ with $f \in \mathbb{L}^2([0,1]; \mathbb{R}^d)$, then

$$\mathbf{\Lambda} = \left(\varphi_\ell \left(\frac{t}{n} \right) \right)_{(\ell,t) \in \{1,\dots,\tau\} \times \{1,\dots,n\}},$$

  where τ is odd and $(\varphi_1, \dots, \varphi_\tau)$ is the τ-dimensional trigonometric basis defined by

$$\varphi_\ell(x) := \begin{cases} 1 & \text{if } n = 1 \\ \sqrt{2}\cos(2\pi m x) & \text{if } \ell = 2m \\ \sqrt{2}\sin(2\pi m x) & \text{if } \ell = 2m+1 \end{cases}$$

  for every $x \in [0,1]$ and $m \in \{1, \dots, (\tau-1)/2\}$.

We define an auxiliary model that will be used for the estimation of $\mathbf{M}$ according to model (2) and the definition of $\underline{\mathbf{M}}$ that is

$$\underline{\mathbf{X}} = \underline{\mathbf{M}} + \underline{\varepsilon}, \tag{3}$$

where $\underline{\mathbf{X}} := \mathbf{X}\mathbf{\Lambda}^+$, $\underline{\varepsilon} := \varepsilon\mathbf{\Lambda}^+$ and $\mathbf{\Lambda}^+ = \mathbf{\Lambda}^*(\mathbf{\Lambda}\mathbf{\Lambda}^*)^{-1}$ is the Moore-Penrose inverse of $\mathbf{\Lambda}$. This model doesn't embed some trend's property anymore. Note that the case with no trend corresponds to take $\tau = n$ and $\mathbf{\Lambda} = \mathbf{I}_n$.

2.2. Estimation procedure.

The estimation procedure consists in two steps: estimate $\mathbf{M}$ for fixed k and τ, then choose these parameters which is a model selection issue.

Step 1: Estimation of $\mathbf{M}$*,* k *and* τ *being fixed.* Consider the least-square estimator of the matrix $\underline{\mathbf{M}}$, that is

$$\widehat{\underline{\mathbf{M}}}_{k,\tau} \in \arg\min_{\mathbf{A} \in \mathcal{S}_{k,\tau}} \|\underline{\mathbf{X}} - \mathbf{A}\|_{\mathcal{F}}^2, \tag{4}$$

where $\|.\|_{\mathcal{F}}$ is the Frobenius norm (for a matrix $\mathbf{A}$, $\|\mathbf{A}\|_{\mathcal{F}} := \mathrm{trace}(\mathbf{A}\mathbf{A}^*)^{1/2}$) and $\mathcal{S}_{k,\tau} \subset \{\mathbf{U}\mathbf{V}\ ;\ \mathbf{U} \in \mathcal{M}_{d,k}(\mathbb{R})$ and $\mathbf{V} \in \mathcal{M}_{k,\tau}(\mathbb{R})\}$. So, a natural estimator of $\mathbf{M}$ is given by

$$\widehat{\mathbf{M}}_{k,\tau} := \widehat{\underline{\mathbf{M}}}_{k,\tau}\mathbf{\Lambda}.$$

Step 2: Choice of k and τ. Alquier and Marie [1] (see Section 4) provides a data-driven selection criterion. Precisely, for a fixed $s > 0$, the final estimator of $\mathbf{M}$ is $\widehat{\mathbf{M}}_s := \widehat{\mathbf{M}}_{\widehat{k}(s),\widehat{\tau}(s)}$ where

$$(\widehat{k}(s), \widehat{\tau}(s)) \in \arg \min_{(k,\tau)\in\mathcal{K}\times\mathcal{T}} \{\|\mathbf{X} - \widehat{\mathbf{M}}_{k,\tau}\|_{\mathcal{F}}^2 + \mathrm{pen}_s(k,\tau)\}$$

with $\mathcal{K} \subset \{1, \ldots, d \wedge n\}$, $\mathcal{T} \subset \{1, \ldots, n\}$, and for $(k, \tau) \in \mathcal{K} \times \mathcal{T}$,

$$\mathrm{pen}_s(k,\tau) := \mathfrak{c}_{\mathrm{pen}} \, \|\mathbf{\Sigma}_\varepsilon\|_{\mathrm{op}} \, k(d + \tau + s) \tag{5}$$

where $\mathfrak{c}_{\mathrm{pen}} > 0$ is a deterministic constant and the operator norm is given by $\|\mathbf{A}\|_{\mathrm{op}} = \sup_{\|x\|=1} \|\mathbf{A}x\|$ with $\|.\|$ the Euclidean norm on $\mathbb{R}^n$. Alquier and Marie [1] (Theorem 4.1) ensures that for every $\theta \in (0, 1)$, with probability larger than $1 - 2e^{-s}$,

$$\|\widehat{\mathbf{M}}_s - \mathbf{M}\|_{\mathcal{F}}^2 \leqslant \min_{(k,\tau)\in\mathcal{K}\times\mathcal{T}} \min_{\mathbf{A}\in\mathcal{S}_{k,\tau}} \left\{ \left(\frac{1+\theta}{1-\theta}\right)^2 \|\mathbf{A}\mathbf{\Lambda} - \mathbf{M}\|_{\mathcal{F}}^2 + \frac{4}{\theta(1-\theta)^2} \mathrm{pen}_s(k,\tau) \right\}.$$

3. Heuristic for the calibration of the penalty constants

The penalty function defined by (5) depends on some constants s and $\mathfrak{c}_{\mathrm{pen}}$ that must to be chosen or calibrated in practice. If the constant s can be easily chosen, the choice of the fixed $\mathfrak{c}_{\mathrm{pen}}$ is a difficult task. It is now usual to calibrate this constant in a data-driven manner. In the literature, different and now well known heuristics have been developed to this purpose. However, they are all dedicated to the selection of one dimension or parameter, that is our case if we aim at selecting k for a fixed τ or τ for a fixed k. In the case of the selection of two parameters, Devijver et al. [11] proposed a two-stage heuristic using for each step one of the previous heuristic. We first recall some heuristics for the selection of one parameter, then we present the two-stage heuristic for the joint selection of (k, τ).

First, in practice, we could take $s = -\log((1-\alpha)/2)$ with α fixed to 99%, 95% or 90%. Here we choose to fix $s = 4$. The penalty function is thus reduced to

$$\mathrm{pen}(k,\tau) := \mathfrak{c}_{\mathrm{cal}} \, k(d + \tau + s) \; ; \forall (k,\tau) \in \mathcal{K} \times \mathcal{T},$$

where $\mathfrak{c}_{\mathrm{cal}} > 0$ is a penalty constant that needs to be calibrated. The resulting adaptive estimator is denoted by $\widehat{\mathbf{M}}_s = \widehat{\mathbf{M}}_{\widehat{k},\widehat{\tau}}$.

Up to our knowledge, there exit the three following heuristics dedicated to the constant calibration question in the model selection frameworks of one parameter:

- the one proposed by Lavielle [19], denoted here ML, that involves a threshold S which is fixed to $S = 0.75$ as suggested by the author.
- the two proposed by Birgé and Massart [6] (see the more recent versions [3] and [4]) that are two versions of the well known slope heuristic: the 'dimension jump' and the 'slope', denoted here BJ and Slope respectively. The both heuristics have been implemented in the R package `capushe` described in [5].

