Co-Clustering of Multivariate Functional Data for the Analysis of Air Pollution in the South of France

Charles Bouveyron, Julien Jacques, Amandine Schmutz, Fanny Simoes, Silvia Bottini

Submitted to the Annals of Applied Statistics

CO-CLUSTERING OF MULTIVARIATE FUNCTIONAL DATA FOR THE ANALYSIS OF AIR POLLUTION IN THE SOUTH OF FRANCE

By Charles Bouveyron[*], Julien Jacques[†], Amandine Schmutz[†], Fanny Simões[‡], and Silvia Bottini[‡]

Université Côte d'Azur, Inria, CNRS, LJAD, Maasai, Nice, France[*], *Université de Lyon, Lyon 2, Laboratoire ERIC, EA 3083, Lyon, France*[†], *Université Côte d'Azur, MSI, Nice, France*[‡]

Nowadays, air pollution is a major treat for public health, with clear relationships with many diseases, especially cardiovascular ones. The spatio-temporal study of pollution is of great interest for governments and local authorities when deciding for public alerts or new city policies against pollution increase. The aim of this work is to study spatio-temporal profiles of environmental data collected in the south of France (Région Sud) by the public agency AtmoSud. The idea is to better understand the exposition to pollutants of inhabitants on a large territory with important differences in term of geography and urbanism. The data gather the recording of daily measurements of five environmental variables, namely three pollutants (PM10, NO_2, O_3) and two meteorological factors (pressure and temperature) over six years. Those data can be seen as multivariate functional data: quantitative entities evolving along time, for which there is a growing need of methods to summarize and understand them. For this purpose, a novel co-clustering model for multivariate functional data is defined. The model is based on a functional latent block model which assumes for each co-cluster a probabilistic distribution for multivariate functional principal component scores. A Stochastic EM algorithm, embedding a Gibbs sampler, is proposed for model inference, as well as a model selection criteria for choosing the number of co-clusters. The application of the proposed co-clustering algorithm on environmental data of the Région Sud allowed to divide the region composed by 357 zones in six macro-areas with common exposure to pollution. We showed that pollution profiles vary accordingly to the seasons and the patterns are similar during the 6 years studied. These results can be used by local authorities to develop specific programs to reduce pollution at the macro-area level and to identify specific periods of the year with high pollution peaks in order to set up specific health prevention programs. Overall, the proposed co-clustering approach is a powerful resource to analyse multivariate functional data in order to identify intrinsic data structure and summarize variables profiles over long periods of time.

1. Introduction. There is a growing body of evidence that air pollution is a significant threat for health worldwide (WHO, 2013). Air pollution is composed of particulate matter (PM) and gaseous pollutants, such as nitrogen dioxide (NO_2) and ozone (O_3) (IAR, 2016). The time exposure to air pollution leads diverse impact on the health. A short-term exposure to an intense pollution event increases hospital admission and mortality rate, causing mainly respiratory and cardiovascular diseases (Benbrahim-Tallaa et al., 2012; Hamra et al., 2014); whereas a long-term exposure reduces life expectancy (Lelieveld et al., 2015; Pascal et al., 2016). Although PM and NO_2 are mainly produced by human activities such as fossil fuel combustion, biomass burning due to agricultural activities, traffic, heating/cooling systems, industrial activities in general; O_3 is a secondary product, meaning that it is not directly produced by human activities. It is naturally produced in the stratosphere when highly energetic solar radiation strikes molecules of oxygen, O_2, and cause the two oxygen atoms to split apart in a process called photolysis. The Région Sud is an ideal context for the accumulation of this pollutant, regardless of the urbanization/industrialization of the subareas, because of its meteorological conditions: long sun days and rare rain events all over the year, especially in summer, where this pollutant is already at its highest level. Air pollution is a major concern not only in big cities but also in territories with medium-sized cities and mountainous zones. Mass of air moves from high pressure zones to low pressure and vice versa, thus particles matter can be moved from one zone to another along with the mass of air, exposing rural zones to pollution generated by adjacent industrial zones. Hence, pollution trend in specific zones does not depend only on the local pollution production, but is influenced by surrounding zones and meteorological factors. The understanding of air pollution and its spatio-temporal dynamic is of great interest for governments and local authorities in order to set up new city policies to lower down pollution or for public alerts when pollution increase above secure levels for the citizens. However, studies usually focus on isolated cities and do not take into account meteorological features, making their conclusions weak or not representative to generate prediction models. Their main limitation is due to the absence of powerful statistical or mathematical models able to analyse the complex spatio-temporal dynamic of pollution in big zones. Here we chose to model the Région Sud in the south of France, using a novel statistical method which allows studying the behavior of multivariate variables in order to understand pollution dynamic in this region. This study shows the ability of our co-clustering approach to identify intrinsic structures in these complex data that well suits to describe and analyse pollution behavior. Our results will allow local authorities to set up pollution politics adapted to the heterogeneous territory of the region and will give an instrument to analyse environmental data that can be expanded to other regions/countries.

1.1. *The Région Sud and the AtmoSud Agency.* The Région Sud (formerly known as Provence-Alpes-Côte d'Azur) covers a territory of 31,400 km^2 between Marseille, Nice and Gap, and hosts more than 5 millions of inhabitants. It has a wide variety of landscapes, from the Alps mountains to plains and coastal areas hosting big cities like Nice and Marseille. Most of the population of the region lives in the Mediterranean coastline in the south. The wide variety of landscapes and the unbalanced population distribution all over the territory, make the study and the modeling of air pollution difficult. Thus, the French Ministry of the Environment in 2012 created the AtmoSud agency to monitor the air quality in the Région Sud. Among its tasks, AtmoSud fulfills a mission of public interest by informing and educating citizens, the State, communities and economic actors about pollution trends offering decision support to implement the most relevant actions to improve air quality. AtmoSud relies on a set of eighty fix and mobile sensors (see sensor locations in Figure 1) which measure several pollutants and meteorological variables. Based on these daily measurements, AtmoSud is able to release every day a detailed map of the pollutants and pollution forecast for the coming days, with a resolution of 4 km, using the sophisticated model Chimere (Menut et al., 2013) to interpolate sensor measurements.

In the present study, we collected environmental data from AtmoSud agency, specifically daily pollution and meteorological factors measurements for the period from 2013 to 2018 including: the maximum daily value observed for NO_2 and O_3, the daily average value of PM10 and of maximal and minimal observed temperature (T) and pressure (P) for each of the 357 areas of the Région Sud. Figures 2 and 3 show a sample of the data. The raw data can be obtained from `https://www.atmosud.org` and the pretreatments we carried out are described in Section 5.1. The resulting pretreated data set is available at `https://github.com/UCA-MSI/AirQualityPACA_Data`.

1.2. *Functional co-clustering.* The aim of this work is to study the environmental database, constituted by 3 pollution and 2 weather-related variables, in order to identify spatio-temporal clusters representing peculiar pollution trends. Such data can be seen as multivariate functional data: multiple quantitative entities evolving during time collected simultaneously for the same individual (Ramsay and Silverman, 2005; Jacques and Preda, 2014b). In order to analyze and understand multivariate data, we propose to identify subgroups of individuals (i.e. areas, in the present work) which have a similar

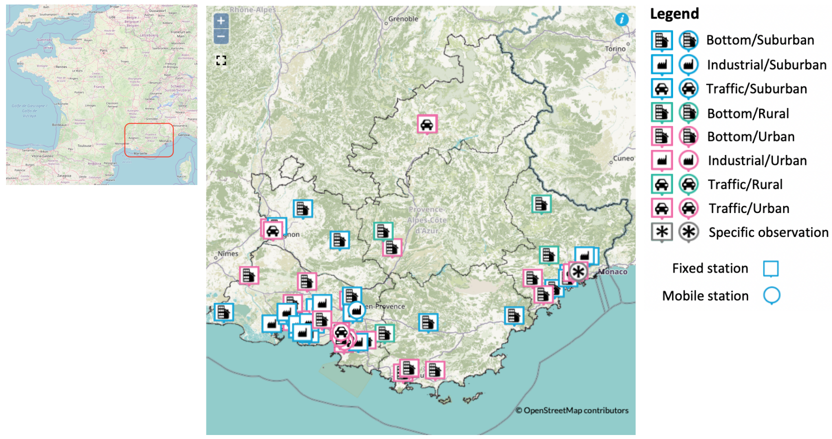

Fig 1: Map of the Région Sud and the locations of the pollution sensors.

profile for each of the 5 variables. This task could be resolved by multivariate functional data clustering techniques (Tokushige et al., 2007; Kayano et al., 2010; Ieva et al., 2013; Bouveyron et al., 2020), but the interpretation of the cluster means, which will be 6-years long curves in the present case, will be obviously difficult for a human experts, who in practice expect to have a weekly or monthly view of the problem. Thus, in order to provide useful summaries to local authority experts and decision makers, we split the 6-year period into weeks of 7 days, which is the most common analysis range for such problems. Consequently, we built a big matrix with 357 rows (areas) and 313 columns (weeks), in which each element is a 5-variate curve. To analyze such massive data, we propose to simultaneously cluster the rows into homogeneous groups of areas and the columns into homogeneous groups of weeks. Such kind of analysis is known as co-clustering (Govaert and Nadif, 2013). The co-clustering will identify homogeneous blocks of cities and weeks having a similar behavior according to the different environmental variables. It is important to note that, with such approach, temporal and spatial dependence that can occur in the data are ignored. Nevertheless, as it will be explained in Section 5.1, the meteorological and pollutants variables under study have usually very local effects in space and time, which makes this assumption acceptable in the current context.

In statistics and machine learning literature, methods for the co-clustering

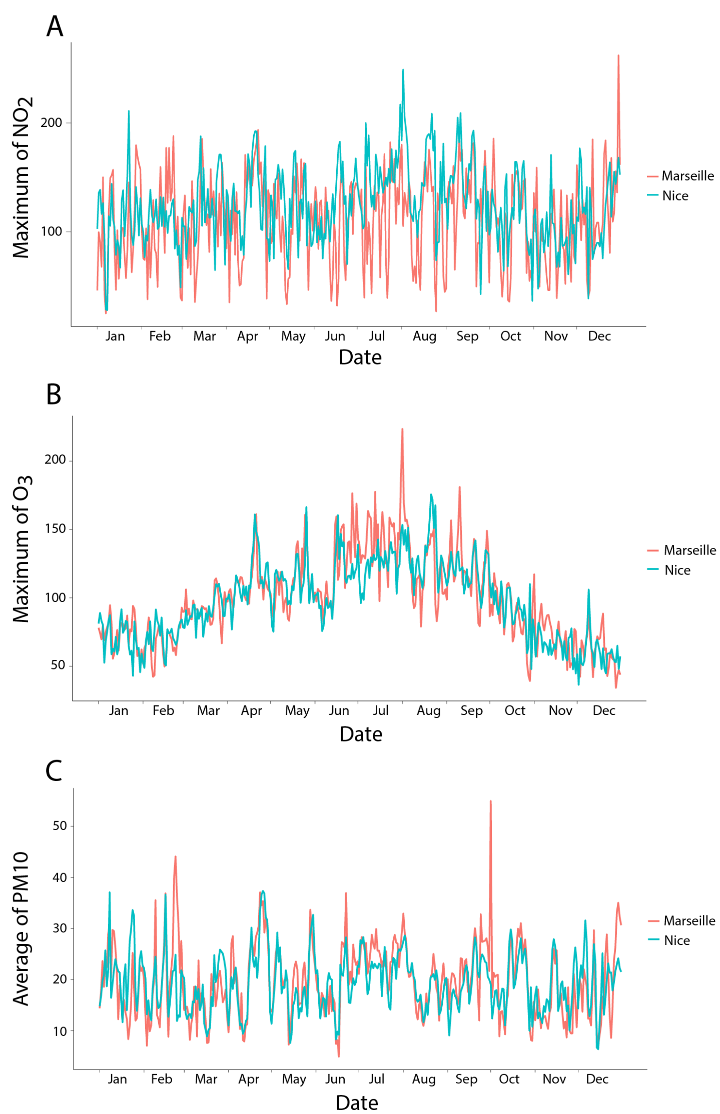

Fig 2: Daily distribution of pollutants in Marseille (red) and Nice (blue) for the year 2018: maximum of NO_2 (A) and of O_3 (B) and the average of PM10 (C).

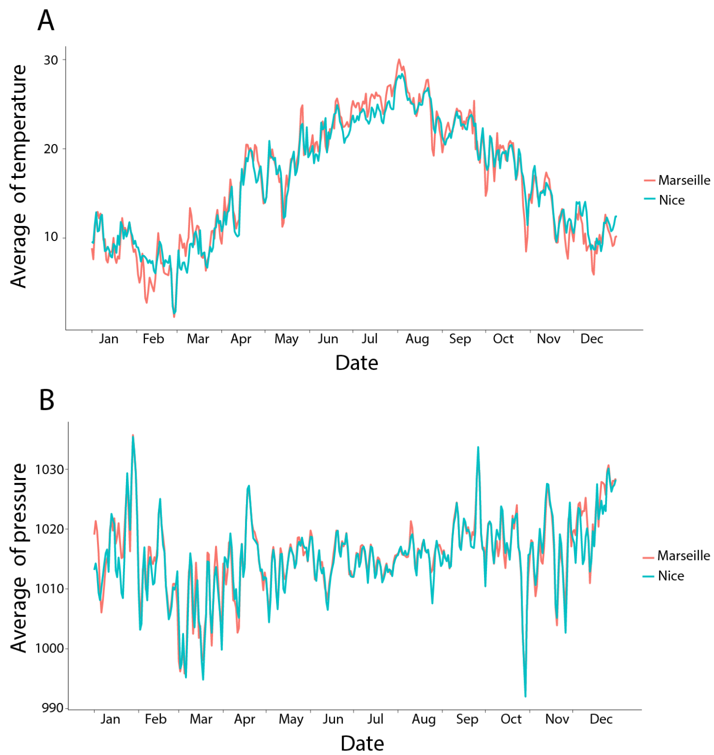

Fig 3: Daily distribution of meteorological factors in Marseille (red) and Nice (blue) for the year 2018: average of temperature (A) and average of pressure (B).

of rows and columns of a data matrix can be split into two main categories: deterministic approaches (see for instance George and Merugu, 2005; Banerjee et al., 2007; Wang and Huang, 2017) and model-based approaches (Govaert and Nadif, 2013; Bouveyron et al., 2019). The selection of the number of row and column clusters is one of the most important tasks in co-clustering analysis and model-based approaches provide a well defined framework for model selection. Moreover, model-based approaches are usually very flexible: taking into account groups of different sizes, they allow to manage different types of data. All these reasons prompted us to employ the model-based point of view.

