



# Segmental dynamic factor analysis for time series of curves

Allou Samé<sup>1</sup> · Gérard Govaert<sup>2</sup>

Received: 27 September 2015 / Accepted: 21 September 2016 / Published online: 3 October 2016  
© Springer Science+Business Media New York 2016

**Abstract** A new approach is introduced in this article for describing and visualizing time series of curves, where each curve has the particularity of being subject to changes in regime. For this purpose, the curves are represented by a regression model including a latent segmentation, and their temporal evolution is modeled through a Gaussian random walk over low-dimensional factors of the regression coefficients. The resulting model is nothing else than a particular state-space model involving discrete and continuous latent variables, whose parameters are estimated across a sequence of curves through a dedicated variational Expectation-Maximization algorithm. The experimental study conducted on simulated data and real time series of curves has shown encouraging results in terms of visualization of their temporal evolution and forecasting.

**Keywords** Functional time series · Visualization and forecasting · Mixture of regressions · Dynamic factor analysis · Variational EM · Condition monitoring

## 1 Introduction

Functional data or curve-valued data are becoming more and more frequent in many applicative fields including demography (Hyndman and Ullah 2007), electricity (Ferraty and Vieu

2006), finance (Diebold and Li 2006; Hays et al. 2012), meteorology (Besse et al. 2000) to cite just a few examples. More particularly, this paper is concerned with the issue of dimension reduction and visualization of time series of curves. The application which motivated this study concerns the monitoring of the railway switches which enable trains to change tracks at junctions. This task is accomplished by analyzing specific times series of curves, each curve being the electric power consumed during a switch operation whose duration is about 5 s. In this context, two temporal aspects have to be taken into account in the data representation: the time within curves due to their functional nature and the time between curves which stipulates that the curves are ordered.

Basically, since a time series of curves can be viewed as a set of multivariate observations, it can be analyzed through classical multivariate methods such as principal component analysis (PCA) (Pearson 1901; Jolliffe 2002), factor analysis (FA) (Lawley 1940), or mixture of factor analyzers (McLachlan et al. 2003; Baek et al. 2010). But, even if these methods are able to represent multivariate data into a fewer dimensional space, they are not designed to capture the temporal aspects evoked previously. In particular, the same representation is obtained even after changing the order of time of the data, which may make interpretation difficult. One can use dynamic factor analysis (Molaneaar 1985; Zuur et al. 2003), whose goal is to extract from a multivariate time series a lower dimensional time series. Compared to PCA and FA, it constitutes a better alternative for the curves temporal order to be considered into the representation. Another alternative is the functional principal component analysis (Ramsay and Silverman 1997; Ferraty and Vieu 2006), which treats the data not just as multivariate observations, but as the discretized values of smooth functions, and allows the time within each curve to be taken into account. An extension of functional principal component analysis has been developed

✉ Allou Samé  
allou.same@ifsttar.fr

Gérard Govaert  
gerard.govaert@hds.utc.fr

<sup>1</sup> Université Paris-Est, IFSTTAR, COSYS, GRETTIA,  
77447 Marne-la-Vallée, France

<sup>2</sup> Laboratoire Heudiasyc, UMR CNRS 7253, Université de  
Technologie de Compiègne, Compiègne, France

by Gareth et al. (2000) to deal with curves measured at irregular and sparse sets of time points which can differ across curves. But again, even if the temporal aspect within curves is taken into account, this is not the case of the order between curves. To forecast demographic curves, Hyndman and Ullah (2007) have adopted a two-step approach which consists in first decomposing the curves into a spline basis function expansion and then fitting univariate time series models such as autoregressive moving-average (ARMA) models to each of the factors. Hays et al. (2012) have recently introduced, within the context of yield curve forecasting, an approach merging these two steps into a single model which consists in a dynamic factor analysis model with curve-valued factor loadings.

For our purposes, as the power consumption curves to be analyzed are nonlinear and possibly made of several regimes, a mixture of regression models, which uses logistic functions to manage the transitions between regimes (Chamroukhi et al. 2009b) is exploited. The proposed dynamic factor model then consists in allowing the regression coefficients of the latter model to evolve across the curves through dynamic latent factors. This model can be formulated as a specific state-space model (Shumway and Stoffer 2011; Kalman 1960) involving discrete latent variables (to represent the regimes within curves) and continuous latent variables (to represent the dynamic across the curves). This latent variable context enables parameter estimation via the Expectation-Maximization (EM) algorithm within the framework of variational approximations (Jordan et al. 1999; Beal 2003). Although this paper focused on specific time series of curves, the developed methodology is quite general and can be applied to other functional time series.

The paper is organized as follows. Section 2 briefly recalls the dynamic factor analysis model and Sect. 3 gives an overview of the specific latent-segmentation regression model used for modeling our power consumption curves. Then, Sect. 4 introduces the proposed dynamic factor analytic model for time series of curves and its parameter estimation via a variational EM algorithm. In Sect. 5, the proposed methodology is used to analyze both sequences of synthetic curves and real-world sequences including power consumption curves acquired from successive switch operations on the French railway network.

Throughout this paper, the following notations will be used. An observed time series of curves will be denoted by  $(\mathbf{x}_1, \dots, \mathbf{x}_T)$ , where  $\forall t = 1, \dots, T$ ,  $\mathbf{x}_t = (x_{t1}, \dots, x_{tS})' \in \mathbb{R}^S$  consists of  $S$  real values observed over a regularly spaced time grid indexed by the integers  $s = 1, \dots, S$ . The segments specific to the proposed model will be indexed by  $k = 1, \dots, K$ , where  $K$  is the number of segments. To simplify the notations, the sums and products relative to subscripts  $t$ ,  $s$ , and  $k$  will be written without indicating their limits of variations. For instance,  $\sum_{t,s}$  will refer to

$\sum_{t=1}^T \sum_{s=1}^S$ ,  $\sum_{t,s,k}$  will refer to  $\sum_{t=1}^T \sum_{s=1}^S \sum_{k=1}^K$ ,  $\sum_k$  will refer to  $\sum_{k=1}^K$ ,  $\prod_{t,s}$  will refer to  $\prod_{t=1}^T \prod_{s=1}^S$ , and  $\prod_t$  will refer to  $\prod_{t=1}^T$ . Using the same convention, the abbreviations  $\forall t, s, k$ ,  $\forall t, s$ ,  $\forall s$ , and  $\forall k$  will also be used. The transpose of a matrix  $M$  will be denoted as  $M'$ .

## 2 Review of dynamic factor analysis

Before addressing the methods specific to functional data, this section briefly reviews dynamic factor analysis (DFA) also known as common trend analysis, which has its origins in both the fields of economics (Geweke 1977; Harvey 1989) and psychology (Molaneaar 1985).

The dynamic factor analysis model supposes that the observed  $S$ -dimensional time series  $(\mathbf{x}_t)_{t=1, \dots, T}$  depends linearly on a  $q$ -dimensional ( $q \ll S$ ) series of factors  $(\mathbf{f}_t)_{t=0, \dots, T}$  distributed according to a Gaussian random walk. It is defined by

$$\mathbf{x}_t = \mathbf{A} \mathbf{f}_t + \mathbf{b} + \boldsymbol{\varepsilon}_t \quad \boldsymbol{\varepsilon}_t \sim \mathcal{N}(0, \mathbf{W}) \quad \forall t, \quad (1)$$

$$\mathbf{f}_t = \mathbf{f}_{t-1} + \boldsymbol{\eta}_t \quad \boldsymbol{\eta}_t \sim \mathcal{N}(0, \mathbf{V}) \quad \forall t, \quad (2)$$

where  $\mathbf{A}$  is the  $S \times q$  factor loadings matrix and  $\mathbf{b}$  is a  $S \times 1$  level vector. The random vectors  $\boldsymbol{\varepsilon}_t \in \mathbb{R}^S$  and  $\boldsymbol{\eta}_t \in \mathbb{R}^q$  are independent Gaussian noises with zero mean and respective covariance matrices  $\mathbf{W}$  and  $\mathbf{V}$ . The model also assumes that the initial factor  $\mathbf{f}_0$  is distributed, independently from  $\boldsymbol{\varepsilon}_t$  and  $\boldsymbol{\eta}_t$ , according to the density  $\mathcal{N}(\boldsymbol{\mu}_0, \mathbf{V}_0)$ . From this formulation, the dynamic factor analysis model can be seen as a generalization of the classical factor analysis model, where the factors are supposed to evolve over time. This model is also part of the more general framework of state-space models (Shumway and Stoffer 2011), the particularity of dynamic factor analysis being the assumption that states have a lower dimension than observations.

For this model to be identifiable, the following restrictions, which impose  $q^2 + q$  constraints on the parameters, are generally suggested (Harvey 1989; Bai and Wang 2015): (i) matrix  $\mathbf{A} = (A_{ij})$  satisfies  $A_{ij} = 0$  if  $j > i$  and the diagonal elements of its first  $q \times q$  block are strictly positive; (ii)  $\mathbf{V} = \mathbf{I}_q$  where  $\mathbf{I}_q$  is the identity matrix in  $\mathbb{R}^q$ ; and (iii) the first  $q$  elements of  $\mathbf{b}$  are set to 0.

Direct optimization techniques were originally used to estimate the set of parameters  $\boldsymbol{\theta} = \{\boldsymbol{\mu}_0, \mathbf{V}_0, \mathbf{A}, \mathbf{b}, \mathbf{W}\}$  of the dynamic factor model (Molaneaar 1985). However, due to the latent variables context of DFA, the Expectation-Maximization (EM) algorithm (Dempster et al. 1977; McLachlan and Krishnan 2008) provides an efficient framework for parameter estimation. Inspired by the general formulation introduced by Shumway and Stofer (1982) for state-space models, Zuur et al. (2003) have proposed an

Expectation-Maximization algorithm to estimate the parameter  $\theta$  maximizing the log-likelihood in the framework of a marine environmental application. Once the parameters have been estimated, the latent factors are recovered using the expectation:

$$\hat{f}_t = \mathbb{E}[f_t | \mathbf{x}_1, \dots, \mathbf{x}_T; \theta] \quad \forall t, \quad (3)$$

computed via Kalman filtering and smoothing recursions (Shumway and Stofer 1982; Zuur et al. 2003).

### 3 Regressive modeling of curves with changes in regime

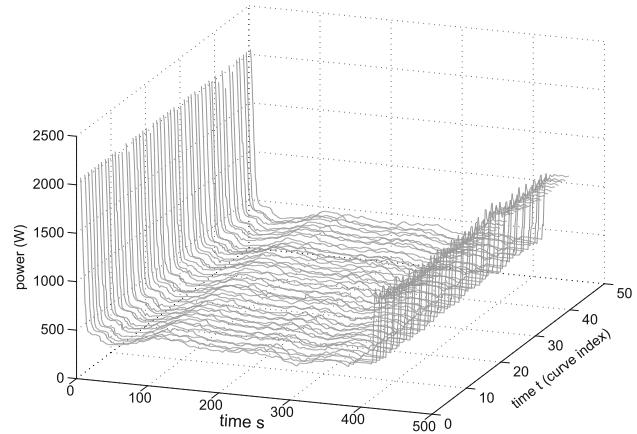
This section is concerned with the issue of modeling individual curves that may be subject to changes in regime, a problem which is also known as time series segmentation or change-point detection. We tackle this problem using a polynomial regression mixture model with time-varying proportions. This model, so-called “regression model with hidden logistic process” (RHLP), was originally introduced in (Chamroukhi et al. 2009b) within a maximum likelihood framework. It may be noted that the change-point detection problem can efficiently be tackled using Gaussian processes as models describing the temporal dependence within segments, and Bayesian approaches for parameters and change-point estimation (Fearnhead and Liu 2007; Saatçi et al. 2010; Garnett et al. 2010). Although the latter approach has not been adopted in this article, it remains an efficient alternative for online time series segmentation.

We start by describing the curves with changes in regime specific to railway switch operations. Then, the RHLP model initiated in (Chamroukhi et al. 2009b) is succinctly recalled. Finally, its matrix formulation is highlighted with a view to introducing our dynamic factor analytic model.