For the joint selection of (k, τ), following the strategy proposed by Devijver et al. [11], we propose the following two-stage heuristic: first, we choose the best τ for each $k \in \mathcal{K}$ via the criterion

$$\widehat{\tau}(k) \in \arg\min_{\tau\in\mathcal{T}} \{\|\mathbf{X} - \widehat{\mathbf{M}}_{k,\tau}\|_{\mathcal{F}}^2 + \mathfrak{c}_{\mathrm{cal},\tau} \, k(d + \tau + s)\},$$

where the penalty constant $\mathfrak{c}_{\mathrm{cal},\tau}$ is calibrated using one of the previous heuristic, then we select the best k among them via the criterion

$$\widehat{k} \in \arg\min_{k\in\mathcal{K}} \{\|\mathbf{X} - \widehat{\mathbf{M}}_{k,\widehat{\tau}(k)}\|_{\mathcal{F}}^2 + \mathfrak{c}_{\mathrm{cal},k} \, k(d + \widehat{\tau}(k) + s)\},$$

where the penalty constant $\mathfrak{c}_{\mathrm{cal},k}$ is calibrated using the same heuristic to be consistant, and $\widehat{\tau} = \widehat{\tau}(\widehat{k})$.

Note that, in practice $\mathcal{K} = \{1, \dots, k_{\max}\}$ and $\mathcal{T} = \{k+1, \dots, \tau_{\max}\}$ where $k_{\max}$ is the maximal rank and $\tau_{\max}$ is the maximal value of τ where these two quantites need to be specified.

4. Simulation study

In this section, we first study the behavior of the three heuristics for the model selection issue for k and τ separatly (Study 1), then we illustrate the importance of accounting for the trend in the estimation procedure when it exists in the series (Study 2) and finally we study the perfomance of the proposed method when both k and τ are selected (Study 3) on simulated data. In this framework, we only consider the smooth trend case.

4.1. Simulation design and quality criteria.

4.1.1. *Simulation design.* We simulated datasets with $d = 100$ and $n = 600$ as follows:

(1) we generate a matrix $\underline{\mathbf{M}} = \mathbf{UV}$ by simulating $\mathbf{U} \in \mathcal{M}_{d,k}(\mathbb{R})$ and $\mathbf{V} \in \mathcal{M}_{k,\tau}(\mathbb{R})$ for which the entries of $\mathbf{U}$ and $\mathbf{V}$ are assumed to be i.i.d. and follows a centered Gaussian distribution with same standard deviation σ_{uv} fixed to 0.5;
(2) two cases are considered according to the presence or not of a trend in the simulated series: if there is no trend, then $\tau = n$ and $\mathbf{M} = \underline{\mathbf{M}}$, and otherwise $\mathbf{M} = \underline{\mathbf{M}}\mathbf{\Lambda}$ with the matrix $\mathbf{\Lambda}$ of the smooth case. To distinguish between these two cases in the sequel, we call them $\text{dataset}_{\text{NoTrend}}$ and $\text{dataset}_{\text{Trend}}$ respectively;
(3) the rows of the error matrix ε are assumed to be i.i.d. and to follow a centered Gaussian distribution of variance σ^2 (i.e. $\mathbf{\Sigma}_\varepsilon = \sigma^2 \mathbf{I}_n$).

We fix $k = 3$ and $\tau = 25$. We consider different values for the residual standard deviation σ in order to have different levels of difficulty for the estimation problem. First, according to the previous considerations, $\text{Var}(\mathbf{M}_{ij}) = k\sigma_{uv}^4$ for $\text{dataset}_{\text{NoTrend}}$ and $\text{Var}(\mathbf{M}_{ij}) = \tau k \sigma_{uv}^4$ for $\text{dataset}_{\text{Trend}}$. Let us consider $s \in \{0.1, 0.5, 1.5, 2\}$. In order to have the same estimation difficulty (same ratio between σ and the standard deviation of M_{ij}) for the two datasets, we set $\sigma = s$ for $\text{dataset}_{\text{NoTrend}}$ and $\sigma = \sqrt{\tau}\ s$ for $\text{dataset}_{\text{Trend}}$. The obtained four cases are judged as 'Easy', 'Medium', 'Difficult' and 'Hard' respectively. For each combination of parameters, we've simulated 200 datasets.

Let us precise that when the trend is not considered in the estimation procedure, the resulting estimator is $\widehat{\mathbf{M}}_{k \text{ or } \widehat{k},n}$ (if k is selected or not) and when it is considered the resulting estimator is $\widehat{\mathbf{M}}_{k \text{ or } \widehat{k},\tau \text{ or } \widehat{\tau}}$ (if both k and τ are selected or one of them or none).

4.1.2. *Quality criteria.* The performance of our procedure is assessed via:

- the estimated k and/or τ; and
- the squared Frobenius norm between $\mathbf{M}$ and its estimate $\widehat{\mathbf{M}}_{\widehat{k},\widehat{\tau}}$.

Moreover we also consider the Frobenius norm between $\mathbf{M}$ and

- the estimator of $\mathbf{M}$ for the true k and/or τ, that is $\widehat{\mathbf{M}}_{k,\tau}$; and
- the trajectorial oracle that is $\widehat{\mathbf{M}}_{\tilde{k},\tilde{\tau}}$ where $(\tilde{k}, \tilde{\tau}) = \arg\min_{(k,\tau)\in\mathcal{K}\times\mathcal{T}} \|\mathbf{M} - \widehat{\mathbf{M}}_{k,\tau}\|_{\mathcal{F}}^2$ when both k and τ are selected, $\widehat{\mathbf{M}}_{k,\tilde{\tau}}$ where $\tilde{\tau} = \arg\min_{\tau\in\mathcal{T}} \|\mathbf{M} - \widehat{\mathbf{M}}_{k,\tau}\|_{\mathcal{F}}^2$ when k is fixed, and $\widehat{\mathbf{M}}_{\tilde{k},n}$ where $\tilde{k} = \arg\min_{k\in\mathcal{K}} \|\mathbf{M} - \widehat{\mathbf{M}}_{k,n}\|_{\mathcal{F}}^2$ when no trend is considered.

4.2. **Study 1: behavior of the three heuristics for the selection of k or τ.** We first study the selection of k for $\text{dataset}_{\text{NoTrend}}$ when no trend is considered in the estimation procedure. We consider two different values of the maximal rank $k_{\max} \in \{15, 35\}$. The results are presented in Figure 1. When the noise is small, i.e. the estimation problem is easy (cases 'Easy' and 'Medium'), all the heuristics recover the true rank, and therefore the obtained estimators perform as well as $\widehat{\mathbf{M}}_{k,n}$ (the estimator of $\mathbf{M}$ for k fixed to its true value). When the estimation problem gets more difficult (cases 'Difficult' and 'Hard'), the heuristics have a tendency to underestimate the rank. This underestimation behavior seems to be logical and even desirable in the particular 'Hard' case. Indeed, we observe that in terms of Frobenius norm, the obtained estimators perform better compared to the one with the true rank. Moreover, they

have performance close to the oracle. Comparing the three heuristics, the Slope heuristic shows better performance compared to the two other heuristics. This is particularly marked for the 'Medium' case and $k_{\max} = 15$. We can note that the behavior of the three heuristics can be affected by the choice of $k_{\max}$. This problem is well known for both the BJ and Slope heuristics (see Arlot [4] for more explanations in the case univariate series analysis).