One of the most famous model for co-clustering is the latent block model (LBM, Govaert and Nadif (2013)). According to the LBM, the elements of a block are modeled by a parametric distribution. Each block is therefore interpretable thanks to the block-distribution's parameters. Moreover, model selection criteria, such as the ICL criterion (Biernacki et al., 2000), can be used for model selection purposes, including the choice of the number of co-clusters. This technique proved its efficiency for co-clustering of several types of data: continuous (Nadif and Govaert, 2008), nominal (Bhatia et al., 2014), binary (Laclau et al., 2017), ordinal (Jacques and Biernacki, 2018; Corneli et al., 2019), functional data (Bouveyron et al., 2018; Chamroukhi and Biernacki, 2017; Ben Slimen et al., 2018) or even mixed-type data (Selosse et al., 2021).

Since our database is composed by functional variables, from now on we focused on functional data. Ben Slimen et al. (2018) proposed a co-clustering algorithm based on a vectorial LBM applied on the functional principal components scores of the curves. Bouveyron et al. (2018) extended this work by proposing a functional latent block model assuming that the functional principal components of the curves are block-specific and live into a low-dimensional subspace. Chamroukhi and Biernacki (2017) presented another co-clustering model based on a latent block model where the probability density function is estimated thanks to a regression model with a hidden logistic process. Unfortunately, all these works are designed for univariate functional data and are not able to handle in an appropriate way the multivariate functional data that we consider in this study.

1.3. *Contributions and organization of the work.* In the present work, a co-clustering algorithm for multivariate functional data is proposed to handle the environmental data of the Région Sud that we gathered thanks to the AtmoSud agency. The proposed algorithm, named *multiFunLBM*, extends to the multivariate case the methodology proposed by Bouveyron et al. (2018).

As it will be explained thereafter, Bouveyron et al. (2018) assume a Gaussian distribution on the expansion coefficients of the functional data into a predefined basis of functions. It is not possible to extend this approach directly to the multivariate case by concatenating the coefficients of the different functional variables, because this will increase drastically the coefficient vectors' dimensions, and thus will lead to the well-known curse of dimensionality issue. We consequently need to use some dimension reduction techniques, in this case Multivariate Functional Principal Component Analysis (Jacques and Preda, 2014b). However, in order not to lose any information with these techniques, we keep all the principal components but model them parsimoniously, with cluster-specific parsimonious Gaussian models. The application of the *multiFunLBM* algorithm to the AtmoSud data allowed to identify six spatio-temporal clusters which represent sub groups of areas and weeks with specific pollution trends.

The paper is organized as follows. Section 2 presents the co-clustering model and Section 3 is devoted to model inference. An experimental study of the algorithm on simulated data is presented in Section 4. Section 5 is dedicated to the analysis of the environmental database of the South of France. Some concluding remarks and further work are finally discussed in Section 6.

2. A co-clustering model for multivariate functional data. This section introduces a generative model for co-clustering multivariate functional data, such as the ones of AtmoSud.

2.1. *Data and functional reconstruction.* Functional data, which are the observations of a random variable living into an infinite dimensional space, are in practice observed only at a finite set of time points. Let $\boldsymbol{x} = (\boldsymbol{x}_{ij})_{1\leq i\leq n, 1\leq j\leq p}$ be the data matrix of dimension $n \times p$, where each element $\boldsymbol{x}_{ij}$ is a multivariate curve $\boldsymbol{x}_{ij} = (x^1_{ij}(t), \ldots, x^S_{ij}(t))$ with $t \in [0, T]$. Let us denote by i the row index, by j the column index and let s index the dimension of the multivariate curves. In our application, i refers to the cities (identified by postcodes, $n = 357$), j refers to a week of study ($p = 313$) and s refers either to some pollutant or to a weather-related variable ($S = 5$). Note that the model and its inference which are presented in this work can be nevertheless used for any other similar set of multivariate functional data.

In practice, the functional expressions of the curves $x^s_{ij}(t)$ are not known and we only have access to discrete observations at a finite set of times: $\{x^s_{ij}(t_1), x^s_{ij}(t_2), ...\}$. A common way to reconstruct the functional form is to assume that the observations can be decomposed into a finite dimensional space spanned by a basis of functions. So each observed curve x^s_{ij} $(1 \leq i \leq$

$n, 1 \le j \le p$, $1 \le s \le S$) can be expressed as a linear combination of basis functions $\{\phi_r^s\}_{r=1,\ldots,R_s}$:

$$x_{ij}^s(t) = \sum_{r=1}^{R_s} c_{ijr}^s \phi_r^s(t) \tag{1}$$

with R_s the number of basis functions used to reconstruct the sth functional variable. These basis functions can be for instance Fourier or spline bases. It is worth noticing that the choice of the most appropriate basis (as well as the number of basis functions) is an open problem in the unsupervised context (Jacques and Preda, 2014a). In practice, this choice is done empirically such that the reconstruction is judged reasonable by the expert. Estimation of the basis expansion coefficients c_{ijr}^s is classically done by least squares smoothing. We refer the reader to Ramsay and Silverman (2005) for a complete survey on this aspect.

Let $c = (c_{ij})_{1 \le i \le n, 1 \le j \le p}$ be the whole set of coefficients, which contains the coefficients for the row i and column j which corresponds to the concatenation of coefficients c_{ijr}^s for all S functional variables, such that $c_{ij} = (c_{ij1}^1, \ldots, c_{ijR_1}^1, \ldots, c_{ij1}^S, \ldots, c_{ijR_S}^S)^t$.

For the sake of presentation clarity, the same number of basis functions, $R_s = R, \forall s = 1, ..., S$, and the same basis functions, $\{\phi_{rs}\}_{r=1,\ldots,R} = \{\phi_r\}_{r=1,\ldots,R}, \forall s = 1, ..., S$, are considered hereafter for each dimension of the multivariate functional variables. Extension is straightforward. Let consequently $\phi(t)$ be the $S \times SR$ matrix that gathers basis functions of all S functional variables:

$$\phi(t) = \begin{pmatrix} \phi_1(t) & ... & \phi_R(t) & 0 & ... & 0 & ... & 0 & ... & 0 \\ 0 & ... & 0 & \phi_1(t) & ... & \phi_R(t) & ... & 0 & ... & 0 \\ & & & & ... & & & & & \\ 0 & ... & 0 & 0 & ... & 0 & ... & \phi_1(t) & ... & \phi_R(t) \end{pmatrix}.$$

With these notations, Equation 1 can be written in matrix terms:

$$x_{ij}(t) = \phi(t) c_{ij}. \tag{2}$$

It is worth noticing that, depending on the basis function chosen, the basis functions may not be orthonormal, *i.e.* $\Phi = \int_0^T \phi(t)^t \phi(t) dt$ may not be the identity matrix. This is specifically the case when considering B-splines or polynomial functions. Conversely, Fourier basis functions are by construction such that $\Phi = I$. In any case, it is of course possible to express the expansion coefficients within an orthonormal basis function system:

$$x_{ij}(t) = \psi(t) \Phi^{1/2} c_{ij}, \tag{3}$$

where $\psi(t) = \phi(t)\Phi^{-1/2}$. Thus, the expansion coefficients of $x_{ij}(t)$ within the orthonormal basis $\{\psi_1(t), ..., \psi_M(t)\}$ are $\Phi^{1/2}c_{ij}$, where $M = S \times R$.

2.2. *The proposed latent block model.* The aim of a co-clustering model is to define row and column partitions in order to summarize the data matrix $\boldsymbol{x}$ into smaller subgroups, usually called blocks, that are eventually distributed in the same way. To this end, let $z = (z_{ik})_{1\leq i\leq n, 1\leq k\leq K}$ be the row partition of the n rows into K groups, and $w = (w_{jl})_{1\leq j\leq p, 1\leq l\leq L}$ the column partition of the p columns into L groups, such as $z_{ik} = 1$ if row i belongs to row-cluster k and 0 otherwise (and similarly for w_{jl}). Thus, one block is defined by a set of curves which belong to a row and column cluster such as $z_{ik}w_{jl} = 1$.

Let us first assume that the partitions z and w are independent:

$$p(x; \Theta) = \sum_{z\in Z}\sum_{w\in W} p(z; \Theta)p(w; \Theta)p(x|z, w; \Theta) \tag{4}$$

where Z is the set of all possible rows partitions into K groups and W the set of all possible columns partitions into L groups. Let us introduce α_k and β_l as the row and column mixing proportions (belonging to $[0, 1]$ and summing to 1), such that:

$$p(z; \Theta) = \prod_{ik} \alpha_k^{z_{ik}} \text{ and } p(w; \Theta) = \prod_{jl} \beta_l^{w_{jl}}.$$

Let us also assume that, conditionally on (z, w), the curves x_{ij} are independent and generated by a block-specific distribution:

$$p(x|z, w; \Theta) = \prod_{ijkl} p(x_{ij}; \Theta_{kl})^{z_{ik}w_{jl}}. \tag{5}$$

Unfortunately, the notion of probability density for functional variable is not well defined. In Delaigle and Hall (2010), it is proved that it can be approximated with the probability density of the functional principal components scores (FPCA, Ramsay and Silverman (2005)). Under the assumption (2) of basis expansion decomposition, these FPCA scores are obtained directly from a PCA of the coefficient c using a metric Φ defined by the scalar product between the basis function. Consequently, model-based approaches for functional data consider probabilistic distribution for either the FPCA scores (Jacques and Preda, 2013, 2014b) or the basis expansion coefficients (Bouveyron et al., 2015, 2020), but it is equivalent.

In addition, depending on the application, the period of observation $[0, T]$ can be long, and the number of basis functions used for reconstruction can be large. Consequently, the coefficient vectors c_{ij} may live in high dimensions. In order to suggest a parsimonious data modeling and to avoid the curse of dimensionality, we further suppose that the curves of each block (k, l), for $k = 1, ..., K$ and $l = 1, ..., L$, can be described into a low-dimensional functional latent subspace specific to each block, with intrinsic dimension $d_{kl} < M = S \times R$. As it will be demonstrated below, this low-dimensional description can be obtained through a principal component analysis for multivariate functional data (MFPCA, Jacques and Preda, 2014b) performed per block. MFPCA is an extension of FPCA to the multivariate functional case, which represents the multivariate curves by a vector of principal scores into an eigenspace formed by multivariate eigenfunctions.

Thus, conditionally to its belonging to block (k, l), each multivariate curve x_{ij} can be represented by its latent counterpart $\delta_{ij} \in \mathbb{R}^{d_{kl}}$. Let us define Q_{kl} the orthogonal matrix of dimension $M \times M$, that can be split into two parts: $Q_{kl} = [U_{kl}, V_{kl}]$ with U_{kl} of dimension $M \times d_{kl}$ and V_{kl} of dimension $M \times (M - d_{kl})$. With these notations, the linear mapping from the original space of c_{ij} to the low-dimensional functional subspace can be written:

$$c_{ij} = \Phi^{-1/2} U_{kl} \delta_{ij} + \epsilon_{ij}.$$

Let us recall that depending on the basis function choice, the orthonormalization matrix Φ may be equal to I (Fourier basis) or not (B-splines, polynomial functions).

Conditionally to the blocks, the latent representations are further assumed to follow a Gaussian distribution with a parsimonious parametrization of the covariance matrix:

$$\delta_{ij} | z_{ik} w_{jl} = 1 \sim \mathcal{N}(m_{kl}, \Delta_{kl}), \tag{6}$$

with $m_{kl} \in \mathbb{R}^{d_{kl}}$ and $\Delta_{kl} = diag(a_{kl1}, \ldots, a_{kld_{kl}})$. Additionally, ϵ_{ij} is assumed to have a centred Gaussian distribution:

$$\epsilon_{ij} | z_{ik} w_{jl} = 1 \sim \mathcal{N}(0, \Xi_{kl})$$

These assumptions induce a Gaussian distribution for the basis expansion coefficients:

$$c_{ij} | z_{ik} w_{jl} = 1 \sim \mathcal{N}(\mu_{kl}, \Sigma_{kl})$$

where $\mu_{kl} = \Phi^{-1/2} U_{kl} m_{kl}$ and $\Sigma_{kl} = \Phi^{-1/2} U_{kl} \Delta_{kl} U_{kl}^t \Phi^{-1/2} + \Xi_{kl}$. Finally,

Ξ_{kl} is assumed to be such

$$Q_{kl}^t\Phi^{1/2}\Sigma_{kl}\Phi^{1/2}Q_{kl} = \left(\begin{array}{cc} \boxed{\begin{array}{ccc} a_{kl1} & & 0\\ & \ddots & \\ 0 & & a_{kld_{kl}}\end{array}} & \mathbf{0} \\ \mathbf{0} & \boxed{\begin{array}{ccc} b_{kl} & & 0 \\ & \ddots & \\ 0 & & b_{kl}\end{array}} \end{array}\right) \begin{array}{l} \left.\vphantom{\begin{array}{c}a\\ \ddots\\ 0\end{array}}\right\} d_{kl} \\ \left.\vphantom{\begin{array}{c}b\\ \ddots\\ 0\end{array}}\right\} M-d_{kl}\end{array}$$

where $a_{kl1} > \ldots > a_{kld_{kl}} > b_{kl}$. With this assumption, the first d_{kl} values express the main part of the variability of the data, while the remaining ones reflect the variance of the noise and are modeled by a unique parameter b_{kl}. Thus, the space spanned by the columns of U_{kl} is a low-dimensional space which contains the main part of information about the data of the block (k, l). The remaining information is considered as noise and modeled by a unique variance parameters b_{kl} for the block (k, l).