#### 3.1 Power consumption curve sequences acquired from switch operations

As mentioned in the introduction, the main motivation behind this study was the monitoring of the railway switches that allow trains to change tracks at junctions. A switch operation consists in moving laterally some linked tapering rails (also known as points) into one of two positions. In the case of the French high-speed lines, this operation is generally operated by an electrical motor (380 V—alternative current).

The monitoring task is performed by temporally analyzing sequences of electrical power consumption curves acquired during switch operations. Each curve is sampled at 100 Hz (100 points recorded per second) and observed over about 5 s (about 500 points per curve). Figure 1 shows a sequence



**Fig. 1** Sequence of 50 power consumption curves acquired during successive switch operations

of 50 power consumption curves acquired during successive switch operations.

The specificity of the curves to be analyzed in this context is that they are subject to various changes in regime as a result of five successive mechanical movements of the physical components associated with the switch mechanism: motor activation, points unlocking, points translation, points locking, and friction (phase where an additional effort is applied to ensure the locking).

The next section gives a general description of the regression model with hidden segmentation, originally introduced in (Chamroukhi et al. 2009b), for the modeling of a single curve  $\mathbf{x}_t (t = 1, \dots, T)$  subject to changes in regime.

#### 3.2 Definition of the RHLP model

The regression model with hidden logistic process (Chamroukhi et al. 2009b), so-called RHLP, assumes that each individual observation  $x_{ts}$  of the curve  $\mathbf{x}_t$  follows one of  $K$  polynomial regression models associated to  $K$  regimes involved in the generation process of the curve. Let us denote by  $\mathbf{z}_t = (z_{t1}, \dots, z_{tS})'$  the hidden random process associated to  $\mathbf{x}_t$ , with  $z_{ts} \in \{1, \dots, K\}$ , which specifies the assignment of the  $x_{ts}$  to the different (sub)regression models. It should be noticed that the variable  $z_{ts}$  can equivalently be represented by the vector of binary variables  $(z_{tsk})_{k=1,\dots,K}$ , where  $z_{tsk} = 1$  if  $z_{ts} = k$  and  $z_{tsk} = 0$  otherwise. The RHLP model is then defined as follows:

$$x_{ts} = \sum_k z_{tsk} (\mathbf{u}'_s \boldsymbol{\beta}_{tk} + \sigma_k \varepsilon_{ts}) \quad \forall t, s, \quad (4)$$

$$z_{ts} \sim \mathcal{M}(1; \pi_1(s, \boldsymbol{\alpha}), \dots, \pi_K(s, \boldsymbol{\alpha})) \quad \forall t, s, \quad (5)$$

where  $\varepsilon_{ts}$  is a random Gaussian noise with zero mean and unit variance,  $\mathbf{u}_s = (1, s, s^2, \dots, s^p)'$  is the vector of regressors associated to a  $p$ -th order polynomial,  $\boldsymbol{\beta}_{tk} \in \mathbb{R}^{p+1}$  and

$\sigma_k > 0$  are, respectively, the coefficient vectors and the noise standard deviation of the  $k$ -th polynomial regression model.

To segment the time series  $x_t$  into contiguous segments, which fulfills the requirements of our application, the latent variables  $z_{ts}$  are supposed to be randomly drawn according to the multinomial distribution  $\mathcal{M}(1; \pi_1(s, \alpha), \dots, \pi_K(s, \alpha))$ , where the probabilities  $\pi_k(s, \alpha)$  are defined as the logistic functions

$$\begin{aligned} \pi_k(s, \alpha) &= P(z_{tsk} = 1; \alpha) \\ &= \frac{\exp(\alpha_{k0} + \alpha_{k1}s)}{\sum_{\ell=1}^K \exp(\alpha_{\ell0} + \alpha_{\ell1}s)}, \end{aligned} \quad (6)$$

with  $\alpha = (\alpha_{k0}, \alpha_{k1})_{k=1, \dots, K} \in \mathbb{R}^{2K}$ . It should be noticed that the logistic functions verify  $0 < \pi_k(s; \alpha) < 1$  and  $\sum_{k=1}^K \pi_k(s; \alpha) = 1$ . For the logistic function parameters to be identifiable, it is supposed that  $(\alpha_{K0}, \alpha_{K1}) = (0, 0)$  (Jiang and Tanner 1999). Therefore, in the rest of the paper, it will be assumed that  $\alpha \in \mathbb{R}^{2K-2}$ .

From the model formulation, it can be shown that  $(x_{t1}, \dots, x_{tS})$  is a sample independently distributed according to the mixture density

$$P(x_{ts}; \theta_t) = \sum_k \pi_k(s; \alpha) \mathcal{N}(x_{ts}; \mathbf{u}'_s \boldsymbol{\beta}_{tk}, \sigma_k^2), \quad (7)$$

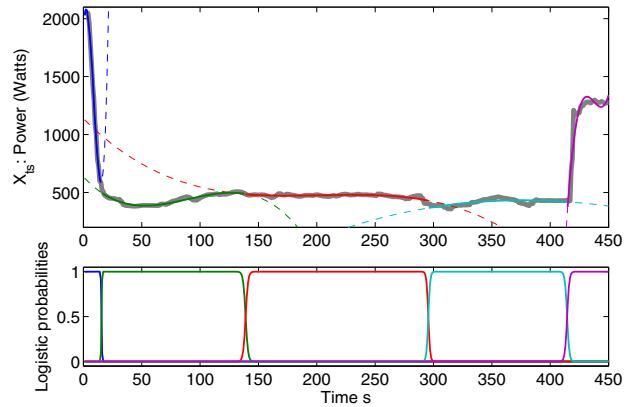
where  $\theta_t = \{\alpha, \sigma_1^2, \dots, \sigma_K^2, \boldsymbol{\beta}_{t1}, \dots, \boldsymbol{\beta}_{tK}\}$  is the parameter of the model.

Given a curve  $\mathbf{x}_t = (x_{t1}, \dots, x_{tS})'$ , the parameter  $\theta_t$  is estimated by maximizing the log-likelihood  $\log P(\mathbf{x}_t; \theta_t)$  through the EM algorithm (Dempster et al. 1977) whose complete description can be found in (Chamroukhi et al. 2009a,b). As an illustration, Fig. 2 shows the “static” modeling provided by this approach on a power consumption curve, with  $K = 5$  segments corresponding to the physical regimes, and a polynomial order  $p = 3$ . This specific polynomial order was found by the experts to be appropriated to represent the regimes involved in the power consumption curves. On the upper graphic, the observed curve is displayed in gray and the estimated polynomial curves are colored, the solid lines corresponding to the intervals where logistic probabilities are greater than 0.8. The bottom graphic corresponds to the estimated logistic probabilities associated to the five components.

The next section describes the matrix formulation of the RHLP model that will be useful to introduce the dynamic factor analytic extension of this model.

### 3.3 Matrix formulation of the model

The equivalent matrix formulation of the RHLP model, in its homoscedastic version, that highlights the linear relationship



**Fig. 2** Regression model estimated from a power consumption curve, with  $K = 5$  segments and a polynomial order  $p = 3$ : (top) the observed curve is displayed in gray and its estimated polynomials are colored; (bottom) corresponding logistic probabilities

between  $\mathbf{x}_t$  and the complete set of regression coefficients  $\boldsymbol{\beta}_t = (\boldsymbol{\beta}'_{t1}, \dots, \boldsymbol{\beta}'_{tK})'$  is defined by

$$\mathbf{x}_t = g(z_t) \boldsymbol{\beta}_t + \sigma \boldsymbol{\varepsilon}_t, \quad (8)$$

where  $g(z_t) = (\tilde{z}_{t1} \mathbf{U} \dots \tilde{z}_{tK} \mathbf{U})$  and  $\boldsymbol{\varepsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_S)$  is a Gaussian noise in  $\mathbb{R}^S$ , and where  $\tilde{z}_{tk}$  ( $k = 1, \dots, K$ ) and  $\mathbf{U}$  are, respectively, the diagonal matrix and the regression matrix defined by

$$\tilde{z}_{tk} = \begin{bmatrix} z_{t1k} & 0 \\ \ddots & \ddots \\ 0 & z_{tSk} \end{bmatrix} \quad \text{and} \quad \mathbf{U} = \begin{bmatrix} \mathbf{u}'_1 \\ \vdots \\ \mathbf{u}'_S \end{bmatrix}. \quad (9)$$

From this matrix formulation, the homoscedastic version of the RHLP model can be seen as a basic regression model whose design matrix  $g(z_t)$  depends on the latent segmentation  $z_t$ .

### 4 Dynamical factor modeling of a sequence of curves with changes in regime

The dynamic factor analytic extension of the RHLP model is introduced in this section, where we are interested in modeling the temporal evolution of a sequence of curves. It consists in allowing the vector  $\boldsymbol{\beta}_t$ , which concatenates the regression coefficients of the  $K$  segments, to evolve across the curves through low-dimensional dynamic factors. This kind of modeling is therefore of great interest for data visualization.

In this section, we denote as  $(z_1, \dots, z_T)$  the discrete random processes associated to the sequence  $(\mathbf{x}_1, \dots, \mathbf{x}_T)$ , with  $z_t = (z_{t1}, \dots, z_{tS})'$ , where  $z_{ts} \in \{1, \dots, K\}$ .

#### 4.1 Model definition

Based on the matrix formulation described in Sect. 3.3, the proposed dynamic factor model for a sequence of curves is defined by the equations

$$\mathbf{x}_t = g(\mathbf{z}_t) \cdot (\mathbf{A} \mathbf{f}_t + \mathbf{b}) + \sigma \boldsymbol{\varepsilon}_t \quad \forall t, \quad (10)$$

$$\mathbf{f}_t = \mathbf{f}_{t-1} + \boldsymbol{\eta}_t \quad \forall t, \quad (11)$$

$$z_{ts} \sim \mathcal{M}(1; \pi_1(s; \boldsymbol{\alpha}), \dots, \pi_K(s; \boldsymbol{\alpha})) \quad \forall t, s, \quad (12)$$

where the initial factor  $\mathbf{f}_0$  is considered here as a parameter,  $\boldsymbol{\varepsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_S)$  and  $\boldsymbol{\eta}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_q)$  are, respectively,  $S$ -dimensional and  $q$ -dimensional Gaussian noises,  $\mathbf{A}$  is a  $K(p+1) \times q$  matrix, and  $\mathbf{b}$  is a  $K(p+1) \times 1$  level vector. The  $q$ -dimensional latent factors  $\mathbf{f}_t$  generating the regression coefficients are supposed to evolve over time according to a Gaussian random walk with variance  $\mathbf{I}_q$ . For identifiability purposes, the matrix  $\mathbf{A}'\mathbf{A}$  is constrained to be diagonal and, as suggested by Zuur et al. (2003), the estimates of  $(\mathbf{f}_1, \dots, \mathbf{f}_T)$  are centered (see Sect. 5.2).

Although more general models such as ARMA models could have been used to describe the latent factors, a Gaussian random walk was preferred in this study for its simplicity and by the fact that no prior information about the factors dynamic was available. As for the model defined by Eqs. (4) and (5), we suppose that the components  $z_{ts}$  ( $s = 1, \dots, S$ ) of the random process  $\mathbf{z}_t$  are independently generated from the multinomial law  $\mathcal{M}(1; \pi_1(s; \boldsymbol{\alpha}), \dots, \pi_K(s; \boldsymbol{\alpha}))$ , where the logistic functions  $\pi_k(s; \boldsymbol{\alpha})$  are those defined by Eq. (6). The set of parameters of this model will be denoted by  $\boldsymbol{\theta} = \{\boldsymbol{\alpha}, \mathbf{f}_0, \mathbf{b}, \mathbf{A}, \sigma^2\}$ .