Then, we study the selection of τ for $\text{dataset}_{\text{Trend}}$ for k fixed to the true value. We fix $\tau_{\max} = 55$. The results are presented in Figure 2. Except with BJ that is more unstable, the heuristics retrieve the true value of τ whatever the estimation difficulty with same performance as the oracle.

From this study, we choose the Slope heuristic for the model selection issue for both k and τ in the sequel and in the developed package.

4.3. **Study 2: accounting for the smooth structure in the trend.** We compare the performance of the procedure on the $\text{dataset}_{\text{Trend}}$ when the trend is considered ($\tau = \widehat{\tau}$) or not ($\tau = n$) for k fixed to the true value. We choose $\tau_{\max} = 55$. The results are represented in Figure 3. Whatever the difficulty of the estimation problem (different values of σ), accounting for the trend increases the precision of the estimation. This is more marked for high values of σ. Note that, similarly as Study 1, the estimation naturally degrades with the increasing of σ.

4.4. **Study 3: selection of k and τ.** Figure 4 shows that the joint heuristic retrieves the true values of k and τ whatever the difficulty of the estimation problem, except very few times. Thus, the perfomance of the estimator of $\mathbf{M}$ is comparable to the one of the estimator $\mathbf{M}_{k,\tau}$ and moreover it has performance close to the oracle (see Figure 5). Compared to Study 1 where $\tau = n$, here for difficult estimation problems, k is not underestimated.

5. Using package TrendTM

5.1. **Comments on the package.** The package is organized around the main and unique function `TrendTM`. In this section, we present the arguments used in a call to this function:

```
TrendTM(X, k.select=FALSE, k.max=20, struct.temp="none", tau.select=FALSE,
tau.max=floor(n/2), type.soft="als")
```

This function returns a list containing six elements:

- `k.est`, the estimated k or the true k when no selection is chosen;
- `tau.est`, the estimated τ or the true τ when no selection is chosen;
- `U.est`, the component U of the decomposition of the final estimator of $\mathbf{M}$;
- `V.est`, the component V of the decomposition of the final estimator of $\mathbf{M}$;
- `M.est`, the estimator of $\mathbf{M}$;
- `contrast`, the squared Frobenius norm of `X-M.est`. If k and τ are fixed, the contrast is an unique value; if k is selected and τ is fixed or if τ is selected and k is fixed, the contrast is a vector containing the norms for each visiting values of k or τ respectively; and if k and τ are selected, the contrast is a matrix with $k_{\max}$ rows and $\tau_{\max}$ colums such that $\text{contrast}_{k,\tau} = \|\mathbf{X} - \widehat{\mathbf{M}}_{k,\tau}\|_{\mathcal{F}}^2$.

5.1.1. *Data structure.* The structure of the data is a matrix with d rows and n columns containing the d time series with size n each. A simulated dataset without temporal structure is available in the package with $d = 30$ and $n = 100$. The user can load the data using the following command:

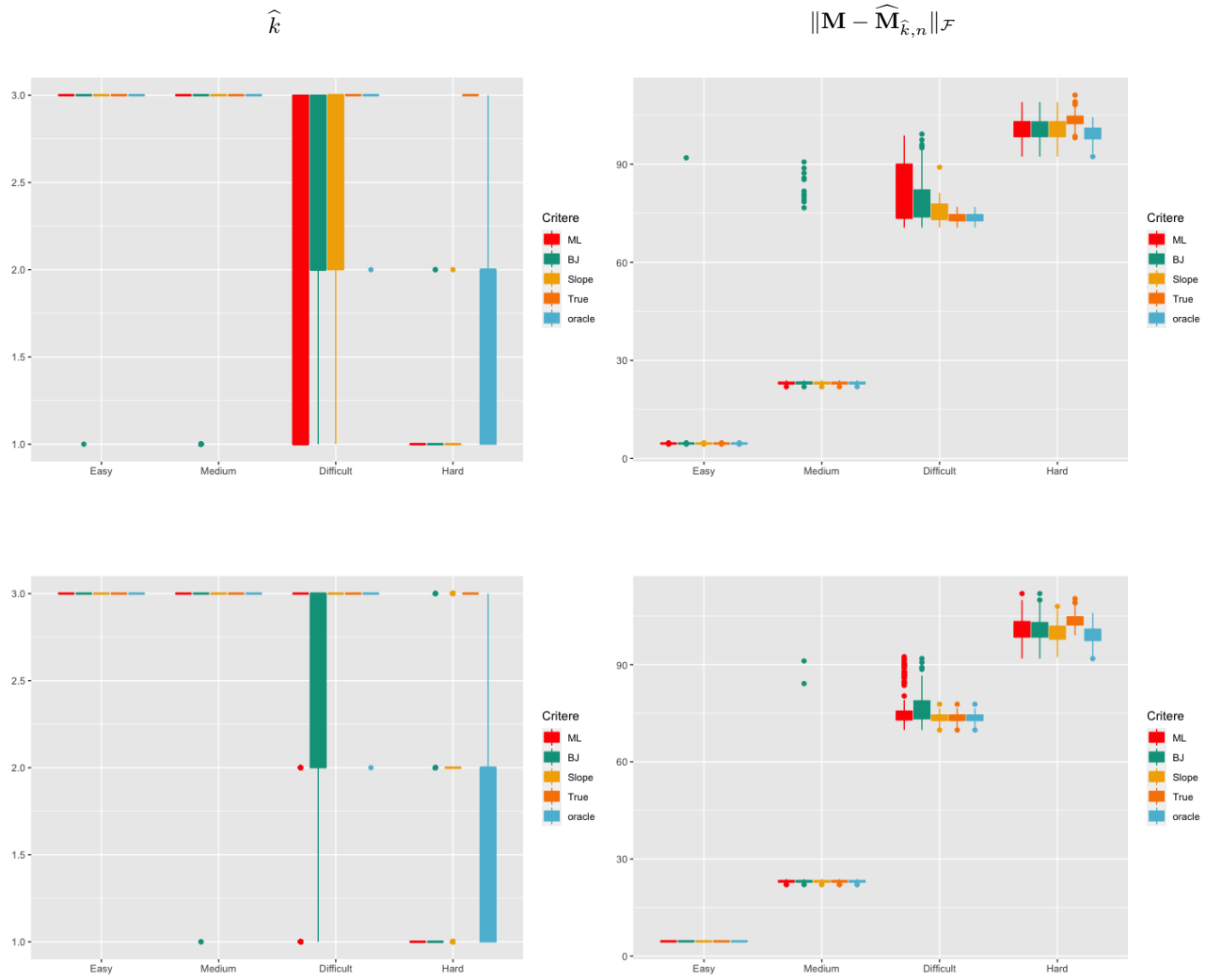


FIGURE 1. Comparison of the three heuristics for the selection of k for dataset$_{\text{NoTrend}}$. Left: estimated number of the rank k and right: boxplot of $\|\mathbf{M} - \widehat{\mathbf{M}}_{\widehat{k},n}\|_{\mathcal{F}}$ for two values of $k_{\max} = 15$ (first line) and $k_{\max} = 35$ (second line), and different values of σ. On each graph and for each value of σ, from left to right, we have the result from ML, BJ, Slope ($\widehat{k}$), the true rank (k) and the oracle ($\tilde{k}$).

```
data(DataX)
X[1:5,1:6]
```

```
           [,1]        [,2]       [,3]        [,4]         [,5]       [,6]
[1,]  0.1787622 -0.14222205 -0.4328011  0.02455812 -0.002211779 -0.1951183
[2,] -0.4288746 -0.58887443 -0.2381704 -0.35792283  0.724254462 -0.5426845
[3,]  0.2826668  0.07020323 -0.4627611 -0.24472557  0.035034332 -0.1256257
[4,]  0.4401198  0.36296884  0.1717901  0.72074220 -1.032545650  0.4435005
[5,]  0.7556780  0.32046226 -0.0846485 -0.01379206 -0.093781836  0.1607618
```

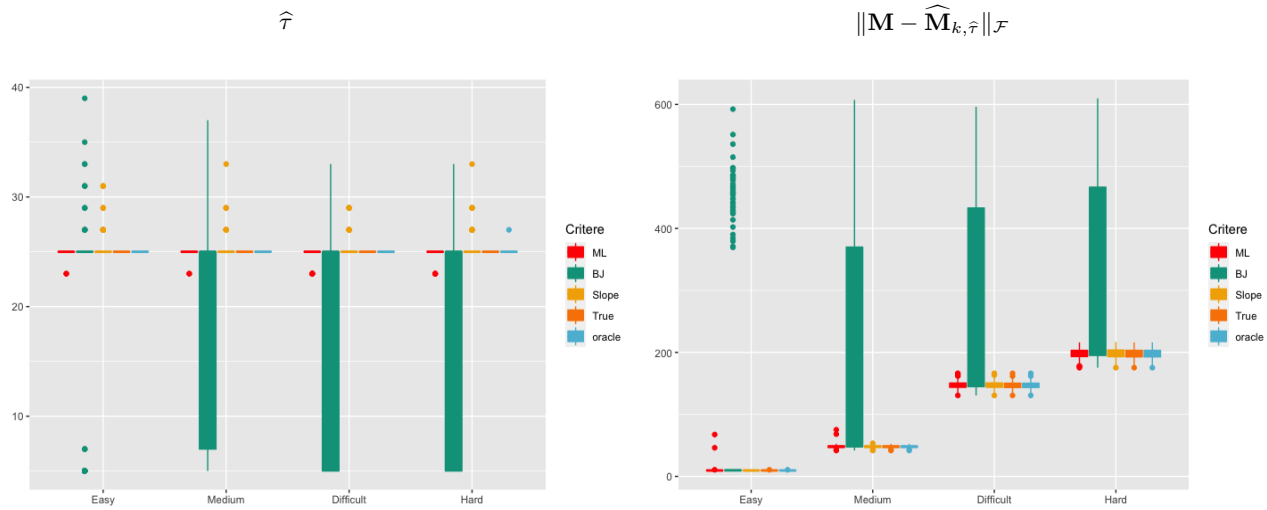
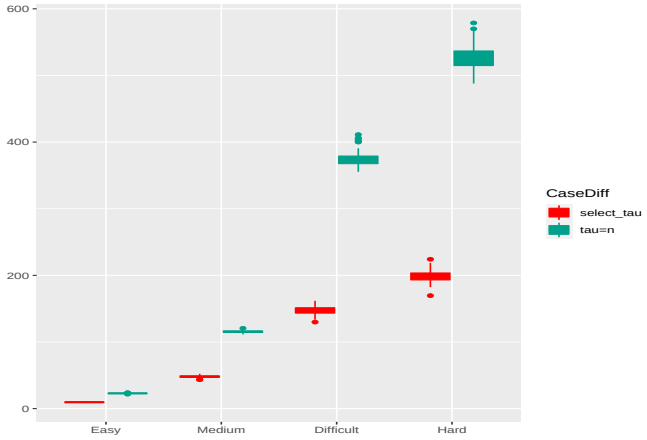


FIGURE 2. Comparison of the three heuristics for the selection of τ for dataset$_{\text{Trend}}$ when k is fixed to the truth ($k = 3$) for different values of σ. Left: estimated τ and right: boxplot of $\|\mathbf{M} - \widehat{\mathbf{M}}_{k,\widehat{\tau}}\|_{\mathcal{F}}$. In each graph and each value of σ, from left to right, we have the result from ML, BJ, Slope ($\widehat{\tau}$), the true value (τ) and the oracle ($\tilde{\tau}$).

5.1.2. *Model selection.* The selection of k or/and τ is requested using the options `k.select` or/and `tau.select` that are booleans. When there is no selection, the option is set to `FALSE` and `k.max`= k or/and `tau.max`= τ. Note that if no trend is considered in the estimation procedure, $\tau = n$, otherwise `tau.max` must be a smaller than n and larger than `k.max`+2 in order to ensure that the rank of $\mathbf{M}$ is k.

5.1.3. *Accounting for a trend.* Let us give more details about the different arguments of `TrendTM` that need to be specified when accounting for a temporal structure in the estimation procedure.



;

FIGURE 3. Boxplot of $\|\mathbf{M} - \widehat{\mathbf{M}}_{k,\tau}\|_{\mathcal{F}}$ with $\tau = \widehat{\tau}$ (select_tau) and $\tau = n$ (tau=n) for different values of σ.