Thus, $p(x_{ij}; \Theta_{kl})$ in Eq.(5) can be approximated by $p(c_{ij}; \mu_{kl}, a_{kl\cdot}, b_{kl}, Q_{kl})$. Let us finally introduce $\theta_{kl} = (\mu_{kl}, a_{kl\cdot}, b_{kl}, Q_{kl})$. The whole set of model parameters is finally denoted by $\theta = (\alpha_k, \beta_l, \theta_{kl})_{1\leq k\leq K, 1\leq l\leq L}$.

2.3. *A family of parsimonious models.* In order to provide more parsimonious models, additional assumptions can be made on the different parameters $a_{kl\cdot}$, b_{kl} and d_{kl}, considering that they are common between clusters or dimensions. This approach allows to generate a family of sub-models of the general model introduced above. In this paper, we will detail the inference procedure to the sub-model assuming $a_{klm} = a_{kl}, \forall m = 1, ..., d_{kl}$, since a good behavior has been observed in practice. Nevertheless, the co-clustering method presented here can be derived for all models of the family extension, following the approach detailed in Bouveyron et al. (2020).

3. Model inference. This section focuses on model inference *via* a SEM-Gibbs algorithm. Model selection and initialization will be also discussed.

3.1. *Model inference through a SEM-Gibbs algorithm.* Since we consider the task of co-clustering, the goal is to estimate the unknown row and column partitions z_{ik} and w_{jl} given the data at hand. Usually, the maximum a posteriori rule is used, based on the estimation of model parameter θ maximizing the observed log-likelihood $L(c; \theta) = \log p(c; \theta)$.

In such a case where latent variables are involved (z_{ik} and w_{jl}), one would instinctively use the expectation-maximization (EM) algorithm (Dempster et al., 1977) to find a candidate $\hat{\theta}$ for the maximum of the log-likelihood. The EM algorithm alternates two steps, the E and M steps, in order to create a converging series of $\theta^{(h)}$ by optimizing a lower bound of the log-likelihood.

This lower bound can be easily exhibited by rewriting the log-likelihood function as follows:

$$\log(p(c|\theta)) = \mathcal{L}(q;\theta) + KL(q||p(\cdot|c,\theta)),$$

where $\mathcal{L}(q;\theta) = \sum_{z,w} q(z,w)\log(p(c,z,w,|\theta)/q(z,w))$ is a lower bound of the log-likelihood and $KL(q||p(\cdot|c,\theta)) = -\sum_{z,w} q(z,w)\log(p(z,w|c,\theta)/q(z,w))$ is the Kullback-Leibler divergence between q and $p(\cdot|c,\theta)$.

The E step of the EM algorithm consists in maximizing the lower bound $\mathcal{L}$ over q for a given value of θ. A straightforward calculation shows that $\mathcal{L}$ is maximized for $q^*(z,w) = p(z,w|c,\theta)$. Unfortunately, in our case, the joint posterior distribution $p(z,w|c,\theta)$ is not tractable as well.

In order to overcome this additional issue, we propose to make use of a Gibbs sampler within the E step to approximate the posterior distribution $p(z,w|c,\theta)$. We refer to Keribin et al. (2010) for a discussion on the inference algorithms in the case of latent block models. The resulting stochastic version of the EM algorithm, called SEM-Gibbs hereafter, has the following structure (at iteration h and starting from an initial column partition $w^{(0)}$ and initial parameter value $\theta^{(0)}$):

- SE step: θ is fixed and $q^*(z,w) \simeq p(z,w|c,\theta)$ is approximated with a Gibbs sampler. The Gibbs sampler consists in alternating the two following steps a certain number of times to simulate the unknown labels with their conditional distribution knowing the observations and a current estimation of the parameters:
  - simulate $z^{(h+1)}|c,w^{(h)}$ according to:

    $$p(z_{ik}=1|c,w^{(h)};\theta^{(h)}) = \frac{\alpha_k^{(h)} f_k(c_i|w^{(h)};\theta^{(h)})}{\sum_{k'} \alpha_{k'}^{(h)} f_{k'}(c_i|w^{(h)};\theta^{(h)})}$$

    with $f_k(c_i|w^{(h)};\theta^{(h)}) = \prod_{jl} p(c_{ij};\theta_{kl}^{(h)})^{w_{jl}^{(h)}}$.
  - simulate $w^{(h+1)}|c,z^{(h+1)}$ according to:

    $$p(w_{jl}=1|c,z^{(h+1)};\theta^{(h)}) = \frac{\beta_l^{(h)} f_l(c_j|z^{(h+1)};\theta^{(h)})}{\sum_{l'} \beta_{l'}^{(h)} f_{l'}(c_j|z^{(h+1)};\theta^{(h)}).}$$

    with $f_l(c_j|z^{(h+1)};\theta^{(h)}) = \prod_{ik} p(c_{ij};\theta_{kl}^{(h)})^{z_{ik}^{(h)}}$.

- M step: $\mathcal{L}(q^*(z,w),\theta^{old})$ is now maximized over θ, where:

$$\mathcal{L}(q^*(z,w),\theta^{old}) \simeq \sum_{z,w} p(z,w|c,\theta^{old}) \log(p(c,z,w|\theta)/p(z,w|c,\theta^{old}))$$
$$\simeq E[\log(p(c,z^{(h+1)},w^{(h+1)}|\theta)|\theta^{old}] + \xi,$$

ξ being a constant term regarding θ. This step therefore reduces to the maximization of the conditional expectation of the complete-data log-likelihood given c, $z^{(h+1)}$ and $w^{(h+1)}$ (Appendix A.1 provides a developed form of $E[\log(p(c,z^{(h+1)},w^{(h+1)}|\theta)|\theta^{old}])$ and leads to the following updates for model parameters.
The mixing proportion and the block mean are updated as follows:
  - $\alpha_k^{(h+1)} = \frac{1}{n}\sum_i z_{ik}^{(h+1)}$ and $\beta_l^{(h+1)} = \frac{1}{p}\sum_j w_{jl}^{(h+1)}$,
  - $\mu_{kl}^{(h+1)} = \frac{1}{n_{kl}^{(h+1)}}\sum_{ij} z_{ik}^{(h+1)} w_{jl}^{(h+1)} c_{ij}$ with $n_{kl}^{(h+1)} = \sum_{ij} z_{ik}^{(h+1)} w_{jl}^{(h+1)}$.

For the variance parameters a_{kl}, b_{kl} and Q_{kl}, let us define the sample covariance matrix $\Omega_{kl}^{(h)}$ of block kl at step h:

$$\Omega_{kl}^{(h)} = \frac{1}{n_{kl}^{(h)}} \sum_{i=1}^{n}\sum_{j=1}^{M} z_{ik}^{(h+1)} w_{jl}^{(h+1)} (c_{ij}-\mu_{kl}^{(h)})^t (c_{ij}-\mu_{kl}^{(h)}).$$

Then, the updates for a_{kl}, b_{kl} and Q_{kl} are:
  - the variance parameters $a_{kl}^{(h+1)}$, are updated by the mean of the d_{kl} largest eigenvalues of $\Phi^{1/2}\Omega_{kl}^{(h)}\Phi^{1/2}$,
  - the variance parameters b_{kl} are updated by $\frac{1}{M-d_{kl}}(trace(\Phi^{1/2}\Omega_{kl}^{(h)}\Phi^{1/2}) - d_{kl}a_{kl}^{(h)})$,
  - the d_{kl} first columns of the matrix of eigenfunctions coefficients $Q_{kl}^{(h)}$ are updated by the eigenfunctions coefficients associated with the largest eigenvalues of $\Phi^{1/2}\Omega_{kl}^{(h)}\Phi^{1/2}$.

Proofs of those results are available in Appendices A.2, A.3 and A.4.

3.2. *Algorithmic considerations.*

Implementation. Regarding the practical implementation, the SEM-Gibbs algorithm is run for a given number of iterations. After a burn-in period, the final estimation $\hat{\theta}$ of the parameters is obtained by the mean of the sample distribution (without the burn-in iterations). Then, a new Gibbs sampler is used to sample $(\hat{z},\hat{w})$ according to $\hat{\theta}$, and the final partition $(\hat{z},\hat{w})$ is obtained by the marginal mode of this sample distribution.

Initialization of the algorithm. As said previously, *multiFunLBM* relies on a SEM-Gibbs algorithm. This algorithm needs to be initialized carefully with values for column partitions and parameters, or similarly with both column and row partitions. To this end, we consider the three following initialization strategies: *random*, *k-means* and *funFEM*. In the *random* case, row and column partitions are randomly sampled from a multinomial distribution with uniform probabilities. The *k-means* strategy consists in initializing the two partitions with those obtained by *k-means* directly applied on a discretized version of the data matrix and its transpose. Finally, the *funFEM* strategy initializes the partitions by applying the *funFEM* algorithm (Bouveyron et al., 2015) on the matrix concatenating the functional variables and its transpose. We will see later, in the numerical experimentation section, that *funFEM* is the one that gives the best results.

3.3. *Choice of the number of clusters.* We now discuss the choice of the hyper-parameters K and L, *i.e.* the number of row clusters and column clusters respectively. The choice of these hyper-parameters is viewed here as a model selection problem. Well established model selection tools are Akaike information criterion (AIC, Akaike 1974), Bayesian information criterion (BIC, Schwarz 1978) and Integrated Classification Likelihood (ICL, Biernacki et al. 2000). However, in the co-clustering case, the likelihood is not tractable for the same reason than the EM algorithm is not usable. Consequently, AIC and BIC are not tractable. Conversely, the ICL criterion can still be considered since it relies on the completed data log-likelihood, which is tractable. Adapted to our model, the ICL criterion is:

$$ICL(K,L) = \log p(c,\hat{z},\hat{w};\hat{\theta}) - \frac{K-1}{2}log(n) - \frac{L-1}{2}log(p) - \frac{\nu}{2}log(np)$$

where $\nu = KLM + 2KL + \sum_{kl} d_{kl}(M - \frac{d_{kl}+1}{2})$ is the number of continuous parameters per block and

$$log\ p(c,\hat{z},\hat{w};\hat{\theta}) = \prod_{ik} \hat{z}_{ik} log(\alpha_k) + \prod_{jl} \hat{w}_{jl} log(\beta_l) + \sum_{ijkl} \hat{z}_{ik}\hat{w}_{jl} log\ p(c_{ij};\hat{\theta}_{kl}).$$

The couple (K^*, L^*) leading to the highest ICL value is selected as the most appropriate number of row and column clusters.

4. Numerical experimentation on simulated data. This section presents numerical experiments on simulated data in order to illustrate the behavior of the proposed methodology in presence of different noise ratio in the data and to study the selection of the number of row and column clusters. The R code for *multiFunLBM* is available on CRAN through the new version of the `funLBM` package for R (Bouveyron et al., 2020).

4.1. *Simulation setup.* We first detail here the simulation setup that is used in the following numerical experiments. Bivariate curves ($S = 2$) are simulated with $K = 4$, $L = 3$. The proportions of row clusters α used is (0.2,0.4,0.1,0.3) and column clusters β is (0.4,0.3,0.3). The first functional variable is designed from four different functions that are used as blocks mean at 31 equispaced time points, $t = 0, 1/30, 2/30, ..., 1$:

$$x_{ij}(t)|z_{ik}w_{jl} = 1 \sim \mathcal{N}(m_{kl}(t), s^2),$$

where $s = 0.3$. The block mean functions m_{kl} are such that $m_{11} = m_{21} = m_{33} = m_{42} = f_1$, $m_{12} = m_{22} = m_{31} = f_2$, $m_{13} = m_{32} = f_3$ and $m_{23} = m_{41} = m_{43} = f_4$, with $f_1(t) = sin(4\pi t)$, $f_2(t) = 0.75 - 0.5\mathbb{1}_{t\in]0.7,0.9[}$, $f_3(t) = h(t)/max(h(t))$ where $h(t) = \mathcal{N}(0.2, \sqrt{0.02})$ and $f_4(t) = sin(10\pi t)$. Then the second variable is designed according to the same process than the first one but with four different functions: $f_1(t) = cos(4\pi t)$, $f_2(t) = 0.75 - 0.5\mathbb{1}_{t\in]0.2,0.4[}$, $f_3(t) = h(t)/max(h(t))$ where $h(t) = \mathcal{N}(0.2, \sqrt{0.05})$ and $f_4(t) = cos(10\pi t)$. The block means functions of the two functional variables are shown on Figure 4.

Starting from this simulation setting, five scenarios are derived by adding some noise fraction within the blocks by randomly simulating a percentage τ of curves using other block means: 0% (scenario 1), 10% (scenario 2), 30% (scenario 3), 50% (scenario 4) and 80% (scenario 5).

Regarding the algorithm setup, we set to 50 iterations the burn-in period of the algorithm and the SEM-Gibbs number of iterations is set to 100. Computation time on a 2,3 GHz Intel Core i7 with 32 Go RAM for one execution of the algorithm with *funFEM* initialization is about 30 seconds for $n = p = 100$ and 6 minutes for $n = p = 500$. In practice, we advice to use parallel computing to execute simultaneously the algorithm with different values of K and L (and select the solution leading to the best ICL criterion).

4.2. *Robustness to noise and influence of initialization.* This first experiments aims at studying the ability of *multiFunLBM* to recover the simulated model in clean but also noisy situations, and depending on the type of initialization of the SEM-Gibbs algorithm. To this end, 20 simulations with $n = p = 100$ have been performed for each scenario with both *k-means*, *funFEM* and *random* initializations. The algorithm is applied for $K = 4$ and $L = 3$ and with Fourier smoothing with 15 basis functions. The quality of estimated partitions is assessed with the Adjusted Rand Index (ARI, Rand 1971). We recall that an ARI of 1 indicates that the partition provided by the algorithm is perfectly aligned with the simulated one. Conversely, an ARI of 0 indicates that the two partitions are just some random matches.