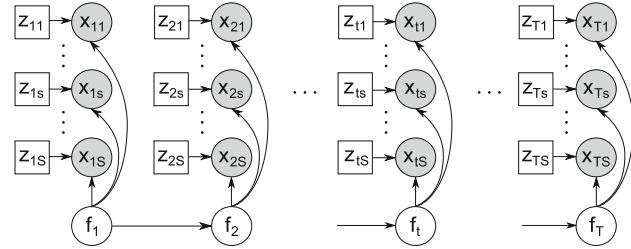
It can be shown that conditionally on  $\mathbf{f}_t$ ,  $x_{ts}$  is distributed according to the mixture of Gaussian densities

$$P(x_{ts} | \mathbf{f}_t; \boldsymbol{\theta}) = \sum_k \pi_k(s; \boldsymbol{\alpha}) \times \mathcal{N}\left(x_{ts}; \mathbf{u}'_s (\mathbf{A}_k \mathbf{f}_t + \mathbf{b}_k), \sigma^2\right), \quad (13)$$

and that, conditionally on  $\mathbf{f}_{t-1}$ ,  $\mathbf{f}_t$  is distributed according to the Gaussian density

$$P(\mathbf{f}_t | \mathbf{f}_{t-1}; \boldsymbol{\theta}) = \mathcal{N}(\mathbf{f}_t; \mathbf{f}_{t-1}, \mathbf{I}_q), \quad (14)$$

where  $\forall k = 1, \dots, K$ ,  $\mathbf{A}_k$  and  $\mathbf{b}_k$  are, respectively, the  $(p+1) \times q$  matrix and  $(p+1) \times 1$  vector resulting from the following block decompositions of  $\mathbf{A}$  and  $\mathbf{b}$ :



**Fig. 3** Graphical representation of the proposed dynamic factor analytic model for functional data; the observed variables are displayed in gray and the discrete variables are the squared ones

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 \\ \vdots \\ \mathbf{A}_K \end{bmatrix} \quad \text{and} \quad \mathbf{b} = \begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_K \end{bmatrix}.$$

Figure 3 shows the graphical representation of the variables involved in the proposed model and their conditional dependencies.

#### 4.2 A dynamic model with specific functional factor loadings

The model described in Sect. 4.1 can also be regarded as a dynamic factor model with specific functional factor loadings. For this purpose, let us observe that Eq.(10) can be developed as

$$x_{ts} = \left( \sum_{k=1}^K z_{tsk} \mathbf{u}'_s \mathbf{b}_k \right) + \sum_{j=1}^q f_{tj} \left( \sum_{k=1}^K z_{tsk} \mathbf{u}'_s \mathbf{a}_{kj} \right) + \sigma \varepsilon_{ts},$$

where the  $f_{tj}$  ( $j = 1, \dots, q$ ) are the components of  $\mathbf{f}_t$ , the  $\varepsilon_{ts}$  ( $s = 1, \dots, S$ ) are the components of  $\boldsymbol{\varepsilon}_t$ , and the  $\mathbf{a}_{kj}$  ( $j = 1, \dots, q$ ) are the columns of the block  $\mathbf{A}_k$  of matrix  $\mathbf{A}$ .

Approximating each variable  $z_{tsk}$  by its expectation  $\pi_k(s; \boldsymbol{\alpha})$ , which no longer depends on  $t$ , leads to the following model:

$$x_{ts} = \mu(s) + \boldsymbol{\varphi}(s)' \mathbf{f}_t + \sigma \varepsilon_{ts} \quad \forall t, s, \quad (15)$$

$$\mathbf{f}_t = \mathbf{f}_{t-1} + \boldsymbol{\eta}_t, \quad \forall t, \quad (16)$$

with  $\boldsymbol{\varphi}(s) = (\varphi_1(s), \dots, \varphi_q(s))'$ , where

$$\varphi_j(s) = \sum_k \pi_k(s; \boldsymbol{\alpha}) \mathbf{u}'_s \mathbf{a}_{kj} \quad \forall j = 1, \dots, q, \quad (17)$$

$$\mu(s) = \sum_k \pi_k(s; \boldsymbol{\alpha}) \mathbf{u}'_s \mathbf{b}_k. \quad (18)$$

This model corresponds to a functional dynamic factor model as the one introduced by Hays et al. (2012) whose loadings are sums of polynomials weighted by logistic functions.

## 5 Parameter estimation

Given an observed curve sequence  $\mathbf{x} = (\mathbf{x}'_1, \dots, \mathbf{x}'_T)'$ , the usual way to estimate the parameters of the proposed dynamic factor model is to maximize the log-likelihood criterion. Denoting the hidden sequences by  $\mathbf{z} = (z'_1, \dots, z'_T)'$  and  $\mathbf{f} = (f'_1, \dots, f'_T)'$ , this criterion is written as

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}) &= \log P(\mathbf{x}; \boldsymbol{\theta}) \\ &= \log \left( \sum_{\mathbf{z}} P(\mathbf{z}; \boldsymbol{\theta}) \int P(\mathbf{x}, \mathbf{f} | \mathbf{z}; \boldsymbol{\theta}) d\mathbf{f} \right),\end{aligned}\quad (19)$$

which corresponds to the logarithm of a mixture of  $K^T S$  Gaussian components, each corresponding to a particular realization of the sequence  $\mathbf{z}$ . Although this latent variable model suggests using the EM algorithm for parameter estimation, the direct application of this algorithm is intractable. This is due to the fact that the number of probabilities required by the E-step increases exponentially with  $T$  and  $S$ . An alternative solution would consist in evaluating the required posterior distributions using Monte Carlo Markov Chain (MCMC) approaches (Neal 1993), but in this paper, we opted for a variational method (Jordan et al. 1999; Beal 2003; Blei and Lafferty 2006) which is computationally more efficient than MCMC. It results in the maximization of a lower bound of the log-likelihood  $\mathcal{L}(\boldsymbol{\theta})$  using an EM-like algorithm.

### 5.1 Variational criterion

According to the variational approximation theory (Jordan et al. 1999; Beal 2003), we maximize the lower bound of the log-likelihood, defined by

$$\mathcal{F}(Q, \boldsymbol{\theta}) = \mathbb{E}_Q \left[ \log \left( \frac{P(\mathbf{x}, \mathbf{f}, \mathbf{z}; \boldsymbol{\theta})}{Q(\mathbf{z}, \mathbf{f})} \right) \right], \quad (20)$$

where  $Q$  denotes any probability distribution over the hidden variables  $(\mathbf{z}, \mathbf{f})$  and  $\mathbb{E}_Q[\cdot]$  denotes the expectation operator over these variables, with respect to  $Q$ . Starting from an initial parameter  $\boldsymbol{\theta}^{(0)}$ , the EM algorithm then consists in alternating the following two steps until convergence (Neal and Hinton 1998):

- E-step :  $Q^{(c+1)} = \arg \max_Q \mathcal{F}(Q, \boldsymbol{\theta}^{(c)})$ ,
- M-step :  $\boldsymbol{\theta}^{(c+1)} = \arg \max_{\boldsymbol{\theta}} \mathcal{F}(Q^{(c+1)}, \boldsymbol{\theta})$ .

Due to the multiple variables interactions of the proposed model, some constraints are added to the distribution  $Q$  so that the maximization defined by the E-step can be solved more easily. For our purposes, the usual mean field approximation (Jordan et al. 1999; Beal 2003), which postulates that the distribution  $Q$  can be factorized independently over

the latent variables, has been adopted. It has further been assumed that the distribution of the continuous latent variables  $f_t$  is Gaussian. Consequently, the distribution  $Q$  can be written as

$$Q(\mathbf{z}, \mathbf{f}) = \prod_{t,s,k} (\tau_{tsk})^{z_{tsk}} \prod_t \mathcal{N}(\mathbf{f}_t; \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t), \quad (21)$$

where the  $\tau_{tsk}$ , which verify  $\tau_{tsk} > 0$  and  $\sum_k \tau_{tsk} = 1$ , are the expectations of the binary variables  $z_{tsk}$ , and the couples  $(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$  are the means and covariance matrices of the continuous latent variables  $\mathbf{f}_t$ . Since the distribution  $Q$  is entirely characterized by the three sets of variational parameters  $\boldsymbol{\tau} = (\tau_{tsk})_{tsk}$ ,  $\boldsymbol{\mu} = (\boldsymbol{\mu}_t)_t$  and  $\boldsymbol{\Sigma} = (\boldsymbol{\Sigma}_t)_t$ , the criterion  $\mathcal{F}$  can be written as a function of  $(\boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\theta})$ . As shown in Appendix 1, this criterion is given, up to an additive constant, by

$$\begin{aligned}\mathcal{F}(\boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\theta}) &= \sum_{t,s,k} \tau_{tsk} \mathcal{A}_k(x_{ts}, \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t, \boldsymbol{\theta}) \\ &\quad + \sum_t \mathcal{B}(\boldsymbol{\mu}_{t-1}, \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_{t-1}, \boldsymbol{\Sigma}_t) \\ &\quad - \sum_{t,s,k} \tau_{tsk} \log \tau_{tsk} + \frac{1}{2} \sum_t \log \det(\boldsymbol{\Sigma}_t),\end{aligned}\quad (22)$$

with

$$\begin{aligned}\mathcal{A}_k(x_{ts}, \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t, \boldsymbol{\theta}) &= \log \left( \pi_k(s; \boldsymbol{\alpha}) \mathcal{N}(x_{ts}; \mathbf{u}'_s (\mathbf{A}_k \boldsymbol{\mu}_t + \mathbf{b}_k), \sigma^2) \right) \\ &\quad - \frac{1}{2\sigma^2} (\mathbf{u}'_s \mathbf{A}_k \boldsymbol{\Sigma}_t \mathbf{A}'_k \mathbf{u}_s) \quad \forall t, s, k,\end{aligned}\quad (23)$$

and

$$\begin{aligned}\mathcal{B}(\boldsymbol{\mu}_{t-1}, \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_{t-1}, \boldsymbol{\Sigma}_t) &= \log \mathcal{N}(\boldsymbol{\mu}_t; \boldsymbol{\mu}_{t-1}, \mathbf{I}_q) \\ &\quad - \frac{1}{2} \left( \text{Tr}(\boldsymbol{\Sigma}_t) + \text{Tr}(\boldsymbol{\Sigma}_{t-1}) \right) \forall t,\end{aligned}\quad (24)$$

by adopting the convention  $\boldsymbol{\mu}_0 = \mathbf{f}_0$  and  $\boldsymbol{\Sigma}_0 = \mathbf{0}$ .

### 5.2 Variational EM algorithm

Starting from an initial parameter  $\boldsymbol{\theta}^{(0)}$ , the proposed variational EM algorithm consists in alternating the following two steps until convergence of the criterion  $\mathcal{F}$ .

#### 5.2.1 Variational E-step

Denoting by  $c$  the current iteration, this step consists in solving the problem

$$(\boldsymbol{\tau}^{(c+1)}, \boldsymbol{\mu}^{(c+1)}, \boldsymbol{\Sigma}^{(c+1)}) = \operatorname{argmax}_{(\boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\Sigma})} \mathcal{F}(\boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\theta}^{(c)}),$$

which itself requires using a fixed point iterative procedure. In this article, we retain the strategy which consists in starting from the variational parameters  $(\boldsymbol{\mu}^{(c)}, \boldsymbol{\Sigma}^{(c)})$  obtained at the previous iteration, and then iterating the following three maximizations until convergence:

$$\boldsymbol{\tau}^{(c+1)} = \operatorname{argmax}_{\boldsymbol{\tau}} \mathcal{F}(\boldsymbol{\tau}, \boldsymbol{\mu}^{(c)}, \boldsymbol{\Sigma}^{(c)}, \boldsymbol{\theta}^{(c)}), \quad (25)$$

$$\boldsymbol{\mu}^{(c+1)} = \operatorname{argmax}_{\boldsymbol{\mu}} \mathcal{F}(\boldsymbol{\tau}^{(c+1)}, \boldsymbol{\mu}, \boldsymbol{\Sigma}^{(c)}, \boldsymbol{\theta}^{(c)}), \quad (26)$$

$$\boldsymbol{\Sigma}^{(c+1)} = \operatorname{argmax}_{\boldsymbol{\Sigma}} \mathcal{F}(\boldsymbol{\tau}^{(c+1)}, \boldsymbol{\mu}^{(c+1)}, \boldsymbol{\Sigma}, \boldsymbol{\theta}^{(c)}). \quad (27)$$

This strategy gave good results in practice. It should be noticed that, with this strategy, the algorithm has to be initialized with the triplet  $(\boldsymbol{\theta}^{(0)}, \boldsymbol{\mu}^{(0)}, \boldsymbol{\Sigma}^{(0)})$ .