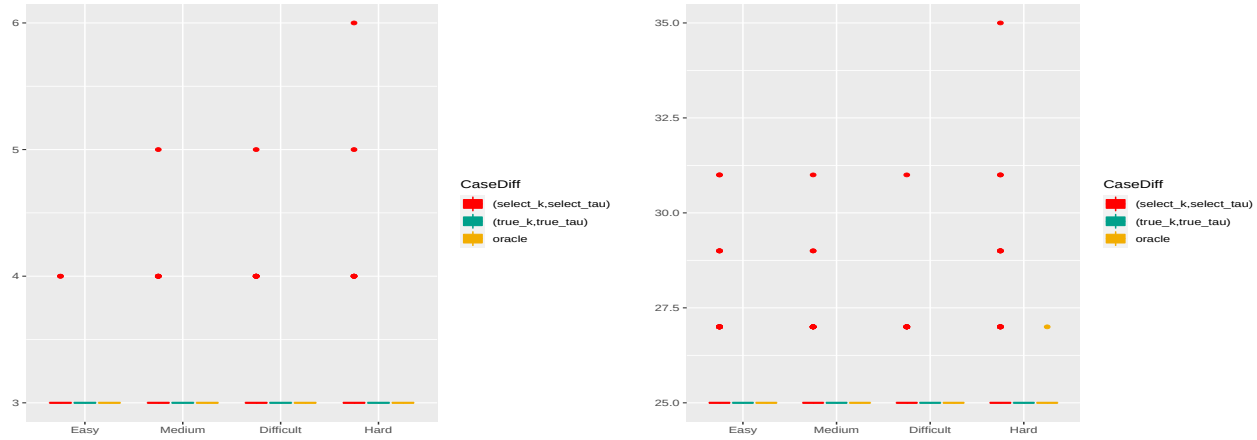
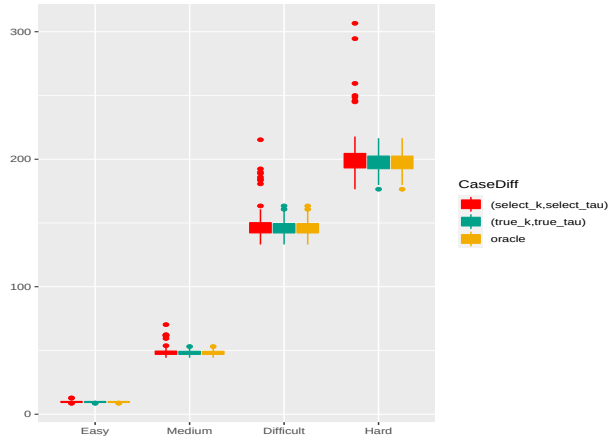


FIGURE 4. Left: estimated k. Right: estimated τ for different values of σ.



;

FIGURE 5. Boxplot of $\|\mathbf{M} - \widehat{\mathbf{M}}_{k,\tau}\|_{\mathcal{F}}$ with $(k,\tau) = (\widehat{k},\widehat{\tau})$ the selected k and τ, $(k,\tau) = (k,\tau)$ the true values and $(k,\tau) = (\tilde{k},\tilde{\tau})$ the oracle for different values of σ.

Two temporal structures are considered: periodic trend and smooth trend. This can be specified using the option `struct.temp`, `struct.temp="periodic"` or `struct.temp="smooth"` respectively. Recall that the selection of τ is only possible when a smooth trend is considered. Thus, when

- `struct.temp="periodic"`, then `tau.select=FALSE` and `tau.max`$= \tau$. In this case, τ must be such that n is a multiple of τ;
- `struct.temp="smooth"`, then `tau.select` is either `FALSE` or `TRUE`. Whatever this choice, `tau.max` must be an odd number.

When no trend is taken into account, `struct.temp="none"` and `tau.max`$= n$.

The function `TrendTM` returns `M.est=U.est%*%V.est%*%Lambda`, where the matrix $\mathbf{\Lambda}$ depends on the considered temporal structure. If needed, the code to compute this matrix and its Moore-Penrose inverse $\mathbf{\Lambda}^+$ (denoted below `LambdaP`) for both temporal structures and `tau.max`$= \tau$ is

```
if (struct.temp=="periodic"){
```

```
   p <- n/tau.max
   A <- matrix(1, ncol=p,nrow=1)
   B <- diag(tau)
   Lambda <- kronecker(A,B)
   LambdaP <- kronecker(t(A),B)*(1/p)
 }
if (struct.temp=="smooth"){
   times.eval <-seq(1/n,1,by=1/n)
   fbasis_obj= fda::create.fourier.basis(rangeval=c(0,1),nbasis=tau,period = 1)
   fbasis_evals <- fda::eval.basis(times.eval, fbasis_obj)
   Lambda <- t(fbasis_evals)
   LambdaP <- t(Lambda)/n
 }
```

5.1.4. *Estimation of* $\mathbf{M}$*,* k *and* τ *being fixed (Step 1).* The least-square estimator of $\mathbf{M}$, $\widehat{\mathbf{M}}_{k,\tau}$ given by (4), is obtained by using the `softImpute` function from the R package of the same name developed by Hastie and Mazumder for matrix completion [13]. In this package, two algorithms are implemented: 'svd' and 'als'. In a simulation study, we observed that they have both provided the same accuracy of the estimator (results not shown). We decide to use the 'als' algorithm by default but the choice is left free to the user in our package `TrendTM`. In this package, this choice is specified using the option `type.soft`.

5.1.5. *The Slope heuristic.* Let us now focus on the selection problem of the rank k, and write the penalty as $\text{pen}(k) = \mathfrak{c}_{\text{cal,k}}\ \varphi(k)$. The Slope heuristic, proposed by [6], consists in estimating the slope $\widehat{s}$ of the contrast $\|\mathbf{X} - \widehat{\mathbf{M}}_{k,n}\|_{\mathcal{F}}^2$ as a function of $\varphi(k)$ with k 'large enough' and defining $\mathfrak{c}_{\text{cal,k}} = -2\widehat{s}$. The implementation of this heuristic requires the choice of the dimensions on which to perform the regression, that can be difficult in practice. To deal with this problem, Baudry et. al [5] proposed to make robust regressions for dimensions between k and $k_{\max}$ for $k = 1, 2, \ldots$, resulting in different selected $\widehat{k}$. The choice of the final dimension is the maximal value $\widehat{k}$ such that the length of successive same $\widehat{k}$ is greater than the option point of the function DDSE. In order to avoid some implementation problems as such condition is not reached and no k is selected, we decide to take the value $\widehat{k}$ associated to the maximal length of successive same $\widehat{k}$.