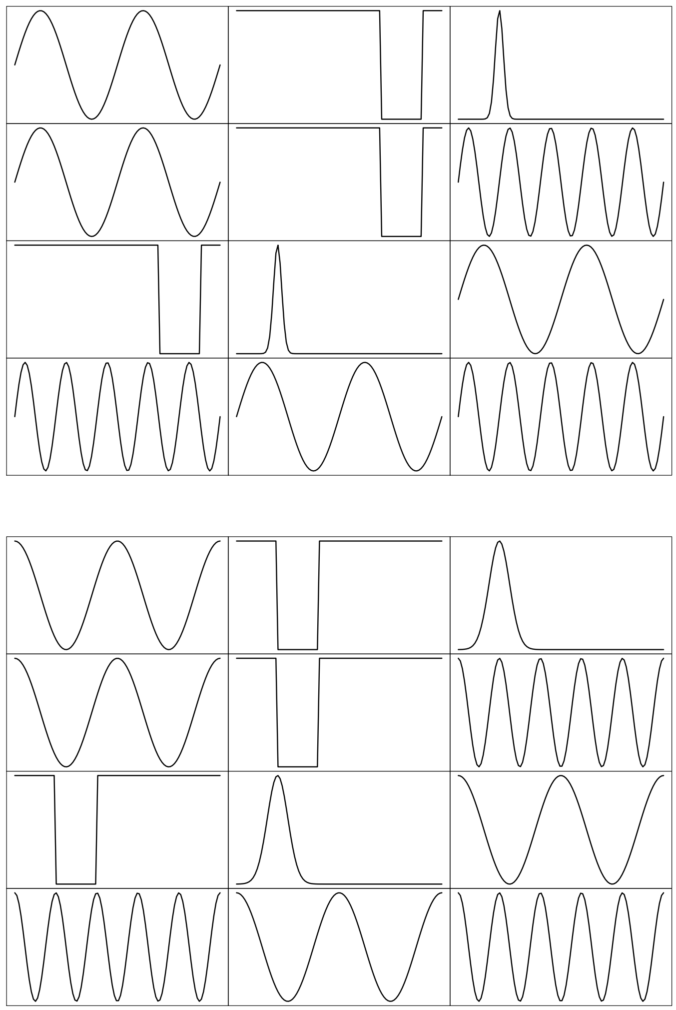

Fig 4: Block means functions for the first variable (top) and second variable (bottom). On each plot, the row number correspond to k and the column number to l

Results are shown in Figure 5 for both row (left panels) and columns partitions (right panels). We can see that co-clustering results are almost perfect for the 4 first scenarios with *funFEM* initialization (bottom panels), and the 2 first scenarios for *k-means* initialization. As expected, the algorithm performance decreases while noise increases, but median ARI value is always above 0.8 in the case of four first scenarios with *k-means* and *funFEM* initialization. Moreover *k-means* initialization performs better than random when the noise ratio is upper than 50%. And the *funFEM* initialization performs better than the *k-means* one. In view of the good behavior of *funFEM* initialization, we recommend to use the *funFEM* initialization rather than the two others available in the algorithm. Of course, if different initializations are used, the user will have to select the solution with the highest log-likelihood.

Let's note that in these experiments, we consider symmetric number of rows and columns ($n = p$). Contrary to usual statistical models in which the number n of observations and the number p of variables do not have the same importance (the greater n the better is the inference, the greater p the worse is the inference), in co-clustering their roles are totally symmetric: the quality of the inference increase both with n and p (Keribin et al., 2015). This is confirmed by the results obtained with a larger sample size ($n = p = 500$), available in Appendix A.5, which are better than those with $n = p = 100$.

4.3. *Model selection.* In this section, the selection of the number of clusters using the ICL criterion we derived earlier is investigated. Data are generated as previously. The simulation setting is repeated 20 times. For each of the 20 generated data sets, the SEM-Gibbs algorithm is run with K and L values ranging from 2 to 6 clusters, with *funFEM* initialization. Each time, the model selected by ICL is reported.

For a sample size of $n = p = 500$, results are shown in Table 1. The results indicate that *multiFunLBM* in combination with the ICL criterions is able to perfectly recover the actual model with a noise ratio from 0 to 30% of the data volume. Then, as expected, the performance of the criterion decreases. However, for a noise ratio of 50%, the ICL criterion is still able to identify the right simulation model in 90% of the cases. Finally, for 80% of noise, the algorithm is not able to recover the partitions (Figure 5), the ICL criterion is also not able to find the right number of co-clusters.

For a smaller sample size, $n = p = 100$, the ICL criterion tends to select less clusters that it should do (results are in Appendix A.6). This result is not surprising since such a criterion has asymptotic properties (Keribin et al., 2015). In practice, selecting a too small number of clusters in clustering or

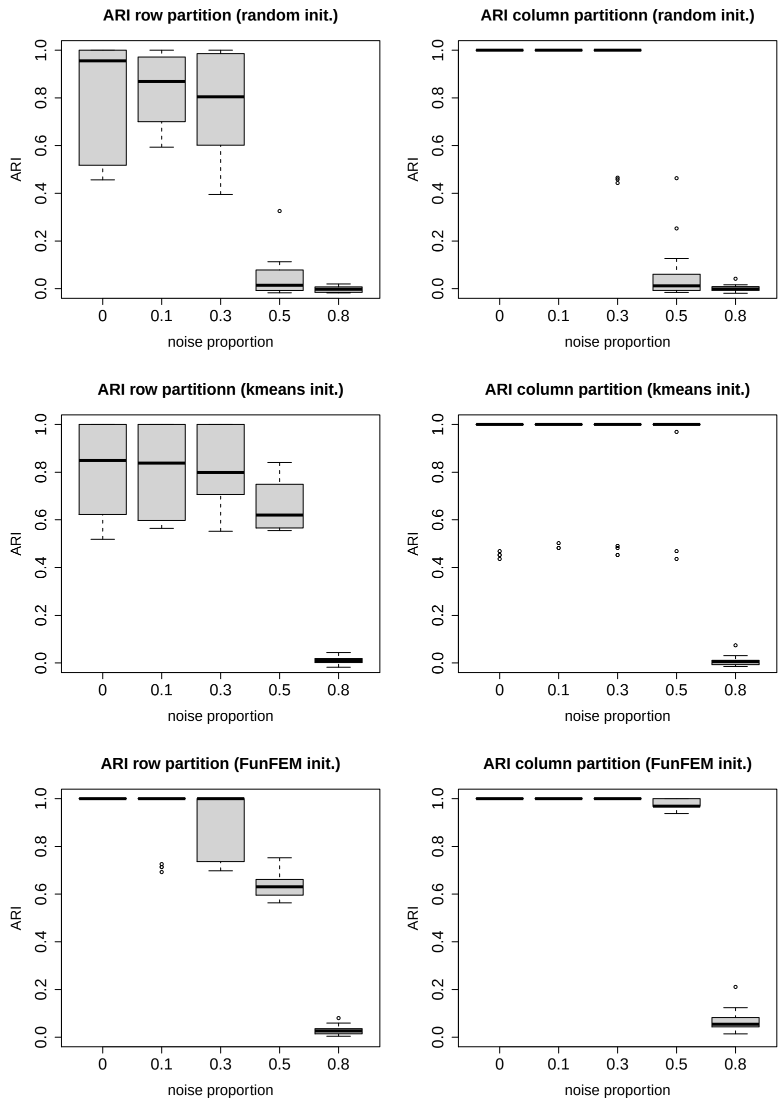

Fig 5: ARI results for *multiFunLBM* with $n = p = 100$ according to the noise ratio and depending on the type of initialization: random (top), *k-means* (middle) and *funFEM* (bottom).

co-clustering is not problematic, it will just lead to less precise analyses and interpretations of the data. If we want to have more precise analyses, the sample size should be sufficiently large.

TABLE 1
Percentage of selection of each model (K, L) by ICL among the 20 simulated data sets, with $n = p = 500$. The actual values for (K, L) are $(4, 3)$.

Scenario $\tau = 0$						Scenario $\tau = 0.3$						Scenario $\tau = 0.5$					
K/L	2	3	4	5	6	K/L	2	3	4	5	6	K/L	2	3	4	5	6
2	0	0	0	0	0	2	0	0	0	0	0	2	0	0	0	0	0
3	0	100	0	0	0	3	0	0	0	0	0	3	0	0	0	0	0
4	0	**0**	0	0	0	4	0	**100**	0	0	0	4	0	**90**	0	0	0
5	0	0	0	0	0	5	0	0	0	0	0	5	0	10	0	0	0
6	0	0	0	0	0	6	0	0	0	0	0	6	0	0	0	0	0

4.4. *Robustness to spatial and temporal dependences.* In the proposed co-clustering algorithm, the row and column partitions z and w are assumed to be independent (Eq. 4). This assumption, which can be seen as a strong one, is in fact useful when combined with the idea of clusters since it forces the model to find row and column clusters that carry out the data structure. In the present experiment, the robustness of this assumption is evaluated since many applications from real-world, such as the one presented in the next section, may depart from this hypothesis. To this end, the robustness to spatial and temporal dependences is evaluated by introducing dependence between the rows and/or the columns of x.

Data are simulated according to the previous simulation setting, and then spatial and temporal dependences are introduced as follows. For the column dependence, which corresponds in the application under study in this paper to a temporal dependence, each simulated curve $x^s_{ij}(t)$ is multiplied by a factor $\tau j/p$, where τ is a parameter controlling the dependence strength. Thus, greater is the index j of the curve, larger is the multiplication factor of the function $x^s_{ij}(t)$. This dependence simulates an increasing trend in the observation.

For the row dependence, a 2-dimensional spatial trend is simulated, by assuming that the n observations are spatially distributed onto a square of size $\sqrt{n} \times \sqrt{n}$, the first observation ($i = 1$) being on the bottom-left corner and the last one ($i = n$) on the top-right corner. The trend increases from the bottom-left corner to the top-right corner as illustrated by Figure 6. The trend is linear and such that the first observation is multiplied by a factor 1 and the n-th observation by a factor τ.

Figure 7 shows the evolution of the ARI on rows and columns regarding the dependency parameter τ, which we vary in the range $[1, 2]$. As one can observe, the proposed methodology turns out to be relatively robust to row

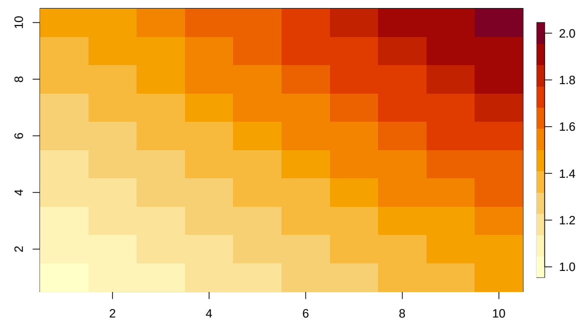

Fig 6: Simulated spatial dependency between the rows. The dependency parameter is here fixed to $\tau = 2$.

and column dependencies, with an ARI slowly decreasing until about 0.4 when the strength of the dependence is the highest. We can therefore expect the *multiFunLBM* algorithm to perform well in real-world situation where moderate spatial and temporal dependencies are present.

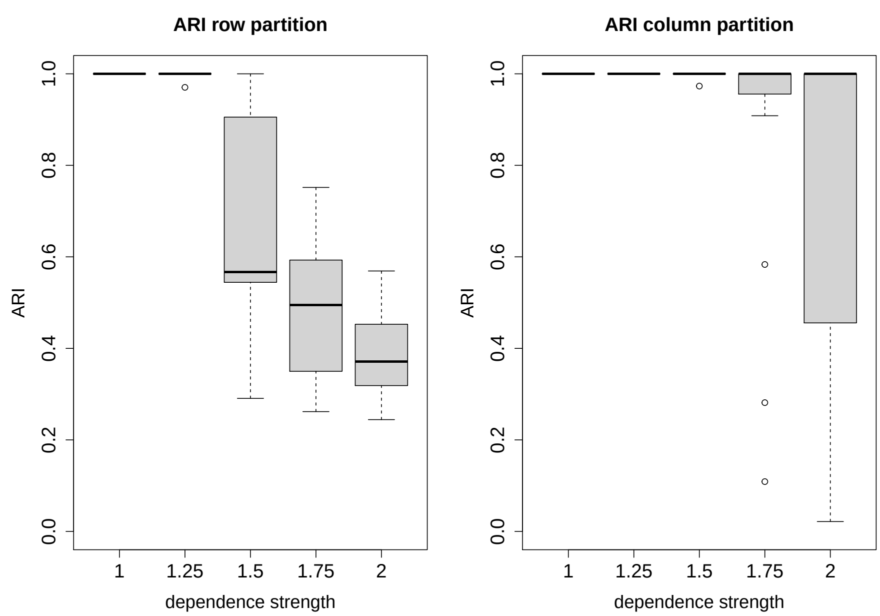

Fig 7: ARI results for *multiFunLBM* with $n = p = 100$ according to the strength of the spatial and temporal dependencies.

5. Co-clustering of environmental data from Région Sud. This section presents the application of the methodology introduced above to the analysis of the multivariate spatio-temporal functional data set of pollution in the South of France.

5.1. *Data and and pretreatments.* The environmental database is composed of five variables, respectively three pollutants (maximum of NO_2 and O_3 and average of PM10) and two meteorological measurements (average of temperature (T) and pressure (P)). On the spatial point of view, these measurements are collected at a resolution of 4 km: the region is divided in 4km squares, each square represents a measure, constituting a grid of size 1995 for pollution and of size 1967 for meteorological variables (Figure 8A-B). Unfortunately, the two grids do not overlap, thus we needed to set up a procedure to merge the two databases.