The maximization problem defined by Eq. (25) is solved, under the constraints  $\sum_k \tau_{tsk} = 1, \forall t, s$ . The method of Lagrange multipliers yields,  $\forall t, s, k$ ,

$$\tau_{tsk}^{(c+1)} = \frac{\exp(\mathcal{A}_k(x_{ts}, \boldsymbol{\mu}_t^{(c)}, \boldsymbol{\Sigma}_t^{(c)}, \boldsymbol{\theta}^{(c)}))}{\sum_\ell \exp(\mathcal{A}_\ell(x_{ts}, \boldsymbol{\mu}_t^{(c)}, \boldsymbol{\Sigma}_t^{(c)}, \boldsymbol{\theta}^{(c)}))}. \quad (28)$$

As shown in Appendix 2, the sequence  $\boldsymbol{\mu}^{(c+1)}$  verifying Eq. (26) is given by the following Forward–Backward recursions:

- Forward: starting from  $\tilde{\boldsymbol{\Sigma}}_0 = \mathbf{0}$  and  $\tilde{\boldsymbol{\mu}}_0 = \boldsymbol{f}_0^{(c)}$ , compute, for  $t = 1, \dots, T$ ,

$$\tilde{\boldsymbol{\Sigma}}_t = \left[ \boldsymbol{D}_t^{(c+1)} + (\mathbf{I}_q + \tilde{\boldsymbol{\Sigma}}_{t-1})^{-1} \right]^{-1}, \quad (29)$$

$$\tilde{\boldsymbol{\mu}}_t = \tilde{\boldsymbol{\mu}}_{t-1} + \tilde{\boldsymbol{\Sigma}}_t \left( \boldsymbol{d}_t^{(c+1)} - \boldsymbol{D}_t^{(c+1)} \tilde{\boldsymbol{\mu}}_{t-1} \right), \quad (30)$$

with

$$\boldsymbol{d}_t^{(c+1)} = \frac{1}{\sigma^{2(c)}} \sum_{s,k} \tau_{tsk}^{(c+1)} \boldsymbol{A}_k^{(c)'} \boldsymbol{u}_s (x_{ts} - \boldsymbol{u}'_s \boldsymbol{b}_k^{(c)}), \quad (31)$$

$$\boldsymbol{D}_t^{(c+1)} = \frac{1}{\sigma^{2(c)}} \sum_{s,k} \tau_{tsk}^{(c+1)} \boldsymbol{A}_k^{(c)'} \boldsymbol{u}_s \boldsymbol{u}'_s \boldsymbol{A}_k^{(c)}. \quad (32)$$

- Backward: starting from  $\boldsymbol{\mu}_T^{(c+1)} = \tilde{\boldsymbol{\mu}}_T$ , compute retrospectively, for  $t = T-1, \dots, 1$ ,

$$\boldsymbol{\mu}_t^{(c+1)} = \tilde{\boldsymbol{\mu}}_t + \boldsymbol{J}_t \left( \boldsymbol{\mu}_{t+1}^{(c+1)} - \tilde{\boldsymbol{\mu}}_t \right), \quad (33)$$

with  $\boldsymbol{J}_t = \tilde{\boldsymbol{\Sigma}}_t (\mathbf{I}_q + \tilde{\boldsymbol{\Sigma}}_t)^{-1}$ .

For reasons of identifiability of the initial latent factor, we adopt the strategy proposed by Zuur et al. (2003) which consists, after performing the Forward–Backward recursions, to center the sequence  $(\boldsymbol{\mu}_1^{(c+1)}, \dots, \boldsymbol{\mu}_T^{(c+1)})$ , by setting

$$\boldsymbol{\mu}_t^{(c+1)} \leftarrow \boldsymbol{\mu}_t^{(c+1)} - \frac{1}{T} \sum_t \boldsymbol{\mu}_t^{(c+1)} \quad \forall t. \quad (34)$$

This strategy, which works well in practice, has been used conjointly with the updating of the initial latent factor  $\boldsymbol{f}_0^{(c)}$  (M-step).

The maximization defined by Eq. (27) can easily be solved by canceling the partial derivatives of  $\mathcal{F}$  with respect to each of the matrices  $\boldsymbol{\Sigma}_t$ . We get

$$\boldsymbol{\Sigma}_t^{(c+1)} = \begin{cases} \left( 2\mathbf{I}_q + \boldsymbol{D}_t^{(c+1)} \right)^{-1} & \text{if } t \leq T-1, \\ \left( \mathbf{I}_q + \boldsymbol{D}_T^{(c+1)} \right)^{-1} & \text{if } t = T. \end{cases} \quad (35)$$

### 5.2.2 M-Step

This step consists in computing the parameter vector  $\boldsymbol{\theta}^{(c+1)}$  which maximizes  $\mathcal{F}$  w.r.t.  $\boldsymbol{\theta}$ . This is done by performing the following three maximizations:

$$\boldsymbol{f}_0^{(c+1)} = \operatorname{argmax}_{\boldsymbol{f}_0} \mathcal{F}(\boldsymbol{\tau}^{(c+1)}, \boldsymbol{\mu}^{(c+1)}, \boldsymbol{\Sigma}^{(c+1)}, \boldsymbol{\theta}), \quad (36)$$

$$\boldsymbol{\alpha}^{(c+1)} = \operatorname{argmax}_{\boldsymbol{\alpha}} \mathcal{F}(\boldsymbol{\tau}^{(c+1)}, \boldsymbol{\mu}^{(c+1)}, \boldsymbol{\Sigma}^{(c+1)}, \boldsymbol{\theta}), \quad (37)$$

$$(\boldsymbol{A}^{(c+1)}, \boldsymbol{b}^{(c+1)}) = \operatorname{argmax}_{(\boldsymbol{A}, \boldsymbol{b})} \mathcal{F}(\boldsymbol{\tau}^{(c+1)}, \boldsymbol{\mu}^{(c+1)}, \boldsymbol{\Sigma}^{(c+1)}, \boldsymbol{\theta}), \quad (38)$$

$$\sigma^{2(c+1)} = \operatorname{argmax}_{\sigma^2} \mathcal{F}(\boldsymbol{\tau}^{(c+1)}, \boldsymbol{\mu}^{(c+1)}, \boldsymbol{\Sigma}^{(c+1)}, \boldsymbol{\theta}), \quad (39)$$

detailed in Appendix 3. It should be noticed that  $\boldsymbol{\alpha}^{(c+1)}$  is obtained through the well-known Iteratively Reweighted Least Squares (IRLS) algorithm (Green 1984) which is equivalent to a Newton–Raphson algorithm.

### 5.2.3 Pseudo-code of the Variational EM algorithm

The main steps of the proposed variational EM procedure are summarized in Algorithm 1 whose Matlab code can be obtained from the following link: <http://allousame.free.fr/softwares/sfdfa/>.

### 5.2.4 Starting values

It is well known that EM-type algorithms converge toward a local optimum of their objective function, which depends on initial parameters. This is even more pronounced for relatively large parameter sizes. To allow our variational EM algorithm to converge toward a good local optimum, several strategies were tested. Finally, the following initialization strategy has been retained:

- the multivariate version of the RHL approach is jointly run on the curves  $(\mathbf{x}_1, \dots, \mathbf{x}_T)$ ; more specifically, this

**Algorithm 1:** Variational EM algorithm for segmental functional dynamic factor analysis

---

**Input:** sequence of curves  $(\mathbf{x}_1, \dots, \mathbf{x}_T)$ , number of factors  $q$ , number of segments  $K$ , polynomial order  $p$ , initial parameter  $\boldsymbol{\theta}^{(0)}$ , initial variational parameters  $(\boldsymbol{\mu}^{(0)}, \boldsymbol{\Sigma}^{(0)})$

 $c \leftarrow 0$ 
**repeat**

- E-step**
- Compute  $\tau_{tsk}^{(c+1)} \forall t, s, k$  (Eq. 28)
- Compute  $\boldsymbol{\mu}_t^{(c+1)} \forall t$  (Eq. 29–34)
- Compute  $\boldsymbol{\Sigma}_t^{(c+1)} \forall t$  (Eq. 35)
- M-step**
- Compute  $\mathbf{f}_0^{(c+1)}$  (Eq. 68)
- Compute  $\boldsymbol{\alpha}^{(c+1)}$  (IRLS: Algo. 3)
- Compute  $\mathbf{b}_k^{(c+1)}$  and  $\mathbf{A}_k^{(c+1)} \forall k$  (Eq. 74,75)
- Transform  $\mathbf{A}^{(c+1)}, \mathbf{f}_0^{(c+1)}$  and  $\boldsymbol{\mu}^{(c+1)}$  (Eq. 76–78)
- Compute  $\sigma^2{}^{(c+1)}$  (Eq. 79)
- $c \leftarrow c + 1$

**until** the variational criterion  $\mathcal{F}$  converges;

**Output:** Parameter vector  $\hat{\boldsymbol{\theta}}$  and variational parameters  $(\hat{\boldsymbol{\tau}}, \hat{\boldsymbol{\Sigma}}, \hat{\boldsymbol{\mu}})$

---

method allows common logistic regression coefficients and variances to be estimated from these curves;

- the initial parameters  $\boldsymbol{\alpha}$  and  $\sigma^2$  are, respectively, set to the obtained logistic regression coefficients and averaged variances;
- a joint segmentation of the curves  $(\mathbf{x}_1, \dots, \mathbf{x}_T)$  (along the time axis subscripted by  $s$ ) is deduced from the logistic functions  $\pi_k(\cdot, \boldsymbol{\alpha}) (k = 1, \dots, K)$ ; for each curve  $\mathbf{x}_t$  ( $t = 1, \dots, T$ ), let  $\boldsymbol{\beta}_t \in \mathbb{R}^{K(d+1)}$  denote the regression coefficients concatenated over the  $K$  segments;
- the parameters  $\mathbf{A}$  and  $\mathbf{b}$  are initialized by applying a classical dynamic factor analysis with  $q$  factors on the multivariate dataset  $(\boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_T)$ ;
- the variational parameters  $\boldsymbol{\mu}_t$  ( $t = 1, \dots, T$ ) are initialized as the estimated dynamic factors [see Eq.(3)] and the initial factor  $\mathbf{f}_0$  is set to  $\boldsymbol{\mu}_1$ ; and
- finally, the variational parameters  $\boldsymbol{\Sigma}_t$  ( $t = 1, \dots, T$ ) are initialized to the identity matrix.

### 5.3 Algorithmic complexity

The algorithmic complexity of the variational E-Step is  $O(TSKq^3p)$  which mainly comprises the computation of the probabilities  $\pi_k(s; \boldsymbol{\alpha})$ , the computation of the densities  $\mathcal{N}(x_{ts}; \mathbf{u}'_s(\mathbf{A}_k \boldsymbol{\mu}_t + \mathbf{b}_k), \sigma^2)$ , and the inversion of  $q \times q$  matrices required by the Forward–Backward recursions as well as for the updating of  $\boldsymbol{\Sigma}_t$ . For its part, the M-step has a computational complexity of  $O(I_{IRLS}TSK^3p^3q^3)$ , where  $I_{IRLS}$  is the number of iterations of the IRLS loop. This complexity comprises the inversion of a  $2(K - 1) \times 2(K - 1)$  matrix in the IRLS algorithm, the inversion of a  $p \times p$  matrix

to update  $\mathbf{b}_k$ , and the inversion of a  $qp \times qp$  matrix to update  $\mathbf{A}_k$ . Therefore, we can conclude that the proposed variational EM algorithm has a computational complexity of order  $O(I_{EM}I_{IRLS}TSK^3p^3q^3)$ , where  $I_{EM}$  is the number of iterations required by the EM algorithm. From a practical point of view, the computation time was found reasonable for  $T \leq 1000$ ,  $S \leq 1000$  and  $K \leq 10$ .