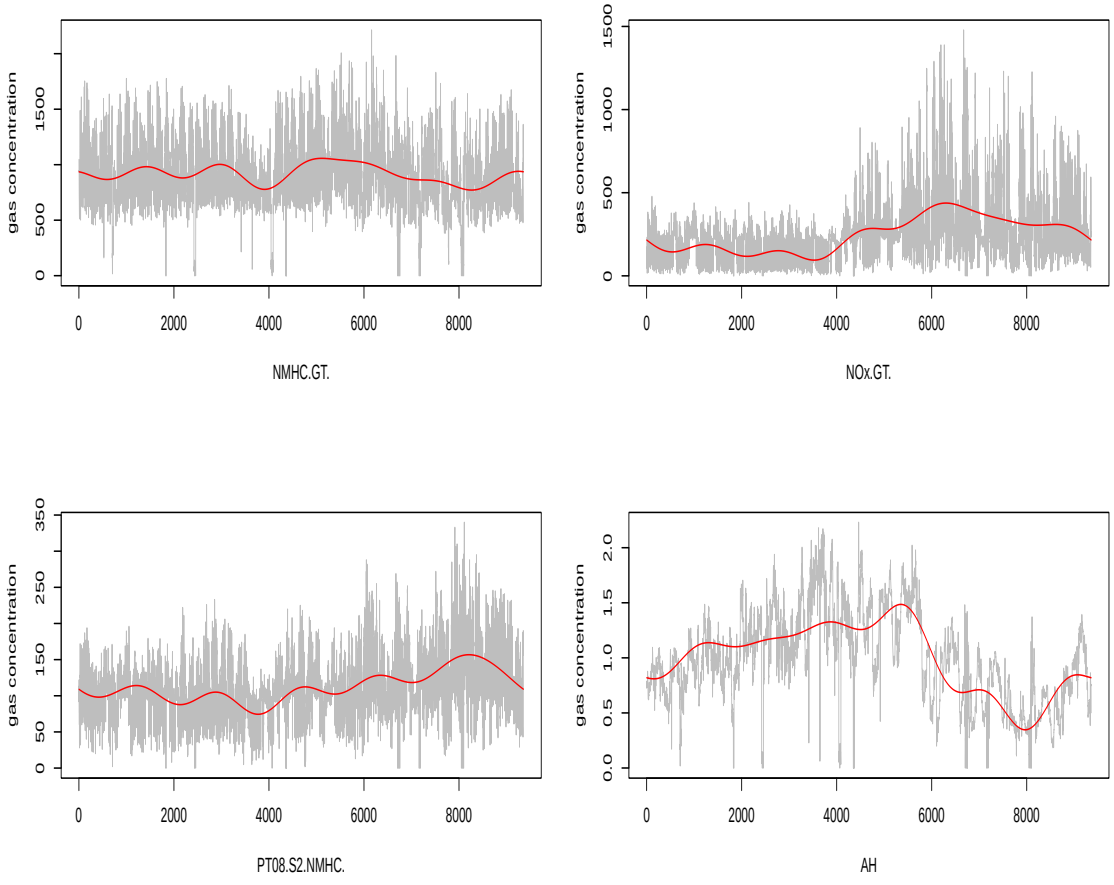
5.2. **Application to pollution dataset.** Let us use the package on a real dataset [1]. The dataset contains the amounts of $d = 13$ toxic gases in the air recorded $n = 9357$ times during one year. We do a first step of data imputation using the function `complete` of the package `softImpute` since missing values (coded with -200) exist in this dataset (see [2] for more details on high dimensional time series completion). Our function `trendTM` is then applied on the completed matrix with a selection of both k and τ. The code is the following:

```
## Used package
library('softImpute')
library('TrendTM')

## Data importation
AirPollution <- read.table('AirQualityUCI.csv',sep=";",header=TRUE)[,-c(1:2)]
AirPollution <- t(as.matrix(AirPollution))

## Imputation
AirPollution[AirPollution==-200] <- NA
AirPollutionSoft=softImpute::softImpute(AirPollution,rank=1,lambda=0)
AirPollutionImp=softImpute::complete(AirPollution,AirPollutionSoft)
```

[1] available at https://archive.ics.uci.edu/ml/datasets/Air+quality

```
## Trend estimation
Trend.AirPollution = TrendTM(AirPollutionImp, k.select=TRUE, k.max=13,
struct.temp="smooth", tau.select=TRUE, tau.max=101)

Trend.AirPollution$k.est
[1] 7
Trend.AirPollution$tau.est
[1] 13
```

The procedure selects $\widehat{k} = 7$ and $\widehat{\tau} = 13$. Figure 6 shows the obtained trend estimation for 4 toxic gases among the 13. The denoising process seems to have been well applied to the data.



;

FIGURE 6. Estimation of the trend of 4 toxic gases (red) compared to the initial series (grey).

6. METHOD USED FOR USUAL PROBLEMS IN STATISTICS

This section deals with an application of Model (2), first to times series clustering, and then to PCA of times series. These applications were not developed in Alquier and Marie [1].

6.1. **Application to time series clustering.** To assume that $\mathbf{M} = \mathbf{UL}$ with $\mathbf{L} = (\ell_j(t))_{j,t} := \mathbf{V\Lambda}$ means that for any $i \in \{1, \ldots, d\}$, the trend $m_i(.)$ of the i-th row of $\mathbf{X}$ satisfies

$$m_i(t) = \sum_{j=1}^{k} \mathbf{U}_{i,j}\ell_j(t) \ ; \forall t \in \{1, \ldots, T\}.$$

In other words, the trend of the i-th time series stored in $\mathbf{X}$ is a linear combination of the trends $\ell_1(.), \ldots, \ell_k(.)$ of k latent time series. Moreover, since $\mathbf{L} = \mathbf{V\Lambda}$, $\ell_1(.), \ldots, \ell_k(.)$ have the same usual time series trend's property, characterized by $\mathbf{\Lambda}$, than $m_1(.), \ldots, m_d(.)$.
The identification of the aforementioned latent time series is part of curves or time series clustering framework, where k corresponds to the number of clusters. To this aim, we propose here to apply the well-known Hierarchical Agglomerative Clustering with the Ward's linkage on the rows of $\widehat{\mathbf{U}}_k$, where $\widehat{\underline{\mathbf{M}}}_{k,\tau} = \widehat{\mathbf{U}}_k\widehat{\mathbf{V}}_\tau$ or $\widehat{\mathbf{M}}_{k,\tau} = \widehat{\mathbf{U}}_k\widehat{\mathbf{V}}_\tau\mathbf{\Lambda}$.

In [14], Jacques and Preda applied and compared different curves clustering methods proposed in the literature on real datasets. We propose to apply our clustering strategy on two datasets: the ECG and Growth datasets. The ECG dataset (taken from the UCR Time Series Classification and Clustering website) consists in 200 electrocardiogram from 2 groups of patients sampled at 96 time instants in which 133 are classified as normal and 67 as abnormal, and the Growth dataset (available in the R package fda) contains the heights of 54 girls and 39 boys measured at 31 stages from 1 to 18 years. In both datasets, we have two groups. The times series of the two datasets are plotted in Figure 7.

We apply our proposed clustering strategy based on the procedure without and with taking into account for a trend (periodic for ECG and smooth for Growth) with the known number of groups $k = 2$. The Correct Classification Rates (CCR) according to the known partitions are given in Table 1 for the ECG dataset and in Table 2 for the Growth dataset. We also report the CCR obtained for the best method among the ones tested in [14]. In addition, we indicate the time taken by the different methods on a laptop 1.6 GHz CPU (note that the time for the method HDDC on FPCA scores is not given since this method is not available). For the ECG dataset, accounting for the trend improves significantly the clustering performance. This is not the case with the Growth dataset which already has a very high CCR without trend. Our clustering performances are the same compared to the best clustering methods but it is much more faster. Note that among the compared methods in [14], the best ones for the two datasets are not the same.