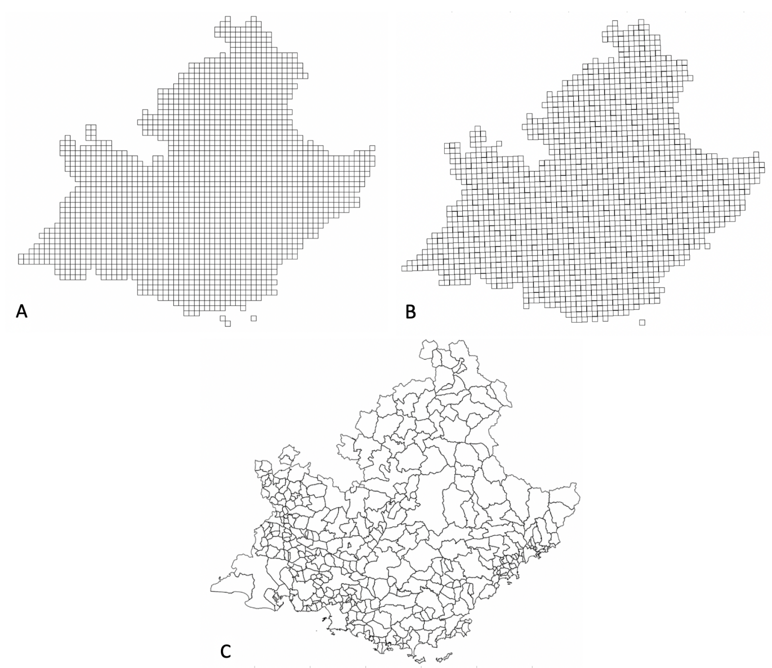

Fig 8: Non-overlapping grids of 4km - squares for pollution (A) and meteorological (B) data. (C) Région Sud divided in 357 areas of non-overlapping postcodes.

Firstly, it was necessary to manage the missing data from the weather and pollution databases: there were 2 % of days missing for the pollution data and 1 % for the weather data over 2191 days between 2013 and 2018. Out of scale measurements due to captures failures or modeling errors were identified and treated as missing data. When data are missing, none of the square of the grid has a value associated to it, thus we needed to reconstruct the data for the entire region over the days that were missing. Two methods were used: interpolation and random sampling. The interpolation method was used for variables with a periodic trend or a seasonality (PM10, O_3, P, T) whereas the random sampling method was considered for other variables (NO_2). The interpolation method was done using the `na.approx()` function of the R package `zoo`. Regarding the random sampling method, the values which replaced the missing data were taken from a window of variable size around the missing data period. The window size was adapted according to the period (i.e. day/s) of missing data to replace: specifically it was equivalent to 4 days + the length of the missing period divided by 2.

In order to merge the pollution and meteorological databases constituted on the two different grids and to make data easier to interpret, we decided to change the data resolution, specifically we created 357 areas (Figure 8C). These areas represent non-overlapping postcodes. A postcode can be associated with several municipalities but can also be different for the same municipality (4 postcodes in Nice for example). Therefore, environmental data had to be transformed from the 4km squares to the area level. The association was therefore made in 2 stages, firstly an association of pollution data - areas then weather data - areas and finally pollution-weather data at the area level. Briefly, when a 4 km square covers the largest surface of an area, it is associated with it. Thus we ended up with an environment database of 357 lines, corresponding to the areas of the region, for each pollution and weather variable. The pollution and meteorological data associated to each area, represent the maximal or the average value of the squares included in the area. Accordingly, if an area covers only one square, it will assume the values associated to the single square. We chose the maximal value for NO_2, O_3 and the average value for PM10, T and P to respect the original measurement done at the grid level.

It is important to notice that the pollutants which are studied here have usually very local effects in space and time. For instance, PM10 can drastically vary in a radius of 200 meters and we can therefore consider that two areas of 4 square km are likely to have different values, in particular on two different days. This remark is also true for meteorological data since the geographical landscape of the studied region is very specific: some mountains of an altitude of 1000 meters can be found less than 5km from the sea (particularly in the neighborhood of Nice). For these reasons, the *multiFunLBM* method we proposed is well adapted to the co-clustering of the data we show here.

5.2. *Experimental protocol.* In order to test the ability of *multiFunLBM* to identify spatio-temporal profiles, we used daily measurements of our five variables, in the 357 areas of the Région Sud, for a period of six years from 2013 to 2018 for a total leading to 2191 measurements per variable and per area. Firstly, the variables were standardized in order to be compared easily (centering and scaling). They were also transformed into functional data by weeks (7 days, start from Tuesday) using a Fourier basis. We chose the week as the time window since we needed a window to divide evenly the period under investigation. There are exactly 313 weeks of 7 days in the period of interest for this study. The Fourier basis was chosen to reconstruct the functions because some variables exhibit a clear periodicity. The number of basis function was set to 7. Then, the *multiFunLBM* algorithm was applied to the environmental database. The number of clusters on the spatial and temporal dimensions (respectively, K and L) were allowed to vary in the range 2 to 10. The appropriate number of clusters was assessed according to ICL criterion (maximum) and the type of initialization used was *funFEM*.

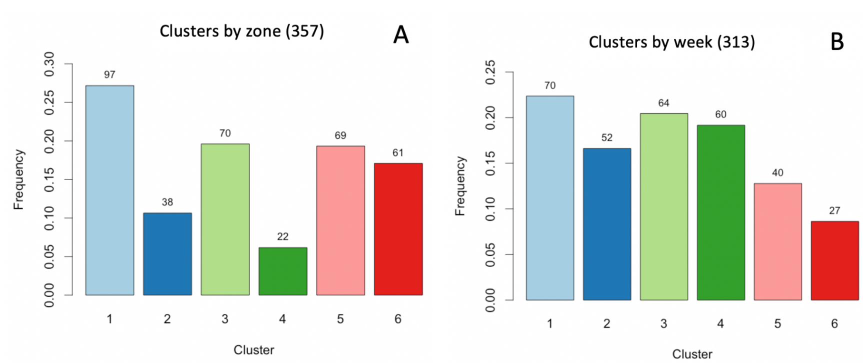

Fig 9: Frequency of the 6 clusters identified by *multiFunLBM* on the spatial (A) and temporal (B) dimension, respectively. The number of areas (A) or weeks (B) for each cluster is reported.

5.3. *Results.* The algorithm identified $K = 6$ clusters for the spatial dimension, and also $L = 6$ clusters for the temporal dimension (see Figures 9A and 9B respectively). Figure 9A shows the frequency and the number of zones (postcodes) in each cluster. The spatial distribution of the six clusters is shown in Figure 10. On overall, the clusters represent well the different areas of the region: cluster 3 groups all areas which are in the mountains, cluster 6 represents the west part of the region, which is organized along the Rhône river. The areas located on the coast are divided in two clusters: cluster 4 that groups the most populated and industrialized zones of the region including the city of Marseille and Nice, while cluster 2 gathers the remaining ones. Interestingly, all areas where the main highway of the region pass by are clustered in cluster 5. Finally, rural zones are gathered in cluster 1. This segmentation confirms that pollution levels are correlated with the geography and the human activity, and that *multiFunLBM* well identifies intrinsic structures of the territory.

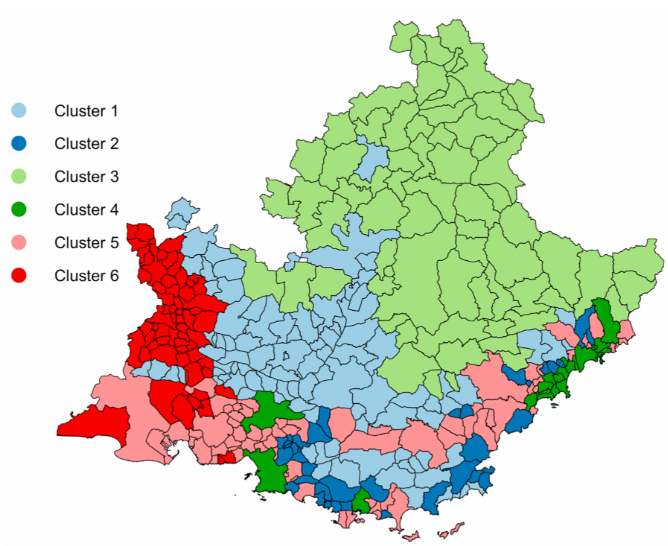

Fig 10: Association of the 357 areas in Région Sud with the clusters on the spatial dimension. Color code is indicated in the figure.

The analysis on the temporal dimension is done by weeks, meaning that the behavior of the five environmental variables is studied with a week resolution. On this dimension, six clusters were identified as well: cluster 1 is the biggest one with 70 weeks (22 %) and cluster 6 the smallest with 27 weeks (9 %), see Figure 9B. Tables 2–4 in Appendix A.7 show respectively the number of weeks by cluster by year, month and season. Figure 11 presents a calendar representation of the 313 weeks by cluster. Interestingly, the clusters mainly group the weeks accordingly to the seasons, independently of the year, proving the ability of *multiFunLBM* to find out data structures and the intrinsic link between pollution and meteorological factors due to the seasons. Specifically, cluster 2 mainly collects summer weeks (June, July, August) all over the six years, while cluster 1 and cluster 5 winter and late autumn ones (November, December, January and February). Spring and early autumn (March, April, May, September and October) weeks are in two clusters, cluster 3 and 4. Finally cluster 6 collects sporadic weeks all over the six years regardless the season, suggesting perhaps peculiar pollution trends in these weeks.

We hereafter propose a deeper analyses of the results regarding the different functional variables.

5.3.1. *Meteorological variables.* To explore the profiles of the five variables by cluster, we plot the average week profiles for each spatial and temporal cluster obtaining 36 profiles for each variable represented in Figure 12 for meteorological and Figures 13, 14 for pollution variables. For this, we used functional boxplots thanks to the `fbplot` function of the `fda` package. Each plot represents the typical behaviors by day of the week of the environmental variable measurements for all the weeks and areas contained in each cluster. For instance, the top-left panel shows for each variable the week profile for all areas contained in cluster 1 on the spatial dimension (97 zones mainly rural) on the weeks falling in cluster 1 on the temporal dimension (70 weeks, mainly winter and late autumn), and so on for the other plots.

Meteorological variables, especially average of temperatures, show flat profiles within a cluster, meaning that no week trend is observed, as expected (Figure 12). Temperature profiles perfectly reflects what is expected by season and area of the region (refer to Figure 12A): winter weeks grouped in clusters 1 and 5 on the temporal dimension (columns) are below the mean level, with the lowest reached for areas on the mountains collected in cluster 3 on the spatial dimension (rows). A similar trend but with opposite values (above the mean) is observed for temporal cluster 2, summer weeks. Conversely, clusters 3 and 4 exhibit quite opposite trends for spring and early autumn weeks, showing the coldest and the warmest profiles respectively, despite the spatial clusters.

Regarding the pressure, expected observations can be made: the highest levels are reached for spatial cluster 3 (mountains) independently by the temporal clusters. On the temporal clusters point of view, the highest levels are obtained for profiles in cluster 1 (winter and late autumn) and the lowest in cluster 3 (spring and early autumn), while summer weeks (cluster 2) show flat profiles around the average. Cluster 4 (spring and early autumn), 5 (winter and late autumn) and 6 (sporadic weeks) on the temporal dimension show profiles modulated by the day of the week, with the highest values in

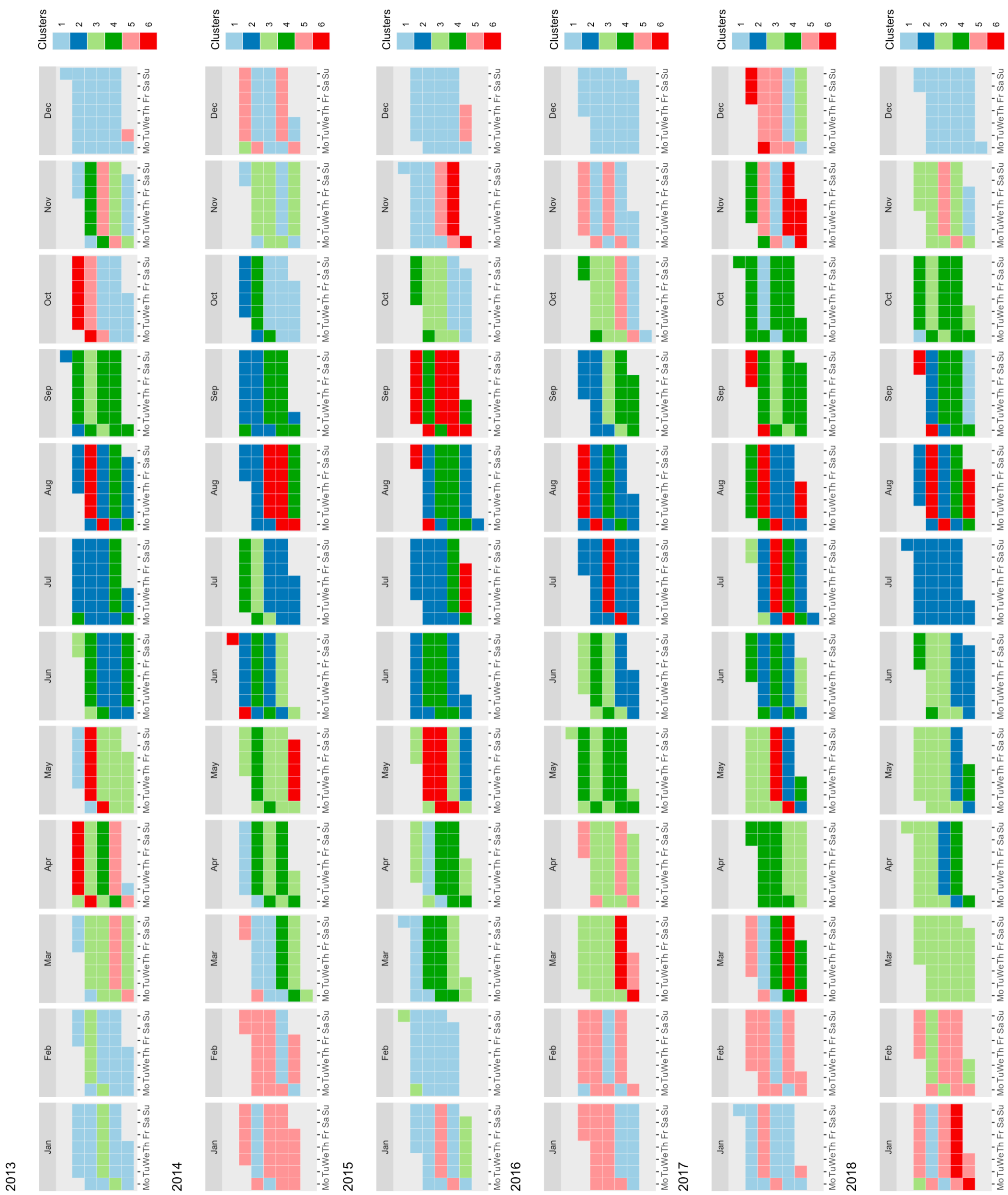

Fig 11: Association of the 313 weeks for the period from 2013 and 2018 in Région Sud with the clusters on the temporal dimension. Color code is indicated in the figure.