### 5.4 Model selection

The specification of the correct number of factors is a central problem for any factor analytic approach. For our model, in addition to the factor dimension, both the number of segments and polynomial order have to be specified. Simultaneously choosing the correct number of segments and polynomial order remains a sensitive issue. Indeed, given a small number of segments, a good fitting of curves can be found using polynomials with a relatively high degree. In the same way, given a small polynomial degree, an accurate fitting can be found using a relatively high number of segments. We therefore suggest setting manually the polynomial order to a relatively small value (not exceeding 3) according to the desired shape of the segments, and then focus the model selection task on the number of factors  $q$  and the number of segments  $K$ . Within the framework of our dynamic mixture model, we propose using the following approximation of the Bayesian Information Criterion (BIC), where the log-likelihood is replaced by its lower bound  $\mathcal{F}$ :

$$BIC(K, q) \approx -2\mathcal{F}(\hat{\boldsymbol{\tau}}, \hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}}, \hat{\boldsymbol{\theta}}) + v(K, q) \log(TS), \quad (40)$$

where the parameters  $(\hat{\boldsymbol{\tau}}, \hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}}, \hat{\boldsymbol{\theta}})$  are those estimated by the variational EM algorithm and  $v(K, q)$  is the number of free parameters of the model. In this model, we have

$$v(K, q) = K(d + 1)(q + 1) - \frac{1}{2}q(q - 1) + 2K - 1, \quad (41)$$

which can be decomposed as follows:

Parameter	Number of free parameters
$\boldsymbol{\alpha}$	$2(K - 1)$
$\mathbf{A}$	$K(d + 1)q - \frac{1}{2}q(q - 1)$
$\mathbf{b}$	$K(d + 1)$
$\sigma^2$	1

From a practical point of view, the variational EM algorithm is run for  $K = 1, \dots, K_{max}$  and  $q = 1, \dots, q_{max}$ , and the couple  $(K, q)$  which maximizes BIC is selected.

## 6 Uses of the model

Various uses of model can been made from the estimates  $(\widehat{\boldsymbol{\tau}}, \widehat{\boldsymbol{\mu}}, \widehat{\boldsymbol{\Sigma}}, \widehat{\boldsymbol{\theta}})$  provided by the variational EM algorithm. These ones are given below.

### 6.1 Extracting low-dimensional dynamic factors

The latent dynamic factors allowing the functional time series to be represented onto a lower dimensional space are estimated as follows:

$$\widehat{\mathbf{f}}_t = \widehat{\boldsymbol{\mu}}_t \quad \forall t. \quad (42)$$

As mentioned in Sect. 4.2, these factors can be associated to specific functional loadings taking the form of polynomial piecewise functions [see Eq. (17)].

### 6.2 Extracting a joint segmentation from curves

A joint segmentation of the set of curves along the time axis subscripted by  $s$  is derived by defining the  $k$ -th segment ( $k = 1, \dots, K$ ) as follows:

$$E_k = \{s \in [1; S] ; \pi_k(s; \widehat{\boldsymbol{\alpha}}) \geq \pi_\ell(s; \widehat{\boldsymbol{\alpha}}) \quad \forall \ell \neq k\}. \quad (43)$$

### 6.3 Functional time series approximation

The functional time series is approximated (or denoised) using the following formulae:

$$\widehat{x}_{ts} = \sum_k \pi_k(s; \widehat{\boldsymbol{\alpha}}) \mathbf{u}'_s (\widehat{\mathbf{A}}_k \widehat{\boldsymbol{\mu}}_t + \widehat{\mathbf{b}}_k) \quad \forall t, s, \quad (44)$$

which is the sum of the estimated polynomials weighted by the logistic functions.

### 6.4 Variational filtering and prediction

Given the parameter vector  $\widehat{\boldsymbol{\theta}}$ , approximating the distribution of  $(\mathbf{z}_t, \mathbf{f}_t)$  conditionally on the observations  $(\mathbf{x}_1, \dots, \mathbf{x}_t)$  (filtering) and conditionally on the observations  $(\mathbf{x}_1, \dots, \mathbf{x}_{t-1})$  (prediction) is of great interest in numerous applications involving real-time processing of data. The set of observations  $(\mathbf{x}_1, \dots, \mathbf{x}_t)$  will be denoted by  $\mathbf{x}_{1:t}$  in this section.

#### 6.4.1 Filtering

According to the variational approach adopted in this paper, we use,  $\forall t$ , the following approximation:

$$\begin{aligned} P(\mathbf{z}_t, \mathbf{f}_t | \mathbf{x}_{1:t}; \widehat{\boldsymbol{\theta}}) &\approx Q_t(\mathbf{z}_t, \mathbf{f}_t) \\ &= \prod_{s,k} \tau_{tsk}^{z_{tsk}} \times \mathcal{N}(\mathbf{f}_t; \mathbf{m}_t, \mathbf{C}_t), \end{aligned} \quad (45)$$

where  $Q_t$  denotes the approximated distribution characterized by the variational parameters  $\mathbf{m}_t$ ,  $\mathbf{C}_t$ , and  $\boldsymbol{\tau}_t = (\tau_{tsk})_{sk}$ . These ones are determined by minimizing the Kullback–Leibler divergence between the true posterior distribution and its approximation  $Q_t$ , a problem which is equivalent to maximizing the criterion  $\mathcal{F}_t$  defined by

$$\mathcal{F}_t(\boldsymbol{\tau}_t, \mathbf{m}_t, \mathbf{C}_t) = \mathbb{E}_{Q_t} \left[ \log \frac{P(\mathbf{x}_t, \mathbf{z}_t, \mathbf{f}_t | \mathbf{x}_{1:t-1}; \widehat{\boldsymbol{\theta}})}{Q_t(\mathbf{z}_t, \mathbf{f}_t)} \right]. \quad (46)$$

Similar calculations as those made for the E-step of the variational EM algorithm lead us to fixed point equations involving  $\boldsymbol{\tau}_t$ ,  $\mathbf{m}_t$ , and  $\mathbf{C}_t$ , which can be solved using Algorithm 2. Within the framework of linear state-space models with time-varying measurement noise variance, a similar variational filtering algorithm has been obtained by Sarkka and Nummenmaa (2009). In practice, the convergence of this algorithm is achieved in less than 5 iterations.

#### 6.4.2 Prediction

Given the sequence  $(\mathbf{x}_1, \dots, \mathbf{x}_t)$ , one can be interested in computing future values of both latent variables and observations. From the variational approximation of the filtering distribution [see Eq. (45)], the following one-step-ahead predictive distribution of the latent variables can be deduced:

#### Algorithm 2: Variational filtering

---

**Input:** Parameter  $\boldsymbol{\theta}$ , current observation  $\mathbf{x}_t = (x_{ts})_s$ , previous estimates  $\mathbf{m}_{t-1}$ ,  $\mathbf{C}_{t-1}$ , with the convention  $\mathbf{m}_0 = \mathbf{f}_0$  and  $\mathbf{C}_0 = 0$

**Initialization**

$\mathbf{m}_t^{(0)}$	=	$\mathbf{m}_{t-1}$
$\mathbf{C}_t^{(0)}$	=	$\mathbf{C}_{t-1}$
$c$	=	0

**repeat**

$\tau_{tsk}^{(c+1)}$	$\propto$	$\exp \left( \mathcal{A}_k(x_{ts}, \mathbf{m}_t^{(c)}, \mathbf{C}_t^{(c)}, \boldsymbol{\theta}) \right) \forall s, k$
$\mathbf{D}_t^{(c+1)}$	=	$\frac{1}{\sigma^2} \sum_{s,k} \tau_{tsk}^{(c+1)} \mathbf{A}_k' \mathbf{u}_s \mathbf{u}'_s \mathbf{A}_k$
$\mathbf{d}_t^{(c+1)}$	=	$\frac{1}{\sigma^2} \sum_{s,k} \tau_{tsk}^{(c+1)} \mathbf{A}_k' \mathbf{u}_s (x_{ts} - \mathbf{u}'_s \mathbf{b}_k)$
$\mathbf{C}_t^{(c+1)}$	=	$\left[ \mathbf{D}_t^{(c+1)} + (\mathbf{I}_q + \mathbf{C}_{t-1})^{-1} \right]^{-1}$
$\mathbf{m}_t^{(c+1)}$	=	$\mathbf{m}_{t-1} + \mathbf{C}_t^{(c+1)} (\mathbf{d}_t^{(c+1)} - \mathbf{D}_t^{(c+1)} \mathbf{m}_{t-1})$
$c$	=	$c + 1$

**until** the variational criterion  $\mathcal{F}_t$  converges;

**Output:** Variational parameters  $\mathbf{m}_t = \mathbf{m}_t^{(c)}$ ,  $\mathbf{C}_t = \mathbf{C}_t^{(c)}$  and  $\boldsymbol{\tau}_t = (\tau_{tsk}^{(c)})_{sk}$

---

$$P(z_{t+1}, f_{t+1} | \mathbf{x}_{1:t}; \hat{\boldsymbol{\theta}}) = \prod_{s,k} \pi_k(s; \hat{\boldsymbol{\alpha}})^{z_{t+1,s,k}} \times \mathcal{N}(f_{t+1}; \mathbf{m}_t, \mathbf{I}_q + \mathbf{C}_t), \quad (47)$$

where  $\mathbf{m}_t$  and  $\mathbf{C}_t$  are given by Algorithm 2. The predictive distribution of the individual variables can then be written as

$$P(z_{t+1,s} | \mathbf{x}_{1:t}; \hat{\boldsymbol{\theta}}) = \prod_k \pi_k(s; \hat{\boldsymbol{\alpha}})^{z_{t+1,s,k}}, \quad (48)$$

$$P(f_{t+1} | \mathbf{x}_{1:t}; \hat{\boldsymbol{\theta}}) = \mathcal{N}(f_{t+1}; \mathbf{m}_t, \mathbf{I} + \mathbf{C}_t), \quad (49)$$

$$\begin{aligned} P(x_{t+1,s} | \mathbf{x}_{1:t}; \hat{\boldsymbol{\theta}}) &= \sum_k \pi_k(s; \hat{\boldsymbol{\alpha}}) \mathcal{N}(x_{ts}; \\ &\quad \mathbf{u}'_s (\hat{\mathbf{A}}_k \mathbf{m}_t + \hat{\mathbf{b}}_k), \hat{\sigma}^2 \\ &\quad + \mathbf{u}'_s \hat{\mathbf{A}}_k (\mathbf{I}_q + \mathbf{C}_t) \hat{\mathbf{A}}'_k \mathbf{u}_s). \end{aligned} \quad (50)$$

The following one-step-ahead forecasts can therefore be deduced:

$$\begin{aligned} \hat{z}_{t+1,s|t} &= \arg \max_k \mathbb{E}[z_{t+1,s,k} | \mathbf{x}_{1:t}; \hat{\boldsymbol{\theta}}] \\ &= \arg \max_k \pi_k(s; \hat{\boldsymbol{\alpha}}), \end{aligned} \quad (51)$$

$$\begin{aligned} \hat{f}_{t+1|t} &= \mathbb{E}[f_{t+1} | \mathbf{x}_{1:t}; \hat{\boldsymbol{\theta}}] \\ &= \mathbf{m}_t, \end{aligned} \quad (52)$$

$$\begin{aligned} \hat{x}_{t+1,s|t} &= \mathbb{E}[x_{t+1,s} | \mathbf{x}_{1:t}; \hat{\boldsymbol{\theta}}] \\ &= \sum_k \pi_k(s; \hat{\boldsymbol{\alpha}}) \mathbf{u}'_s (\hat{\mathbf{A}}_k \mathbf{m}_t + \hat{\mathbf{b}}_k). \end{aligned} \quad (53)$$

Although the main focus of this paper is descriptive, the one-step-ahead forecasts will mainly be used to check the predictive capabilities of the proposed dynamic model, as will be shown in the next section.