The R code of our clustering strategy for the Growth dataset without accounting the trend is the following

```
library(fda)
data("growth")
Growth <-t(cbind(matrix(growth$hgtm,31,39),matrix(growth$hgtf,31,54)))
cls <-c(rep(1,39),rep(0,54))

k.max <- 2
res.Growth <- TrendTM::TrendTM(Growth,k.max=k.max)

Uf.cr <- scale(res.Growth$U.est,scale=TRUE, center=TRUE)
Uf.ward<- hclust(dist(Uf.cr, method = "euclidean")^2, method = "ward.D")
cluster.Uf <- cutree(Uf.ward, k =k.max)
table(cls,cluster.Uf)
```

Note that for the Growth dataset without trend, the Slope heuristic selects $\widehat{k} = 6$ clusters and for the ECG dataset with a 32-periodic trend, it selects $\widehat{k} = 7$. In Figure 8, the series of the ECG dataset are

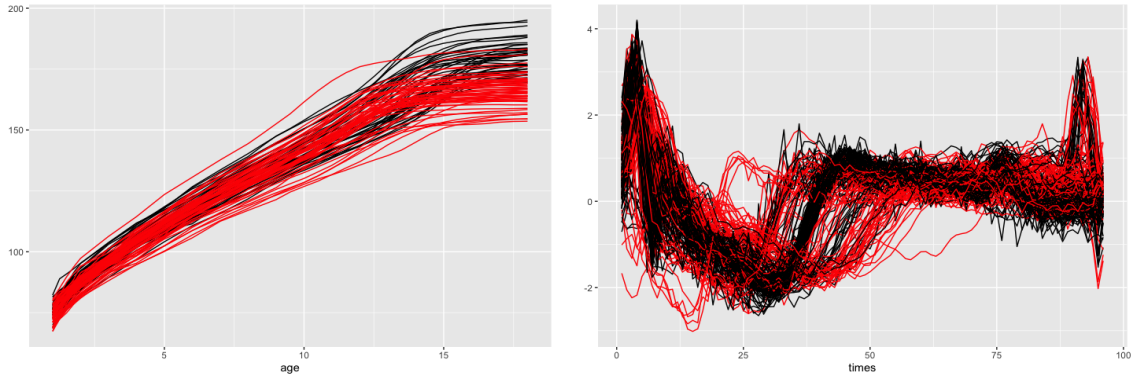


FIGURE 7. Left: Growth dataset (black: boys, red: girls).
Right: ECG dataset (black: normal, red: abnormal).

	$\text{proc}_{\text{NoTrend}}$	$\text{proc}_{\text{Trend}}$ $\tau = 32$-periodic	Best method in [14] Funclust
CCR	74.5	83	84 (reported from [14])
mean times in second (on 30 runs)	0.015	0.008	19.2

TABLE 1. Correct classification rates (CCR) in percentage accounting or not for a trend on the ECG dataset. Mean times in second obtained on 30 runs.

	$\text{proc}_{\text{NoTrend}}$	$\text{proc}_{\text{Trend}}$ smooth with $\widehat{\tau} = 13$	Best method in [14] HDDC on FPCA scores
CCR	97.85	97.85	97.85 (reported from [14])
mean times in second (on 30 runs)	0.003	0.0028	.

TABLE 2. Correct classification rates (CCR) in percentage accounting or not for a trend on the Growth dataset. Mean times in second obtained on 30 runs.

plotted and colored according to the $\widehat{k} = 7$ obtained clusters with a 32−periodic trend on the left, and the associated obtained trend $m_i(.)$ for $i = 1, \ldots, 6$ are plotted on the right.

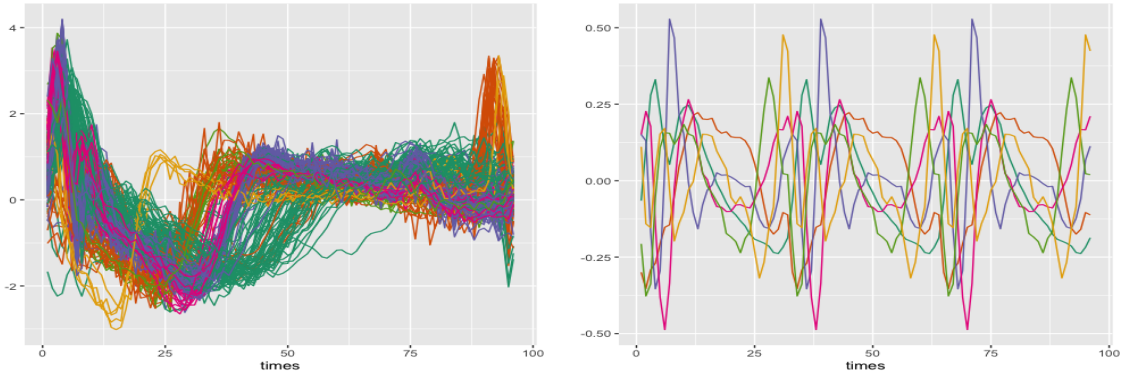


FIGURE 8. Left: ECG dataset colored according to the $\widehat{k} = 7$ obtained clusters with a 32−periodic trend. Right: the associated trends $m_i(.)$ for $i = 1, \ldots, 6$.

6.2. **Application to time series PCA.** The PCA problem can be rephrased as a k-rank approximation of a matrix $\mathbf{X}$ (information captured by the first k axes). More precisely, if the matrix $\mathbf{X}$ is of dimension $d \times n$, the simple PCA solution of rank k is given by

$$\widehat{\mathbf{M}}_{k,n} = \widehat{\mathbf{U}}_k \widehat{\mathbf{V}}_k \in \arg \min_{\mathbf{U} \in \mathcal{M}_{d,k}, \mathbf{V} \in \mathcal{M}_{k,n}} \|\mathbf{X}^c - \mathbf{U}\mathbf{V}\|_{\mathcal{F}}^2,$$

where $\mathbf{X}^c$ is the centered matrix (if A_j denotes the j-th column of the matrix $\mathbf{A}$, $X_j^c = X_j - \bar{X}_j$ with $\bar{X}_j = \sum_{i=1}^d X_{ij}/d$). The matrix of principal components is thus $\mathbf{X}^c \widehat{\mathbf{V}}_k^* = \widehat{\mathbf{U}}_k$. Each line of this $d \times k$ matrix contains the coordinates of the projection of the associated time series on the first k axis. Two projected time series are close if they share globally the same trend's property. However, in high-dimensional spaces, the Euclidean distance used in PCA can lose its meaning and a local trend similarity could be preferred. We thus propose to perform the PCA on the transformed matrix $\underline{\mathbf{X}}^c = \mathbf{X}^c \mathbf{\Lambda}^+$. The solution is $\widehat{\mathbf{M}}_{k,\tau} = \widehat{\underline{\mathbf{M}}}_{k,\tau} \mathbf{\Lambda} = \widehat{\mathbf{U}}_k \widehat{\mathbf{V}}_\tau \mathbf{\Lambda}$ and the matrix of principal components is $\underline{\mathbf{X}}^c \widehat{\mathbf{V}}_k^* = \mathbf{X}^c \mathbf{\Lambda}^+ \widehat{\mathbf{V}}_k^*$.

The projections of the times series on the principal plan (coordinates given by the two first columns of the matrix of principal components) of $\mathbf{X}^c$ and $\underline{\mathbf{X}}^c$ are plotted in Figure 9 on the left and the right respectively for the ECG dataset colored according to the true partition. Recall that for the ECG dataset, we consider that the trend is periodic with period $\tau = 32$.

To illustrate the trend reduction using $\mathbf{\Lambda}^+$ on a period with length τ, the 28th and the 121th time series of $\mathbf{X}^c$ and $\underline{\mathbf{X}}^c$ are plotted in Figure 10 on left and right respectively. As expected according to their features in both matrices, they are not close in the PCA of $\mathbf{X}^c$ whereas they are close in the PCA of $\underline{\mathbf{X}}^c$. That also explains the difference when the custering is performed without trend or with trend (see the previous section). For example, these two time series are clustered in the same group when a *trend property* is taking into account, whereas they are not in the same group when no *trend property* is considered.