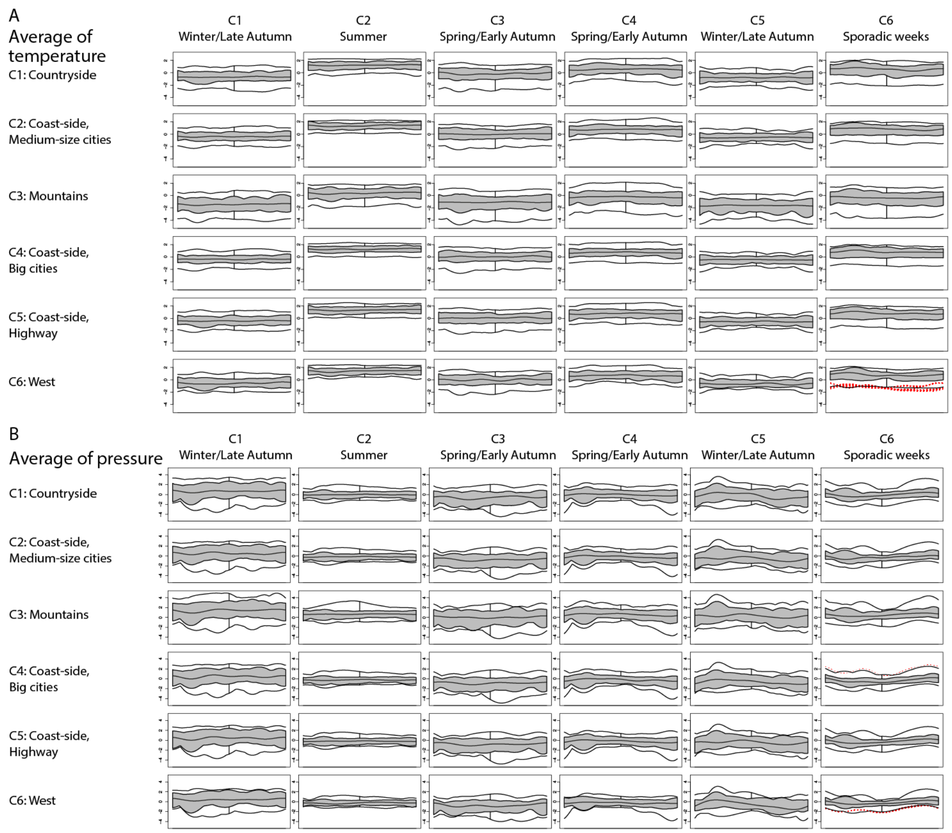

Fig 12: Functional boxplots by cluster on temporal (columns) and spatial (rows) dimensions for meteorological variables: temperature (A) and pressure (B).

the middle of the week (Wednesday, Thursday and Friday) and the lowest in the weekend plus Monday and Tuesday, for the formers, opposite pattern for the latter.

5.3.2. *Pollution variables.*

Ozone. Pollutant variable profiles mostly vary according to the day of the week. As it can be seen on Figure 13A, the most stable trends on the week window are for maximum of O_3, due to the strong dependency of this pollutant to the season more that the day of the week. Indeed, O_3 levels depend on heat and sunshine. The seasonality of this pollutant is well identified by *multiFunLBM* because profiles are similar by column (temporal dimension) and not by row (spatial dimension). The lowest levels are for temporal cluster 1 and 5 (winter and late autumn), while the highest are for cluster 2 (summer) independently of the cluster on the spatial dimension. Interestingly, as observed for meteorological variables, the two clusters of spring and early autumn weeks (cluster 3 and 4) collect weeks from the same period of the year. They do not show similar profiles intra-variables, but each cluster have similar trend inter-variable, confirming the strong relationship between O_3 and meteorological factors, mainly temperature. Overall the O_3 seems not to have a strong spatial dependence, since profiles by temporal dimensions looks similar among clusters of the zones of Région Sud. However, we observe a strong temporal dependence with the highest levels in summer.

Particulate matters. Levels of PM10 depend on intensive anthropogenic activities, such as fossil fuel combustion and biomass burning, and thus we expect a strong spatial dependence. Accordingly, *multiFunLBM* finds well that spatial cluster 5 (zones on the high-way), cluster 4 (big-cities on the coast) and cluster 6 (west), regardless of the temporal clusters, those clusters all show the highest levels, as reported in Figure 13B. Interestingly, we found high levels of PM10 in spatial cluster 1 (countryside) probably because of the biomass burning practice due to agricultural and gardening activities that produce high quantities of particles. Overall, higher levels of average of PM10 are observed mainly in wintertime (cluster 1 and 5 on the temporal dimension) due to a higher use of heating systems and fossil fuel combustion than in summer time. Clusters 1, 4, 5 and 6 on the temporal dimension show clearly the weekly dependency of this pollutant, even though with different levels: all these clusters show higher level of PM10 in the middle of the week (Tuesday, Wednesday and Thursday) and lowest in weekends and Monday, probably suggesting that industrial activities and traffic due to working days have the strongest influence on the concentrations of this pollutant.

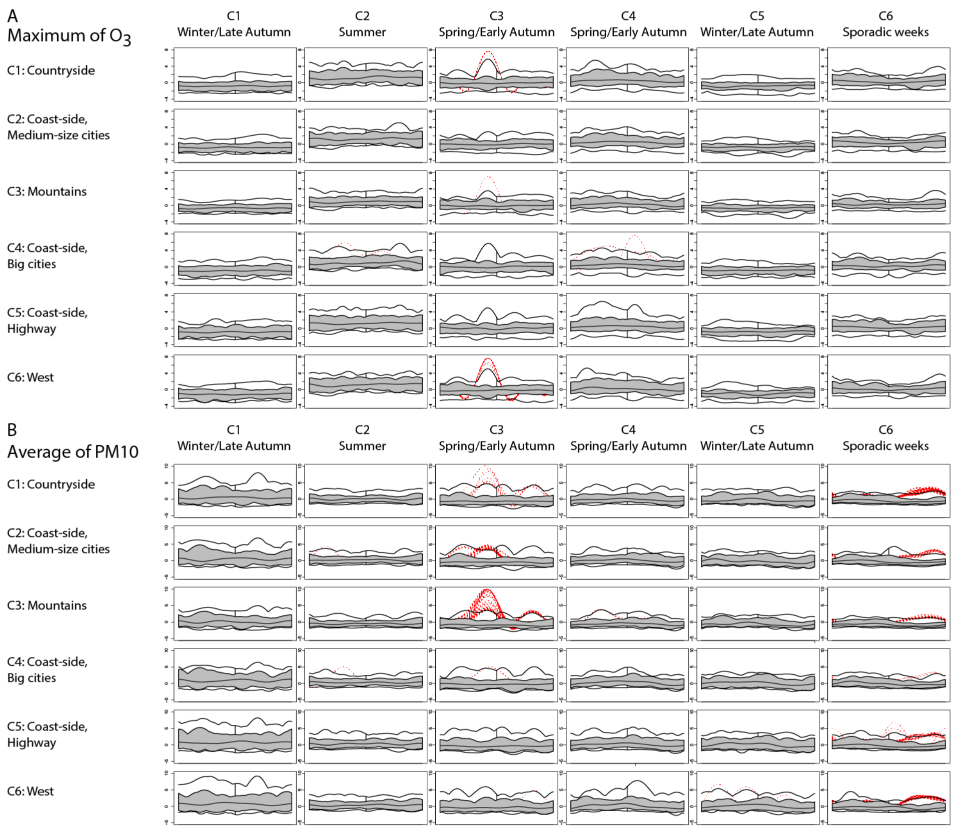

Fig 13: Functional boxplots by cluster on temporal (columns) and spatial (rows) dimensions for maximum of O_3 (A) and average of PM10 (B).

Nitrogen Dioxide. More than 50 % of the NO_2 present in the air is produced by fuel combustion and cars, and is mainly present in densely populated areas. Thus, as expected, the spatial clusters 4 (big cities on the coast) and 5 (zones on the highway) have high levels of maximum of NO_2 regardless of the season (clusters on temporal dimension), as showed in Figure 14. Temporal clusters 1 (winter and late autumn) and 2 (summer) show very similar profiles and levels on the spatial dimension, demonstrating the high spatial dependence of this pollutant that is not influenced by climatic factors due to seasonality. As observed for the other pollutants, spring and early autumn weeks of the six years under study are grouped in two clusters that show two profiles: cluster 3 has a flat profile, i.e. pollutant concentration does not depend on the day of the week, cluster 4 exhibits high levels during the week with a drop during the weekends. These two clusters show that pollution concentration in this time of the year is mostly affected by working activities, and this more than in other periods. In overall, NO_2 shows quite strong week trends, with high levels during the week and low levels in the week-ends due to public and private traffic trends accordingly with working days.

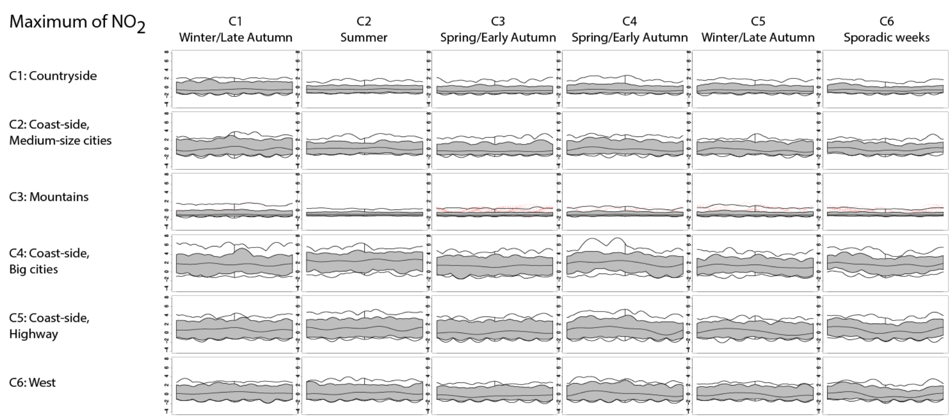

Fig 14: Functional boxplots by cluster on temporal (columns) and spatial (rows) dimensions for maximum of NO_2.

5.4. *Summary of the results.* The six clusters on the spatial dimension collect areas in the mountains, rural areas and west side of the region. Furthermore, the algorithm differentiates as well areas on the coast line depending on inhabitant proportions and industrialization levels. Surprisingly, it also identified one cluster that gathers all the areas where the main highway of the region pass by. On the other dimension, the 313 weeks of the period under investigation are gathered as well in clusters based on seasonal trends: winter and late autumn, spring and early autumn, summer. Policies to limit pollution increase are usually taken at the county level (5 in the Région Sud). However, a county represents a very heterogeneous territory with different industrialization levels, inhabitant proportions and geographical composition, making hard to find policies that suit well all the diversity of scenarios. The clusters identified by *multiFunLBM* may be used by local authorities in order to set up specific policies to lower down pollution or for public alerts when pollution raise above secure levels for the citizens and that are adapted to the different areas. Furthermore, the temporal clusters can help to spot periods of the years that are particularly affected by pollution at the level of the spatial clusters, in order to set up alerts and prevention behaviors for each specific area.

6. Discussion and conclusion. This work was prompted by the need to analyze air pollution in the South of France in order to help the AtmoSud agency and local institution to monitor pollutants dynamics and to spread public alerts when necessary. Here, we introduced a co-clustering of multivariate functional data to fulfill these objectives. The *multiFunLBM* algorithm allows to cluster both individuals (areas) and columns (weeks) simultaneously, in order to propose a summary of the data through homogenous blocks of functional data. The proposed approach relies on a functional latent block model, which assumes for each block a probabilistic distribution for the scores of the multivariate curves obtained from a multivariate functional principal component analysis. Model inference is based on a SEM-Gibbs algorithm which alternates a SE-step where row and column partitions are simulated according to Gibbs algorithm, and a M-step where model parameters are updated conditionally on the previous simulated partitions. Model selection relies on the ICL criterion which has been specifically derived for the proposed model. As far as the authors know, this is the first algorithm available for functional multivariate co-clustering. The proposed *multiFunLBM* model is available on CRAN through the `funLBM` package for R (Bouveyron et al., 2020).

The *multiFunLBM* algorithm has been used to analyze an environmental database supplied by the AtmoSud agency, collecting daily measurements of three pollutants along with pressure and temperature for a period of 6 years in the south of France. Without any knowledge about the geographical composition of the territory, nor the specific seasonality of the territory, the algorithm identifies accurate and meaningful clusters, both on the spatial and temporal dimensions. These clusters could be used in a near future to help local authorities to issue public alerts that are specific to more restricted areas.