## 7 Experimental study

This section is devoted to the evaluation of the proposed segmental functional dynamic factor analysis approach, implemented via the variational EM algorithm described in Sect. 5, using simulated data and real data. It will be called SFDFA in the following. Before developing the experiments, the compared methods are described.

### 7.1 Compared approaches and their dimensioning

Four approaches are compared in this article:

- DFA: the dynamic factor analysis approach described in Section 2 and implemented via the EM algorithm (Zuur et al. 2003). As this approach is directly applied to the original sequence of curves  $(\mathbf{x}_1, \dots, \mathbf{x}_T)$  viewed as a sequence of  $T$  observations in  $\mathbb{R}^S$ , we use a diagonal

covariance matrix  $\mathbf{W}$  to reduce the number of parameters of the model. The number of free parameters of this model can thus be written as

$$v_{\text{DFA}} = S(q+2) - \frac{1}{2}q(q-1), \quad (54)$$

which can be decomposed as follows:

Parameter	Number of free parameters
$\mathbf{A}$	$Sq - \frac{1}{2}q(q-1)$
$\mathbf{b}$	$S$
$\mathbf{W}$	$S$

- FPCA: the functional principal component analysis method (Ramsay and Silverman 1997), which represents data by smooth functions. It consists, in practice, in converting the observations into functional objects [in this paper, we use the smoothing splines approach and choose the smoothing constant minimizing the cross-validated sum of squared errors (Gareth et al. 2000)] and then applying the principal component analysis method to the functions discretized on the temporal interval  $[1; S]$ . The number of free parameters of FPCA, which is composed of the principal components, is given by

$$v_{\text{FPCA}} = \frac{1}{2}S(S+1). \quad (55)$$

- RHLP-PCA: a two-stage approach which consists in the following steps:

- (i) first, a joint segmentation of the curves  $(\mathbf{x}_1, \dots, \mathbf{x}_T)$  into  $K$  segments found along the time axis indexed by  $s$  is performed using the multivariate version of RHLP. In this case, a common logistic regression parameter vector  $\boldsymbol{\alpha}$  is estimated for all the curves but each curve has its own polynomial regression coefficients. For each curve  $\mathbf{x}_t$  ( $t = 1, \dots, T$ ), let  $\boldsymbol{\beta}_t \in \mathbb{R}^{K(d+1)}$  denote the estimated  $d$ -th order polynomial regression coefficients concatenated over the  $K$  segments.
- (ii) then, PCA is performed on the sequence  $(\boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_T)$  and the scores corresponding to the first  $q$  factors are retained.

This approach, which does not consider any dynamic between curves, can be regarded as a baseline. Neglecting the number of parameters of step (i), the number of free parameters required by RHLP-PCA is given by

$$v_{\text{RHLP-PCA}} = \frac{1}{2}K(p+1)[K(p+1)+1]. \quad (56)$$

**Table 1** Numerical example illustrating the number of parameters to be estimated by the compared methods; the X symbol indicates that  $S$ ,  $K$ , or  $p$  is not used for the computation of the number of parameters

Method	$S$	$q$	$K$	$p$	Numb. of free param.
DFA	200	2	X	X	799
FPCA	200	2	X	X	20,100
SFDFA	X	2	10	1	78
RHLP-PCA	X	2	10	1	210
RHLP-DFA	X	2	10	1	79

– RHLP-DFA: a two-stage approach which differs from the previous approach only in the second step where PCA is replaced by DFA. Neglecting the number of parameters of step (i), the number of free parameters required by RHLP-DFA is given by

$$\nu_{\text{RHLP-DFA}} = K(p+1)(q+2) - \frac{1}{2}q(q-1). \quad (57)$$

– SFDFA: the segmental functional dynamic factor analysis approach, which is proposed in the present article, whose number of free parameters is

$$\nu_{\text{SFDFA}} = K(p+1)(q+1) - \frac{1}{2}q(q-1) + 2K - 1. \quad (58)$$

As  $K$  and  $q$  are supposed very small in comparison with  $S$  and  $T$ , and  $p$  is supposed to be less than or equal 3, we have

$$\nu_{\text{RHLP-DFA}} \approx \nu_{\text{SFDFA}} < \nu_{\text{RHLP-PCA}} < \nu_{\text{DFA}} < \nu_{\text{FPCA}}.$$

Table 1 numerically illustrates the number of parameters to be estimated by each of the five methods. It can be seen that the segmentation used by SFDFA contributes to substantially reduce its number of parameters. In this example, the number of parameters required by SFDFA is about 2 times smaller than that of RHLP-PCA, 10 times smaller than that of DFA and 250 times smaller than that of FPCA.

## 7.2 Simulations

The simulations developed here are designed to the evaluation of the SFDFA approach using artificial time series of curves whose parameters are known.

### 7.2.1 Data configurations

The following two configurations were used to generate curve sequences:

– Configuration 1, which corresponds to the proposed model with  $K = 3$  linear segments per curve and a sequence of bi-dimensional factors starting from  $f_0 = (0, 0)'$ . The chosen factor loading matrix and level vector are

$$A = \begin{bmatrix} -0.022 & 0.040 \\ 0.004 & 0.001 \\ 0.204 & 0.081 \\ -0.001 & 0.000 \\ -0.091 & 0.472 \\ 0.001 & -0.002 \end{bmatrix} \quad b = \begin{bmatrix} 5.700 \\ -0.107 \\ 1.520 \\ -0.002 \\ -7.967 \\ 0.065 \end{bmatrix}.$$

The logistic regression parameters have been set to  $\alpha_1 = (180, -2)'$ ,  $\alpha_2 = (140, -1)'$ , and  $\alpha_3 = (0, 0)'$ , so that the generated curves were subject to two change points (the first one at time  $s = 40$  and the second one at time  $s = 140$ ). The variance  $\sigma^2$  has been set to 0.5<sup>2</sup>.

– Configuration 2, under which the data are generated according to the following dynamic factor model with nonlinear functional factor loadings:

$$x_{ts} = f_{t1} \sin\left(\frac{2\pi s}{S}\right) + f_{t2} \cos\left(\frac{2\pi s}{S}\right) + \varepsilon_{ts} \quad (59)$$

$$f_{tj} = f_{t-1,j} + \eta_{tj} \quad j = 1, 2, \quad (60)$$

with

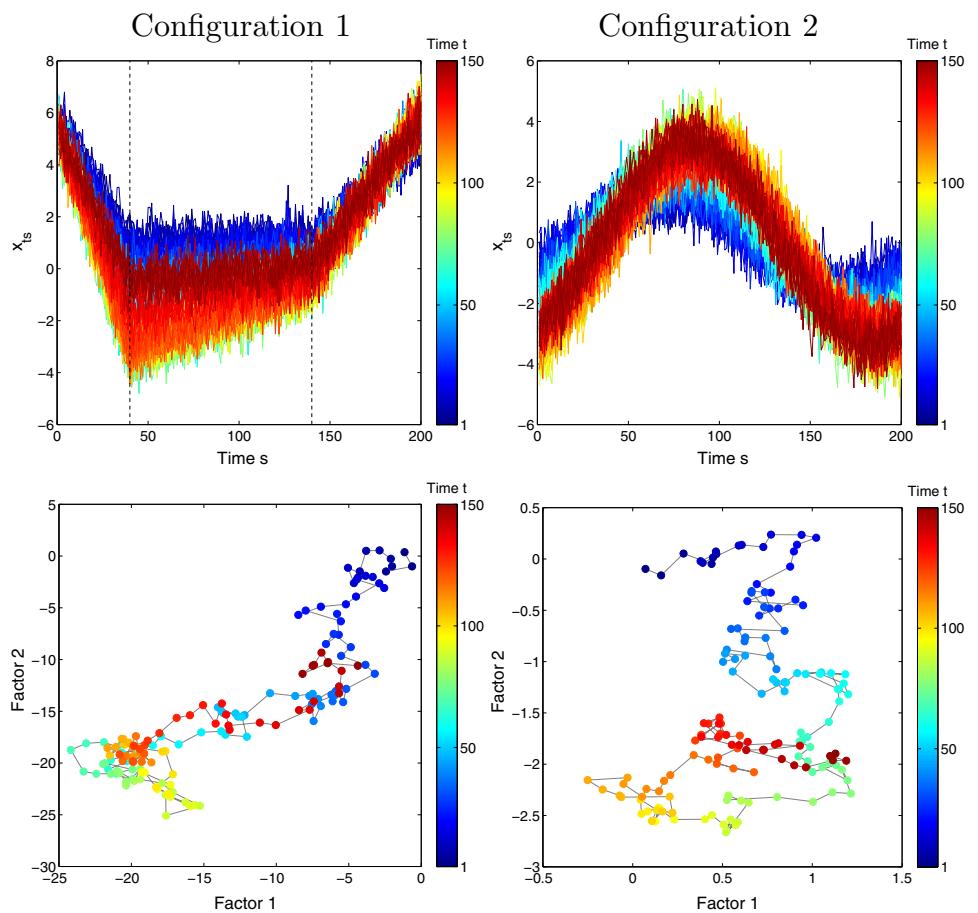
$$\varepsilon_{ts} \sim \mathcal{N}(0, \sigma^2) \quad \eta_{tj} \sim \mathcal{N}(0, v^2)$$

$$f_{01} = f_{02} = 0 \quad \sigma = 0.5 \quad v = 0.1.$$

For each configuration, 100 trajectories of  $T = 150$  factors were generated, and for each trajectory, a sequences of  $T = 150$  curves was simulated, each curve consisting of  $S = 200$  points. Each sequence was divided into two parts: the first 100 curves were used for parameter estimation and model selection (training sequences), while the last 50 curves (test sequences) served to evaluate the predictive performances of the compared models.

Figure 4 displays examples of curve sequences generated according to Configurations 1 and 2, and their associated sequence of bi-dimensional factors. For each configuration, it is clearly noticeable that the factors generated according to the random walks form trajectories into the Euclidean space. It can also be observed that the curves generated according to Configuration 1, which correspond to the SFDFA model, are subject to the change points  $s = 40$  and  $s = 140$ , while the curves generated according to Configuration 2 do not present any change point.