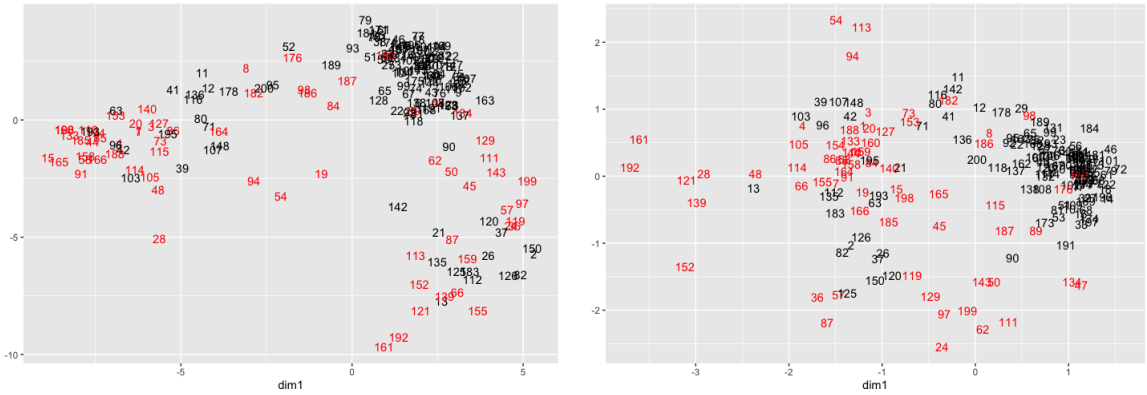


FIGURE 9. PCA on the ECG dataset. Left: on $\mathbf{X}^c$. Right: on $\underline{\mathbf{X}}^c$ with a periodic ($\tau = 32$) trend.

ACKNOWLEDGMENTS

This work has been conducted as part of the project Labex MME-DII (ANR11-LBX-0023- 01) and within the FP2M federation (CNRS FR 2036).

REFERENCES

[1] P. Alquier and N. Marie. Matrix Factorization for Multivariate Time Series Analysis. *Electronic Journal of Statistics* 13, 4346-4366, 2019.

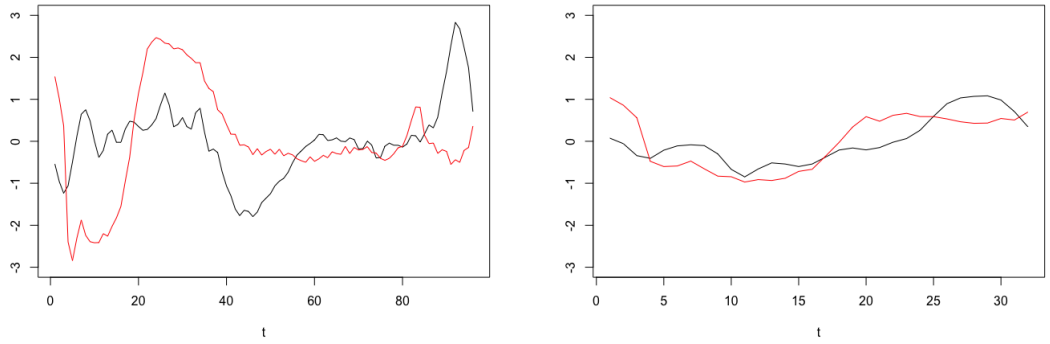


FIGURE 10. The 28th (in black) and the 121th (in red) time series. Left: on $\mathbf{X}^c$. Right: on $\underline{\mathbf{X}}^c$ with a periodic ($\tau = 32$) trend.

[2] P. Alquier, N. Marie and A. Rosier. Tight Risk Bound For High Dimensional Time Series Completion. *Electronic Journal of Statistics* 16, 1, 3001-3035, 2022.
[3] S. Arlot and P. Massart. Data-driven calibration of penalties for least-squares regression. *Journal of Machine Learning Research (JMLR)* 10, 245-279, 2009.
[4] S. Arlot. Minimal penalties and the slope heuristics: a survey. *Journal de la société française de statistique* 160, 3, 1-106, 2019.
[5] J.-P. Baudry, C. Maugis and B. Michel. Slope heuristics: overview and implementation. *Statistics and Computing*, 22, 2, 455-470, 2011.
[6] L. Birgé and P. Massart. Gaussian model selection. *Journal of the European Mathematical Society* 3, 203-268,2001.
[7] P.J. Brockwell and R.A. Davis. *Introduction to Time Series and Forecasting.* Springer, 2016.
[8] T. Cai and A. Zhang. ROP: Matrix Recovery via Rank-One Projections. *The Annals of Statistics* 43, 1, 102-138, 2015.
[9] X. Collilieux, E. Lebarbier and S. Robin. A factor model approach for the joint segmentation with between-series correlation. *Scandinavian Journal of Statistics* 46, 3, 686-705, 2019.
[10] P.S.P. Cowpertwait and A.V. Metcalfe. *Introductory Time Series with R.* Springer, 2009.
[11] E. Devijver, M. Gallopin and E. Perthame. Nonlinear network-based quantitative trait prediction from transcriptomic data. *arXiv preprint arXiv:1701.07899*, 2017.
[12] C. Gouriéroux and A. Monfort. *Time Series and Dynamic Models.* Cambridge University Press, 1997.
[13] T. Hastie and R. Mazumder. softImpute: Matrix Completion via Iterative Soft-Thresholded SVD. *URL https://CRAN. R-project. org/package= softImpute. R package version*, 2015.
[14] J. Jacques and C. Preda. Functional data clustering: a survey. *Advances in Data Analysis and Classification* 8, 3, 231-255, 2014.
[15] A. ustel and M. Svarc. Sequential clustering for functional data. *arXiv preprint arXiv:1603.03640*, 2016.
[16] O. Klopp, K. Lounici and A.B. Tsybakov. Robust Matrix Completion. *Probability Theory and Related Fields* 169, 1-2, 523-564, 2017.
[17] O. Klopp, Y. Lu, A.B. Tsybakov and H.H. Zhou. Structured Matrix Estimation and Completion. *Bernoulli* 25, 4B, 3883-3911, 2019.
[18] V. Koltchinskii, K. Lounici and A.B. Tsybakov. Nuclear-Norm Penalization and Optimal Rates for Noisy Low-Rank Matrix Completion. *The Annals of Statistics* 39, 5, 2302-2329, 2011.
[19] M. Lavielle. Using penalized contrasts for the change-point problem. *Signal Processing* 85, 8, 1501-1510, 2005.
[20] K. Moridomi, K. Hatano and E. Takimoto. Tighter Generalization Bounds for Matrix Completion via Factorization into Constrained Matrices. *IEICE Transactions on Information and Systems* 101, 8, 1997-2004, 2018.

[†]LABORATOIRE MODAL'X, UNIVERSITÉ PARIS NANTERRE, NANTERRE, FRANCE
Email address: emilie.lebarbier@parisnanterre.fr
Email address: nmarie@parisnanterre.fr

[◇]ESME SUDRIA, IVRY-SUR-SEINE, FRANCE
Email address: amelie.rosier@esme.fr