To go further in the analysis of these data, two additional research aspects could be explored. The first would be to take into account the temporal and spatial dependencies that exist within these data. One way to do this would be to assume a smooth spatial dependence of the mixing proportions using a multinomial logistic regression as in Vandewalle et al. (2020). Recent results of Martínez-Hernández and Genton (2020) could also probably help in this task. The second one is to relax the co-clustering structure assumption and to consider bi-clustering algorithm in order to detect more subtle patterns in the data due to local and temporal specific phenomenon. An extension of the work of Orio and Vantini (2019) to the multivariate functional case could be a first solution.

On a larger dimension, the understanding of the spatio-temporal dynamic of air pollution is a current challenge worldwide. Several agencies have been created all over the world in order to monitor air pollution behavior, to identify factors influencing pollution peaks before alerting local authorities and citizens. Nevertheless, air pollution dynamic is extremely complicated and affected by many factors including meteorological variables such as temperature, pressure, wind speed, rain, humidity etc. Due to the high amount of variables to take into account, there is a tendency to focus pollution behavior studies mainly in big cities, as the main producers of pollution. However, masses of air move over bigger territories than single cities, thus spreading the pollution to adjacent areas. There is therefore a need to analyse air pollution on large territories, taking into account not only pollutants but also meteorological factors, and we believe that tools such as *multiFunLBM* may be useful in this context.

Acknowledgements. The authors would like to specially thank the AtmoSud institute (*http://atmosud.org*) for providing the data. This research has benefited from the support of the "FMJH Research Initiative Data Science for Industry". This work has also been supported by the French government, through the 3IA Côte d'Azur and UCAJEDI Investments in the Future project managed by the National Research Agency (ANR) with the reference numbers ANR-19-P3IA-0002 and ANR-15-IDEX-01.

APPENDIX A: APPENDIX

A.1. Developed form of $\boldsymbol{E}[\log(\boldsymbol{p}(\boldsymbol{c},\boldsymbol{z}^{(h+1)},\boldsymbol{w}^{(h+1)}|\boldsymbol{\theta})|\boldsymbol{\theta}^{old}]$. The conditional expectation of the complete-data log-likelihood $H(\theta|\theta^{old})$ has the following form for the model proposed in this work:

$$
\begin{aligned}
H(\theta|\theta^{old}) &\overset{(7)}{=} E[\log(p(c,z^{(h+1)},w^{(h+1)}|\theta)|\theta^{old}] \\
&\overset{(8)}{=} \sum_{i,k} z_{ik}^{(h+1)}\log\alpha_k + \sum_{j,l} w_{jl}^{(h+1)}\log\beta_l + \sum_{i,j,k,l} z_{ik}^{(h+1)}w_{jl}^{(h+1)}log(p(c_{ij};\theta_{kl})),
\end{aligned}
$$

where $z_{ik}^{(h+1)} = E[z_{ik}|\theta^{old}]$ and $w_{jl}^{(h+1)} = E[w_{jl}|\theta^{old}]$. In order to ease the reading of the reminder, the subscript ${}^{(h+1)}$ will be omitted hereafter. Let us now focus on the last quantity of the previous equation.

$$
\sum_{i,j,k,l} z_{ik}w_{jl}\log(p(c_{ij};\theta_{kl})) = \sum_{i,j,k,l} z_{ik}w_{jl}log(\frac{1}{(2\pi)^{M/2}}|\Sigma_{kl}|^{-1/2}\exp(-\frac{1}{2}(c_{ij}-\mu_{kl})^t\Sigma_{kl}^{-1}(c_{ij}-\mu_{kl}))),
$$

where $\Sigma_{kl} = \Phi^{-1/2}Q_{kl}\Delta_{kl}Q_{kl}^t\Phi^{-1/2} + \Xi_{kl}$. Let $n_{kl} = \sum_{i=1}^n\sum_{j=1}^M z_{ik}w_{jl}$ be the number of curves belonging to the block (kl), then:

$$
\begin{aligned}
\sum_{i,j,k,l} z_{ik}w_{jl}\log(p(c_{ij};\theta_{kl})) = &- \frac{1}{2}\sum_{k=1}^K\sum_{l=1}^L n_{kl}\Bigg[d_{kl}log(a_{kl}) + (M-d_{kl})log(b_{kl}) \\
&+ \frac{1}{n_{kl}}\sum_{i=1}^n\sum_{j=1}^M z_{ik}w_{jl}(c_{ij}-\mu_{kl})^t\Phi^{1/2}Q_{kl}\Delta_{kl}^{-1}Q_{kl}^t\Phi^{1/2}(c_{ij}-\mu_{kl})\Bigg] \\
&- \frac{nM}{2}log(2\pi)
\end{aligned}
$$

Since the quantity $(c_{ij}-\mu_{kl})^t\Phi^{1/2}Q_{kl}\Delta_{kl}^{-1}Q_{kl}^t\Phi^{1/2}(c_{ij}-\mu_{kl})$ is a scalar, it is equal to its trace: $tr([(c_{ij}-\mu_{kl})^t\Phi^{1/2}Q_{kl}] \times [\Delta_{kl}^{-1}Q_{kl}^t\Phi^{1/2}(c_{ij}-\mu_{kl})]) = tr([\Delta_{kl}^{-1}Q_{kl}^t\Phi^{1/2}(c_{ij}-\mu_{kl})] \times [(c_{ij}-\mu_{kl})^t\Phi^{1/2}Q_{kl}])$, consequently:

$$
\begin{aligned}
&\frac{1}{n_{kl}}\sum_{i=1}^n\sum_{j=1}^M z_{ik}w_{jl}(c_{ij}-\mu_{kl})^t\Phi^{1/2}Q_{kl}\Delta_{kl}^{-1}Q_{kl}^t\Phi^{1/2}(c_{ij}-\mu_{kl}) \\
= &\frac{1}{n_{kl}}\sum_{i=1}^n\sum_{j=1}^M z_{ik}w_{jl}tr(\Delta_{kl}^{-1}Q_{kl}^t\Phi^{1/2}(c_{ij}-\mu_{kl})(c_{ij}-\mu_{kl})^t\Phi^{1/2}Q_{kl}) \\
= &\; tr(\Delta_{kl}^{-1}Q_{kl}^t\Phi^{1/2}[\frac{1}{n_{kl}}\sum_{i=1}^n\sum_{j=1}^M z_{ik}w_{jl}(c_{ij}-\mu_{kl})^t(c_{ij}-\mu_{kl})]\Phi^{1/2}Q_{kl}) \\
= &\; tr(\Delta_{kl}^{-1}Q_{kl}^t\Phi^{1/2}\Omega_{kl}\Phi^{1/2}Q_{kl}),
\end{aligned}
$$

where $\Omega_{kl} = \frac{1}{n_{kl}} \sum_{i=1}^{n} \sum_{j=1}^{M} z_{ik} w_{jl} (c_{ij} - \mu_{kl})^t (c_{ij} - \mu_{kl})$ is the empirical covariance matrix of the curves of the block (kl). Since the matrix Δ_{kl} is diagonal, so we can write:

$$\frac{1}{n_{kl}} \sum_{i=1}^{n} \sum_{j=1}^{M} z_{ik} w_{jl} (c_{ij} - \mu_{kl})^t Q_{kl} \Delta_{kl}^{-1} Q_{kl}^t (c_{ij} - \mu_{kl}) = \sum_{j=1}^{d_{kl}} \frac{q_{klj}^t \Phi^{1/2} \Omega_{kl} \Phi^{1/2} q_{klj}}{a_{klj}} + \sum_{j=d_{kl}+1}^{M} \frac{q_{klj}^t \Phi^{1/2} \Omega_{kl} \Phi^{1/2} q_{klj}}{b_{kl}},$$

where q_{klj} is jth column of Q_{kl}.

Finally,

$$\begin{aligned} H(\theta|\theta^{old}) &= \sum_{i,k} z_{ik}^{(h+1)} \log \alpha_k + \sum_{j,l} w_{jl}^{(h+1)} \log \beta_l \\ &- \frac{1}{2} \sum_{k,l} n_{kl} \Bigg[d_{kl} log(a_{kl}) + (M - d_{kl}) log(b_{kl}) \\ &+ \sum_{j=1}^{d_{kl}} \frac{q_{klj}^t \Phi^{1/2} \Omega_{kl} \Phi^{1/2} q_{klj}}{a_{klj}} + \sum_{j=d_{kl}+1}^{M} \frac{q_{klj}^t \Phi^{1/2} \Omega_{kl} \Phi^{1/2} q_{klj}}{b_{kl}} \Bigg] - \frac{nM}{2} log(2\pi) \end{aligned}$$

A.2. Parameter $\boldsymbol{Q_{kl}}$ update. We aim to maximize $H(\theta|\theta^{old})$ under the constraint $q_{klj}^t q_{klj} = 1$, with q_{klj} the jth column of Q_{kl}. This is equivalent to look for a saddle point of the Lagrange function:

$$\mathcal{L} = -2H(\theta|\theta^{old}) - \sum_{j=1}^{M} \omega_{klj} (q_{klj}^t q_{klj} - 1)$$

where ω_{klj} are Lagrange multipliers. The gradient of $\mathcal{L}$ in relation to q_{kjl} is:

$$\begin{aligned} \nabla_{q_{klj}} \mathcal{L} &= \nabla_{q_{klj}} \Big(\sum_{k=1}^{K} \sum_{l=1}^{L} n_{kl} [\sum_{j=1}^{d_{kl}} \frac{q_{klj}^t \Phi^{1/2} \Omega_{kl} \Phi^{1/2} q_{klj}}{a_{klj}} + \sum_{j=d_{kl}+1}^{M} \frac{q_{klj}^t \Phi^{1/2} \Omega_{kl} \Phi^{1/2} q_{klj}}{b_{kl}}] \\ &- \sum_{j=1}^{M} \omega_{klj} (q_{klj}^t q_{klj} - 1) \Big). \end{aligned}$$

As a reminder, when W is symmetric, then $\frac{\partial}{\partial x}(x-s)^T W (x-s) = 2W(x-s)$ and $\frac{\partial}{\partial x}(x^T x) = 2x$ (cf. Petersen and Pedersen (2012)), so:

$$\nabla_{q_{klj}} \mathcal{L} = n_{kl} [2 \frac{\Phi^{1/2} \Omega_{kl} \Phi^{1/2}}{\sigma_{klj}} q_{klj}] - 2\omega_{klj} q_{klj}$$

where σ_{klj} is the j-th diagonal term of matrix Δ_k (a_{klj} for $j \leq d_{kl}$ and b_{kl} otherwise).
Thus,

$$\nabla_{q_{klj}}\mathcal{L} = 0 \quad \Leftrightarrow \quad \Phi^{1/2}\Omega_{kl}\Phi^{1/2}q_{klj} = \frac{\omega_{klj}\sigma_{klj}}{n_{kl}}q_{klj}.$$

q_{klj} is the eigenfunction of $\Phi^{1/2}\Omega_{kl}\Phi^{1/2}$ which match the eigenvalue $\lambda_{klj} = \frac{\omega_{klj}\sigma_{klj}}{n_{kl}} = q_{klj}^t\Phi^{1/2}\Omega_{kl}\Phi^{1/2}q_{klj}$. Since the vectors q_{klj} are eigenvectors of Ω_{kl}, we have $q_{klj}^t q_{klm} = 0$ if $j \neq m$. Reporting the value of λ_{klj} in $H(\theta|\theta^{old})$ allows to see that maximizing $H(\theta|\theta^{old})$ regarding q_{klj} is equivalent to minimize the quantity $\sum_{k=1}^{K}\sum_{l=1}^{L} n_{kl}\sum_{j=1}^{d_{kl}}\lambda_{klj}(\frac{1}{a_{kl}} - \frac{1}{b_{kl}})$ regarding to λ_{klj}. Knowing that $(\frac{1}{a_{kl}} - \frac{1}{b_{kl}}) \leq 0$, λ_{kl} has to be as high as possible. therefore, the j-th column q_{klj} of matrix Q_{kl} is estimated by the eigenfunction associated to the j-th highest eigenvalue of $\Phi^{1/2}\Omega_{kl}\Phi^{1/2}$.

A.3. Parameter a_{kl} update.

Partial derivative of $H(\theta|\theta^{old})$ according to a_{kl} correspond to:

$$\begin{aligned}
\frac{\partial H(\theta|\theta^{old})}{\partial a_{kl}} &= -\frac{1}{2}n_{kl}(\frac{d_{kl}}{a_{kl}} - \sum_{j=1}^{d_{kl}}\frac{q_{kjl}^t\Phi^{1/2}\Omega_{kl}\Phi^{1/2}q_{kjl}}{a_{kl}^2}) \\
&= -\frac{1}{2}n_{kl}(\frac{d_{kl}}{a_{kl}} - \sum_{j=1}^{d_{kl}}\frac{\lambda_{klj}}{a_{kl}^2})
\end{aligned}$$

The prerequisite $\frac{\partial H(\theta|\theta^{old})}{\partial a_{kl}} = 0$ implies:

$$\begin{aligned}
\frac{n_{kl}d_{kl}}{a_{kl}} &= \frac{n_{kl}}{a_{kl}^2}\sum_{j=1}^{d_{kl}}\lambda_{klj} \\
\Leftrightarrow \quad a_{kl} &= \frac{1}{d_{kl}}\sum_{j=1}^{d_{kl}}\lambda_{klj}
\end{aligned}$$

with λ_{kl} the eigenvalues of block kl.

A.4. Parameter b_{kl} update.