**Fig. 4** Examples of curve sequences generated according to Configuration 1 (left) and Configuration 2 (right), and their associated bi-dimensional factors; colors (from blue to red) indicate the curves temporal ordering. (Color figure online)

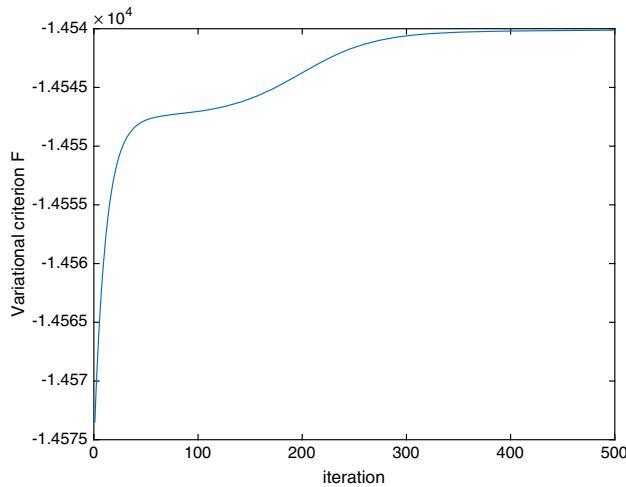


### 7.2.2 Model selection

Before evaluating the relative performances of the five compared methods, the optimal number of segments  $K$  and factor dimension  $q$  of SFDFA were computed for each training sequence, by running this method with  $K$  varying from 1 to  $K_{max} = 4$  and  $q$  varying from 1 to  $q_{max} = 4$ , and then selecting the couple  $(K, q)$  that maximizes the BIC criterion defined by Eq. (41). The polynomial order was set to  $p = 1$  (linear segments) for Configuration 1. For Configuration 2, we set  $p = 2$  (quadratic segments) due to the specific nonlinear nature of the simulated curves. We did not perform model selection for DFA, FPCA, RHLP-PCA, and RHLP-DFA. Table 2 gives, for the two configurations, the average BIC value and selection rate for each couple  $(K, q)$  over the 100 random training sequences. For Configuration 1, we obtain the smallest BIC value and highest percentage of selection for  $(K, q) = (3, 2)$  which is the true couple used for simulations. Although the curve sequences generated according to Configuration 2 are not naturally made of segments, it can be seen that the couple  $(K, q) = (3, 2)$  is selected in 72 % of cases and the couple  $(K, q) = (2, 2)$  is selected in 28 % of cases, a variability that can be attributed to the

**Table 2** Average BIC value and selection rate for couples  $(K, q)$

$K$	$q$	Configuration 1		Configuration 2	
		Select. rate (%)	BIC	Select. rate (%)	BIC
1	1	0	72088.82	0	43715.92
	2	0	72167.29	0	43063.66
	3	0	72235.94	0	43139.08
	4	0	72294.65	0	43207.75
2	1	0	43386.22	0	32043.04
	2	0	43181.28	28	30335.25
	3	0	43260.94	0	30422.44
	4	0	43337.51	0	30510.08
3	1	0	30343.45	0	31467.42
	2	100	29606.45	72	29723.94
	3	0	29692.73	0	29837.76
	4	0	29776.97	0	29956.97
4	1	0	30402.11	0	31537.51
	2	0	29687.22	0	29821.72
	3	0	29789.16	0	29957.02
	4	0	29888.80	0	30086.12



**Fig. 5** Example of variational criterion obtained during the iterations of the proposed VEM algorithm

diversity of the latent factor trajectories generated according to a Gaussian random walk. Due to this variability, a part of the sequences generated according to Configuration 2 appears to be well represented by  $K = 3$  second-order polynomials, while the other part seems to be well represented by  $K = 2$  second-order polynomials. Nevertheless, the true number of factors  $q = 2$  is recovered for each Configuration.

Figure 5 displays the convergence of the variational criterion on a curve sequence simulated according to Configuration 1. As expected, the variational criterion is strictly increasing: a rapid increasing is observed during the first 30 iterations, which is followed by a slower increasing until convergence. For the analyzed sequences, convergence was generally achieved in less than 500 iterations.

### 7.2.3 Evaluation of the algorithms accuracy

Due to the dynamic nature of the proposed segmental dynamic factor model, the performances of DFA, FPCA, RHLP-PCA, RHLP-DFA, and SFDSA are compared as regards the quality of prediction. Therefore, the following two-stage procedure was adopted:

- First, the parameters of the four compared models are estimated from the training sequences of curves  $(\mathbf{x}_1, \dots, \mathbf{x}_T)$ , with  $T = 100$ . For each sequence, SFDSA, RHLP-PCA, and RHLP-DFA are run with the number of segments and number of factors obtained for SFDSA using BIC. As shown in the previous analysis,  $(K, q) = (3, 2)$  for Configuration 1, and  $(K, q) = (3, 2)$  or  $(K, q) = (2, 2)$  for Configuration 2. The DFA and FPCA methods have been run with  $q = 2$  factors.
- Then, the performances in terms of forecasting are evaluated on the remaining sequences  $(\mathbf{x}_{T+1}, \dots, \mathbf{x}_{T+v})$ , with

$v = 50$ , using the root mean squared error (RMSE) and the mean absolute percentage error (MAPE) defined by

$$\text{RMSE} = \sqrt{\frac{1}{v \times S} \sum_{t=T}^{T+v-1} \sum_{s=1}^S (x_{t+1,s} - \hat{x}_{t+1,s|t})^2}, \quad (61)$$

$$\text{MAPE} = \frac{1}{v \times S} \sum_{t=T}^{T+v-1} \sum_{s=1}^S \left| \frac{x_{t+1,s} - \hat{x}_{t+1,s|t}}{x_{t+1,s}} \right|, \quad (62)$$

where  $\hat{x}_{t+1,s|t}$  denotes the one-step-ahead prediction of  $x_{t+1,s|t}$  given the previous data  $(\mathbf{x}_1, \dots, \mathbf{x}_t)$ . It should be recalled that for SFDSA,  $\hat{x}_{t+1,s|t}$  is given by Eq. (53), where  $\mathbf{m}_t$  is computed using Algorithm 2. We start this algorithm from  $\mathbf{m}_{t-1} = \boldsymbol{\mu}_T$  and  $\mathbf{C}_{t-1} = \boldsymbol{\Sigma}_T$ . For DFA, the forecast  $\hat{x}_{t+1,s|t}$  is derived from the Forward–Backward recursions (Zuur et al. 2003; Shumway and Stoffer 2011). For the static models FPCA and RHLP-PCA, we use a naïve approach which consists in setting  $\hat{x}_{t+1,s}$  to the reconstructed value of  $x_{t,s}$ .

Table 3 reports the average RMSE and MAPE criteria and their standard deviations (in parentheses) yielded by the five compared approaches. It can be seen that SFDSA has the best performances for the two data configurations. For Configuration 2 where the curves are not naturally made of segments, poorer performances are observed for RHLP-PCA and RHLP-DFA, which can be attributed to their two-stage operating mode.

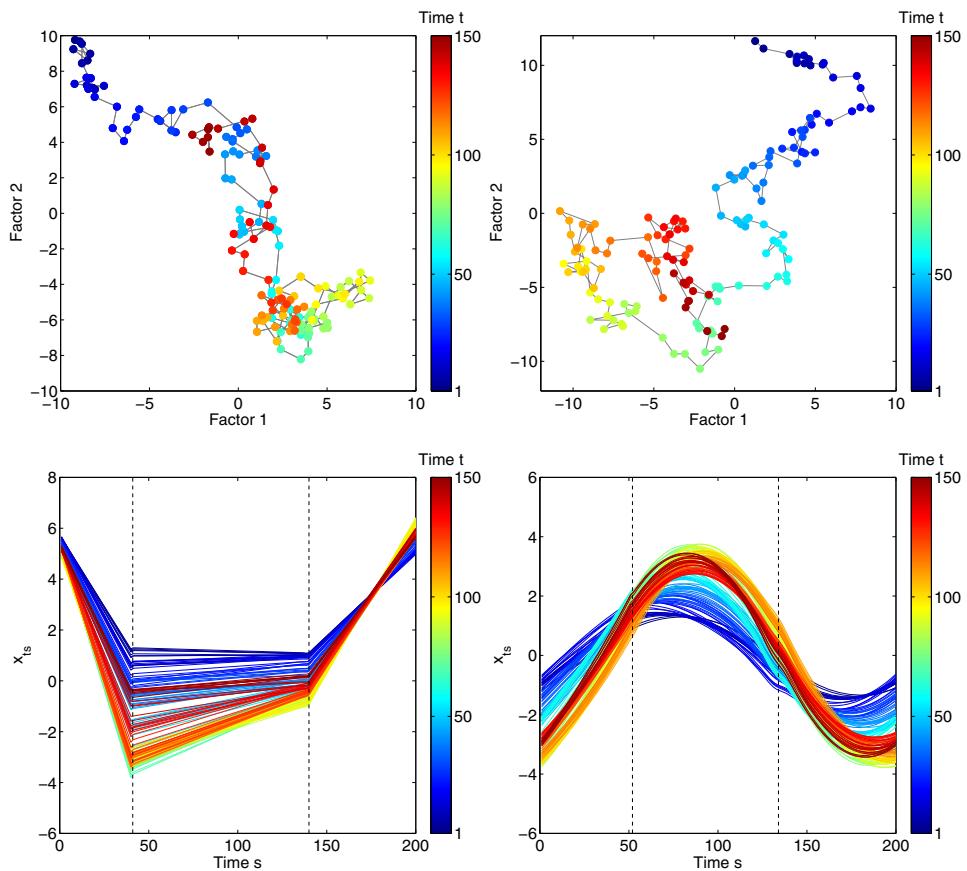
In addition to providing the error between the original sequences and its prediction, the joint segmentation produced by SFDSA, RHLP-PCA, and RHLP-DFA has been evaluated. To this end, only the data generated according to Configuration 1, for which the true segmentation were known, have been used. As the evaluation criterion, we use the percentage of time points of the set  $\{1, \dots, S\}$  incorrectly assigned to the segments. It should be recalled that the segmentation provided by SFDSA, RHLP-PCA, and RHLP-DFA are obtained by maximizing the logistic probabilities which depends only on the parameter vector  $\boldsymbol{\alpha}$ . Therefore, the predicted segmentation based on the test sequence is the same as that derived from the training sequence. For the same reasons, RHLP-PCA and RHLP-DFA provide identical segmentations. The resulting average errors are 1.13 % for SFDSA and 1.73 % for RHLP-PCA and RHLP-DFA. Although these error rates are relatively low, the best segmentation is obtained for SFDSA.

Figure 6 displays the estimated factors  $(\hat{\boldsymbol{\mu}}_t)_{t=1, \dots, 100}$  and the one-step-ahead predictions  $(\hat{\mathbf{f}}_{t+1|t})_{t=100, \dots, 149}$  obtained with SFDSA, for the data shown in Fig. 4. As it can be observed, the estimated factors are, up to a rotation, visually very similar to the true factors. The curves reconstructed from these factors are also displayed, where the dotted lines, which are derived from the logistic probabilities, represent the boundary between the estimated segments.

**Table 3** RMSE and MAPE, averaged over the 100 simulated series of curves, and their standard deviation (in parentheses) obtained with the five compared algorithms

Method	RMSE		MAPE	
	Config. 1	Config. 2	Config. 1	Config. 2
FPCA	0.54(0.021)	0.54(0.022)	1.04(0.569)	1.61(0.949)
DFA	0.54(0.020)	0.54(0.023)	1.04(0.566)	1.61(0.924)
SFDFA	0.51(0.004)	0.51(0.005)	1.01(0.541)	1.59(0.818)
RHLP-PCA	0.52(0.022)	0.67(0.126)	1.04(0.677)	1.70(0.821)
RHLP-DFA	0.54(0.024)	0.63(0.099)	1.05(0.592)	1.69(0.911)

**Fig. 6** Factors (top) and curves (bottom) estimated by SFDFA from the data of Fig. 4; the factors  $(\hat{\mu}_t)_{t=1,\dots,100}$  estimated from the training sequences are plotted together with the one-step-ahead predictions  $(\hat{f}_{t+1|t})_{t=100,\dots,149}$  which are computed from the test sequence; the dotted lines represent the boundaries between the estimated segments. (Color figure online)



**Table 4** Computation time in seconds obtained with the five compared algorithms

Method	Computation time	
	Configuration 1	Configuration 2
FPCA	0.01	0.01
DFA	21.70	19.29
SFDFA	16.70	18.68
RHLP-PCA	0.10	0.06
RHLP-DFA	22.72	16.98

#### 7.2.4 Evaluation in terms of computation time

Table 4 reports the computation time in seconds for the five compared approaches using a PC Desktop with a 3.5 GHz Intel Xeon processor. Only the parameter estimation phase is considered here. Not surprisingly, the approaches based on the principal component analysis (FPCA and RHLP-PCA) are computationally more efficient than the other approaches (DFA, SFDFA, and RHLP-DFA) whose computation time depends linearly on the number of iterations required by their associated EM algorithm. Nevertheless, SFDFA running times are globally better than those of DFA and RHLP-DFA.

### 7.3 Real data

This section is concerned with the evaluation of SFDFA using real functional time series. Although our main objective is to extract a few dynamic factors that accurately describe these data while providing a joint segmentation of the curves, the tasks of prediction and online change-point detection are also investigated. Only the methods SFDFA, RHLPPCA, and RHLPPDA, which perform segmentation and dimensionality reduction, have been compared in this part.