Partial derivative of $H(\theta|\theta^{old})$ according to b_{kl} correspond to:

$$\frac{\partial H(\theta|\theta^{old})}{\partial b_{kl}} = -\frac{1}{2}[\frac{M - d_{kl}}{b_{kl}} - \sum_{j=d_{kl}+1}^{M}\frac{q_{klj}^t\Phi^{1/2}\Omega_{kl}\Phi^{1/2}q_{klj}}{b_{kl}^2}]$$

The prerequisite $\frac{\partial H(\theta|\theta^{old})}{\partial b_{kl}} = 0$ implies:

$$\frac{M - d_{kl}}{b_{kl}} = \sum_{j=d_{k+1}}^{M} \frac{q_{klj}^t \Phi^{1/2} \Omega_{kl} \Phi^{1/2} q_{klj}}{b_k^2}$$

$$\Leftrightarrow \quad b_{kl} = \frac{1}{M - d_{kl}} [tr(\Phi^{1/2} \Omega_{kl} \Phi^{1/2}) - \sum_{j=1}^{d_{kl}} \lambda_{klj}]$$

A.5. Robustness to noise and influence of initialization $n = p = 500$. Figure 15 plots the ARI results for a sample size of $n = p = 500$ corresponding to the simulation setting of Section 4.2.

A.6. Model selection results for $n = p = 100$. Table 2 contains the model selection results for a sample size of $n = p = 100$ corresponding to the simulation setting of Section 4.3.

TABLE 2
Percentage of selection of each model (K, L) by ICL among the 20 simulated data sets, with $n = p = 100$. The actual values for (K, L) are $(4, 3)$.

Scenario $\tau = 0$						Scenario $\tau = 0.3$						Scenario $\tau = 0.5$					
K/L	2	3	4	5	6	K/L	2	3	4	5	6	K/L	2	3	4	5	6
2	0	0	0	0	0	2	5	95	0	0	0	2	100	0	0	0	0
3	0	100	0	0	0	3	0	0	0	0	0	3	0	0	0	0	0
4	0	**0**	0	0	0	4	0	**0**	0	0	0	4	0	**0**	0	0	0
5	0	0	0	0	0	5	0	0	0	0	0	5	0	0	0	0	0
6	0	0	0	0	0	6	0	0	0	0	0	6	0	0	0	0	0

TABLE 3
Number of weeks by year by cluster.

Cluster/Year	2013	2014	2015	2016	2017	2018
1	16	12	17	11	9	7
2	8	9	8	9	7	11
3	11	10	7	12	7	18
4	9	10	11	7	15	8
5	5	9	3	11	8	6
6	4	3	7	3	7	3

A.7. Distribution of weeks by cluster.

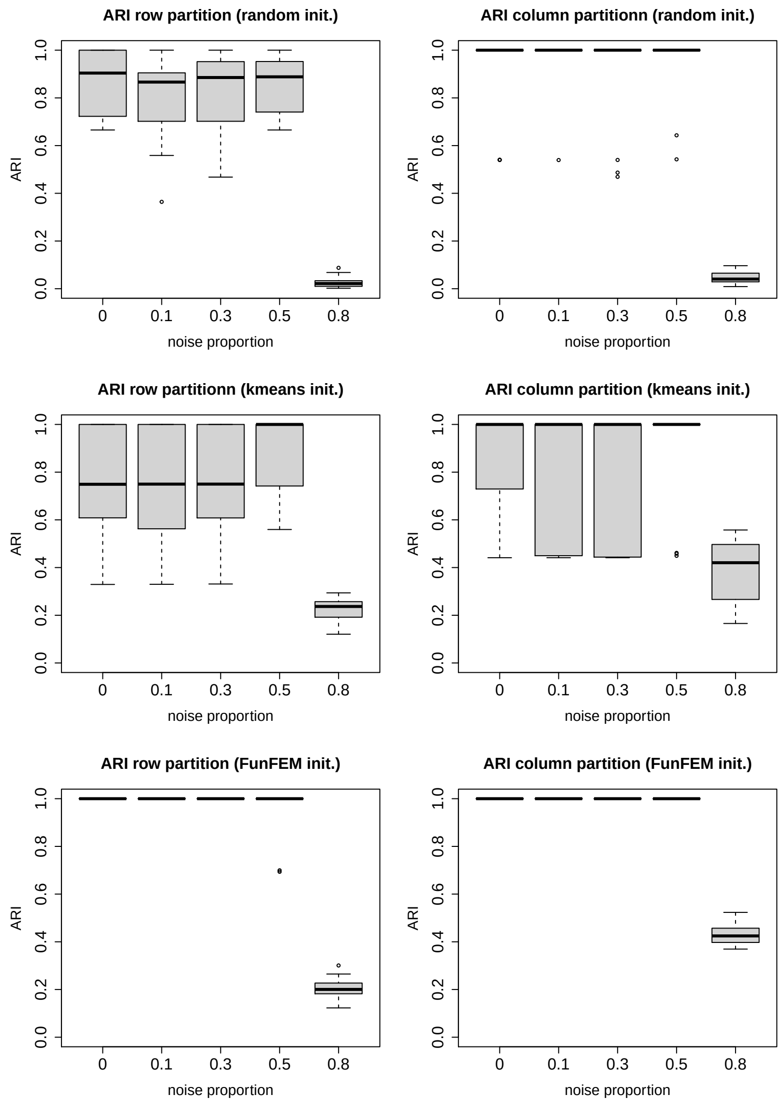

Fig 15: ARI results for *multiFunLBM* with $n = p = 500$ according to the noise ratio and depending on the type of initialization: *random* (top), *k-means* (middle) and *funFEM* (bottom).

REFERENCES

(2013). Review of evidence on health aspects of air pollution - REVIHAAP Project. Technical report, WHO Regional Office for Europe, Copenhagen, Denmark.

(2016). *Outdoor air pollution*, Volume 109 of *IARC Monogr Eval Carcinog Risks Hum*.

TABLE 4
Number of weeks by month by cluster.

Cluster/Month	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1	15	12	6	3	1	0	0	0	1	11	12	23
2	0	0	0	1	3	14	22	15	7	1	0	0
3	3	4	14	14	17	7	2	0	3	6	7	2
4	0	0	5	9	6	9	5	7	15	11	2	0
5	13	14	4	3	0	0	0	0	0	2	6	6
6	1	0	2	1	5	1	3	9	5	1	3	1

TABLE 5
Number of weeks by season by cluster.

Cluster/Season	Winter	Spring	Summer	Autumn
1	38	3	0	35
2	0	11	43	1
3	13	34	7	12
4	4	24	22	19
5	26	4	0	11
6	1	8	13	5

IARC.

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control 9*, 716–723.

Banerjee, A., I. Dhillon, J. Ghosh, S. Merugu, and D. S. Modha (2007). A generalized maximum entropy approach to bregman co-clustering and matrix approximation. *Journal of Machine Learning Research 8*(Aug), 1919–1986.

Ben Slimen, Y., S. Allio, and J. Jacques (2018). Model-Based Co-clustering for Functional Data. *Neurocomputing 291*, 97–108.

Benbrahim-Tallaa, L., R. Baan, Y. Grosse, B. Lauby-Secretan, F. El Ghissassi, and V. e. a. Bouvard (2012). Carcinogenicity of diesel-engine and gasoline-engine exhausts and some nitroarenes. *The Lancet Oncology 13*, 663–664.

Bhatia, P., S. Iovleff, and G. Govaert (2014, December). blockcluster: An R Package for Model Based Co-Clustering. working paper or preprint.

Biernacki, C., G. Celeux, and G. Govaert (2000). Assessing a mixture model for clustering with the integrated completed likelihood. *IEEE Trans. PAMI 22*, 719–725.

Bouveyron, C., L. Bozzi, J. Jacques, and F.-X. Jollois (2018). The functional latent block model for the co-clustering of electricity consumption curves. *Journal of the Royal Statistical Society: Series C Applied Statistics 67*(4), 897–915.

Bouveyron, C., G. Celeux, T. B. Murphy, and A. E. Raftery (2019). *Model-Based Clustering and Classification for Data Science: With Applications in R*. Cambridge University Press.

Bouveyron, C., L. Cheze, J. Jacques, P. Martin, and A. Schmutz (2020). Clustering multivariate functional data in group-specic functional subspaces. *Computational Statistics*, in press.

Bouveyron, C., E. Come, and J. Jacques (2015). The discriminative functional mixture model for the analysis of bike sharing systems. *Annals of Applied Statistics 9*(4), 1726–1760.

Bouveyron, C., E. Côme, and J. Jacques (2015). The discriminative functional mixture model for a comparative analysis of bike sharing systems. *Annals of Applied Statistics 9*(4), 1726–1760.

Bouveyron, C., J. Jacques, and A. Schmutz (2020). *funLBM: Model-Based Co-Clustering of Functional Data.* R package version 2.1.

Chamroukhi, F. and C. Biernacki (2017, July). Model-Based Co-Clustering of Multivariate Functional Data. In *ISI 2017 - 61st World Statistics Congress*, Marrakech, Morocco.

Corneli, M., C. Bouveyron, and P. Latouche (2019, January). Co-Clustering of ordinal data via latent continuous random variables and a classification EM algorithm. working paper or preprint.

Delaigle, A. and P. Hall (2010). Defining probability density for a distribution of random functions. *The Annals of Statistics 38*, 1171–1193.

Dempster, A. P., N. M. Laird, and D. B. Rubin (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B. Methodological 39*(1), 1–38.

George, T. and S. Merugu (2005). A scalable collaborative filtering framework based on co-clustering. In *Data Mining, Fifth IEEE international conference on*, pp. 4–pp. IEEE.

Govaert, G. and M. Nadif (2013). *Co-Clustering* (1st ed.). Wiley-IEEE Press.

Hamra, G., N. Guha, A. Cohen, F. Laden, O. Raaschou-Nielsen, J. Samet, and et al. (2014). Outdoor particulate matter exposure and lung cancer: a systematic review and meta-analysis. *Environ Health Perspectives 112*, 906–911.

Ieva, F., A. Paganoni, D. Pigoli, and V. Vitelli (2013). Multivariate Functional Clustering for the Morphological Analysis of ECG Curves. *Journal of the Royal Statistical Society, Series C (Applied Statistics) 62*(3), 401–418.

Jacques, J. and C. Biernacki (2018, July). Model-Based Co-clustering for Ordinal Data. *Computational Statistics and Data Analysis 123*, 101–115.

Jacques, J. and C. Preda (2013). Funclust: a curves clustering method using functional random variable density approximation. *Neurocomputing 112*, 164–171.

Jacques, J. and C. Preda (2014a). Functional data clustering: a survey. *Advances in Data Analysis and Classification 8*(3), 231–255.

Jacques, J. and C. Preda (2014b). Model based clustering for multivariate functional data. *Computational Statistics and Data Analysis 71*, 92–106.

Kayano, M., K. Dozono, and S. Konishi (2010). Functional Cluster Analysis via Orthonormalized Gaussian Basis Expansions and Its Application. *Journal of Classification 27*, 211–230.

Keribin, C., V. Brault, G. Celeux, and G. Govaert (2015). Estimation and selection for the latent block model on categorical data. *Statistics and Computing 25*(6), 1201–1216.

Keribin, C., G. Govaert, and G. Celeux (2010). Estimation d'un modèle à blocs latents par l'algorithme SEM. In *42èmes Journées de Statistique*, Marseille, France, France.

Laclau, C., I. Redko, B. Matei, Y. Bennani, and V. Brault (2017, August). Co-clustering through Optimal Transport. In *34th International Conference on Machine Learning*, Volume 70 of *Proceedings of the 34th International Conference on Machine Learning*, Sydney, Australia, pp. 1955–1964. Proceedings of Machine Learning Research.

Lelieveld, J., J. Evans, and M. e. a. Fnais (2015). The contribution of outdoor air pollution sources to premature mortality on a global scale. *Nature 525*, 367–371.

Martínez-Hernández, I. and M. G. Genton (2020, May). Recent developments in complex and spatially correlated functional data. *Brazilian Journal of Probability and Statis-*

tics 34(2), 204–229.

Menut, L., B. Bessagnet, D. Khvorostyanov, M. Beekmann, N. Blond, A. Colette, I. Coll, G. Curci, G. Foret, A. Hodzic, S. Mailler, F. Meleux, J.-L. Monge, I. Pison, G. Siour, S. Turquety, M. Valari, R. Vautard, and M. G. Vivanco (2013). Chimere 2013: a model for regional atmospheric composition modelling. *Geosci. Model Dev. 6*, 981–1028.

Nadif, M. and G. Govaert (2008). Algorithms for model-based block gaussian clustering. In *Proceedings of The 2008 International Conference on Data Mining, DMIN 2008, July 14-17, 2008, Las Vegas, USA, 2 Volumes*, pp. 536–542.

Orio, J. D. and S. Vantini (2019). *funBI: a Biclustering Algorithm for Functional Data.*

Pascal, M., P. de Crouy Chanel, V. Wagner, M. Corso, C. Tillier, M. Bentayeb, M. Blanchard, A. Cochet, L. Pascal, S. Host, S. Goria, A. Le Tertre, E. Chatignoux, A. Ung, P. Beaudeau, and S. Medina (2016). The mortality impacts of fine particles in france. *Science of the Total Environment 571*, 416–425.

Petersen, K. B. and M. S. Pedersen (2012, nov). The matrix cookbook. Version 20121115.

Ramsay, J. O. and B. W. Silverman (2005). *Functional data analysis* (Second ed.). Springer Series in Statistics. New York: Springer.

Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association 66*(336), 846–850.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics 6*(2), 461–464.

Selosse, M., J. Jacques, and C. Biernacki (2021). Model-based co-clustering for mixed type data. *Computational Statistics and data analysis 144*.

Tokushige, S., H. Yadohisa, and K. Inada (2007). Crisp and fuzzy k-means clustering algorithms for multivariate functional data. *Computational Statistics 22*, 1–16.

Vandewalle, V., C. Preda, and S. Dabo-Niang (2020). Clustering spatial functional data. In J. Mateu and R. Giraldo (Eds.), *Geostatistical Functional Data Analysis : Theory and Methods.* Chichester, UK: John Wiley and Sons.

Wang, S. and A. Huang (2017). Penalized nonnegative matrix tri-factorization for co-clustering. *Expert Systems with Applications 78*, 64–73.