#### 7.3.1 Description of the data

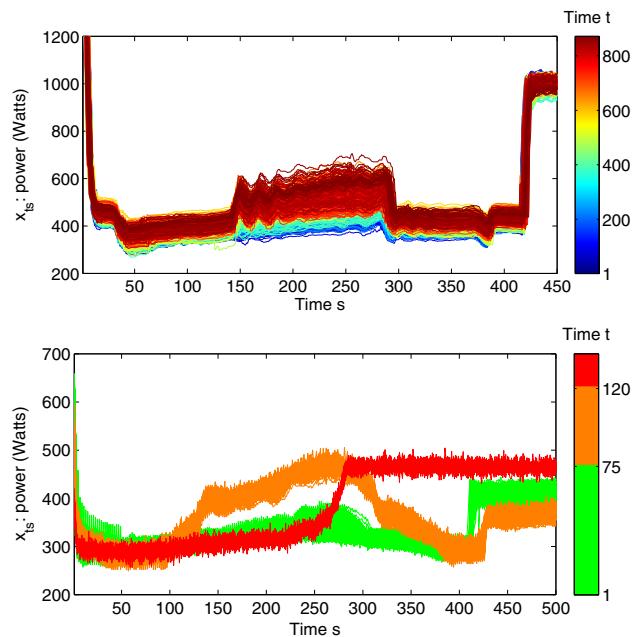
Two power consumption curve sequences acquired from two point machines are analyzed. Each sequence corresponds to consecutive rail switch operations recorded on the French high-speed lines. The curve sequence A is characterized by a low degradation, while the curve sequence B is characterized by three increasing levels of degradation: without defect (from  $t = 1$  to  $t = 75$ ), with critical defect (from  $t = 76$  to  $t = 120$ ), and failure (from  $t = 121$  to  $t = 140$ ). As already mentioned in Sect. 3.1, the peculiarity of the studied curves is their segmental aspect (five electromechanical regimes are involved during a switch operation). The characteristics of these sequences are the following:

- curve sequence A:
  - length of the curve sequence:  $T = 873$ ,
  - number of points per curve:  $S = 450$  (100 points recorded per second).
- curve sequence B:
  - length of the curve sequence:  $T = 140$ ,
  - number of points per curve:  $S = 552$  (100 points recorded per second).

These functional time series are displayed on Fig. 7. For sequence B, the three colors used (green, red, blue) corresponds to the three classes of degradation.

#### 7.3.2 Low-dimensional representation and curve segmentation

Before comparing the low-dimensional representation obtained with SFDFA, RHLPPCA, and RHLPPDA for curve sequences A and B, the optimal number of factors  $q$  has been selected only for the SFDFA approach. To this end, the number of segments and the polynomial order has, respectively, been set to  $K = 5$  and  $p = 3$ . As indicated in Sect. 3.2, these numbers correspond to the five regimes involved during a switch operation and to the order found by the experts to be well suited to the regimes involved in the power consumption curves. The proposed variational EM algorithm has



**Fig. 7** Railway power consumption curve sequences A (top) and B (bottom); for sequence B, the three colors correspond to three classes of degradation; they indicate change points at times  $t = 76$  and  $t = 121$ . (Color figure online)

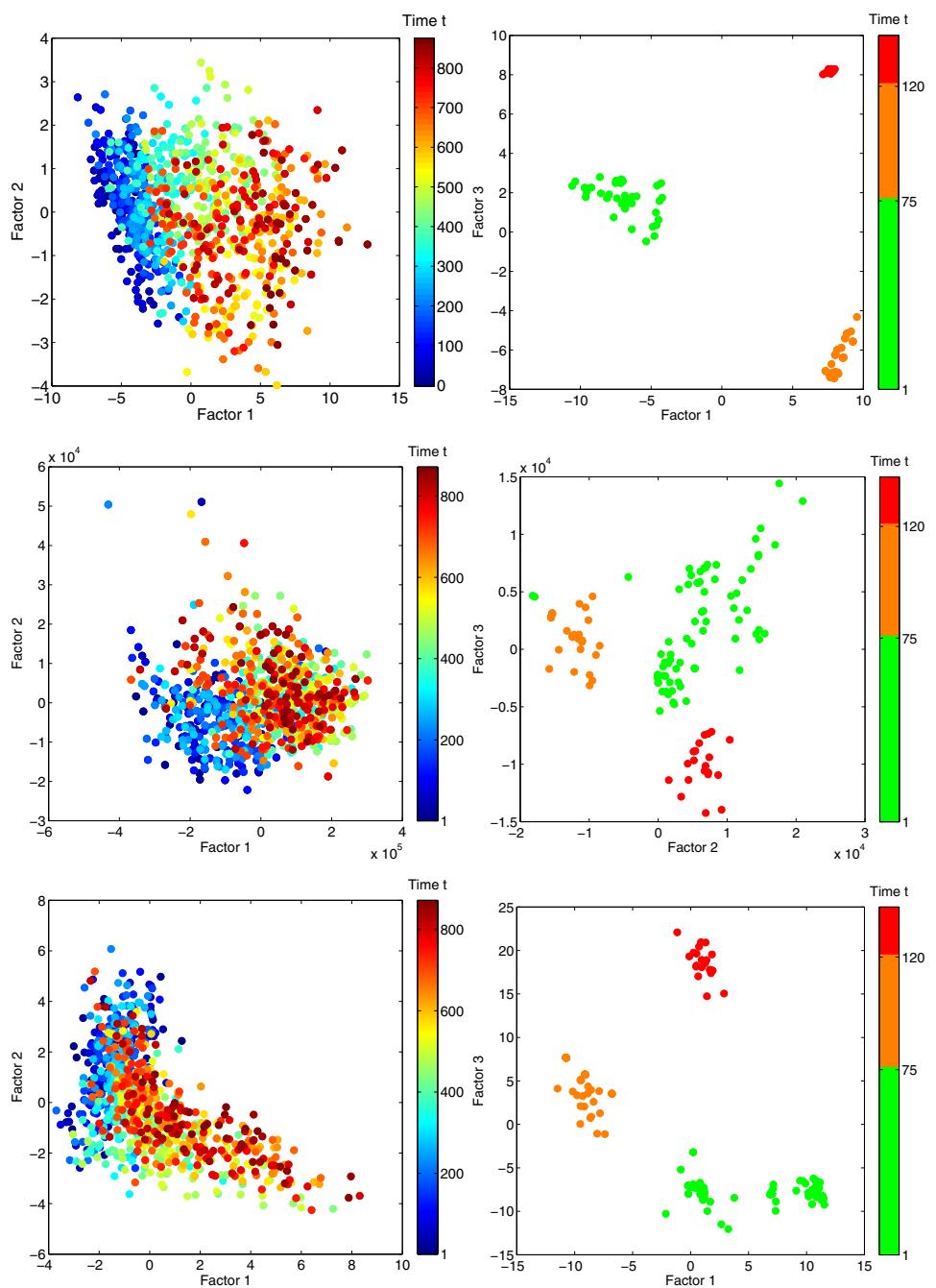
**Table 5** BIC criterion obtained with SFDFA for the two curve sequences

Factors	BIC	
	Sequence A	Sequence B
1	$3.70 \times 10^6$	$6.83 \times 10^5$
2	$3.39 \times 10^6$	$5.90 \times 10^5$
3	$3.52 \times 10^6$	$5.77 \times 10^5$
4	$3.51 \times 10^6$	$5.70 \times 10^5$

therefore been run on sequences A and B for  $q$  varying from 1 to 4, and then the value of  $q$  which minimizes the BIC criterion has been chosen. According to the BIC values reported on Table 5, we retain  $q = 2$  factors for sequence A and  $q = 4$  factors for sequence B. We therefore applied SFDFA, RHLPPCA, and RHLPPDA with ( $K = 5$ ,  $p = 3$ ,  $q = 2$ ) for sequence A and ( $K = 5$ ,  $p = 3$ ,  $q = 4$ ) for sequence B.

The Bi-dimensional plots of Fig. 8 show the factors estimated by the three compared approaches for the two sequences. Since  $q = 4$  factors have been estimated for sequence B, we select the first and third factors for SFDFA and RHLPPDA, and the second and third factors for RHLPPCA. These couples of factors visually provided the best representation in terms of classes separation. For curve sequence A, an evolution of blue points toward red points is observed onto the factors plots. This corresponds to a global growing tendency of the power consumption, which was attributed by the experts in the railway domain to an incipient

**Fig. 8** Bi-dimensional representation of factors estimated by SFdfa (*top*), RHLP-PCA (*middle*), and RHLP-DFA (*bottom*) for the sequences A (*left*) and B (*right*). (Color figure online)

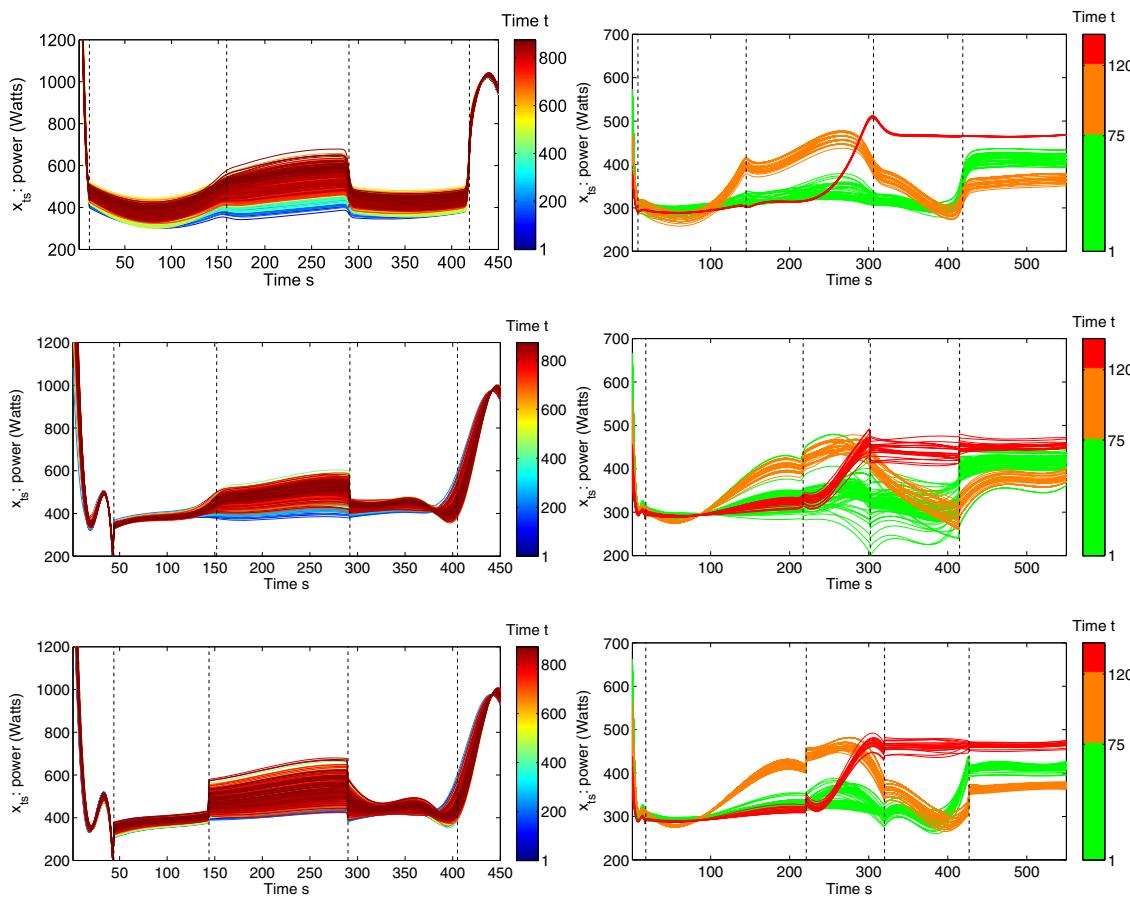


lubrication defect. For curve sequence B, the representation provided by SFdfa shows a better discrimination of classes, than that of its competitors RHLP-PCA and RHLP-DFA.

Figure 9 displays the curves reconstructed by the three methods, where the dotted lines, which are derived from the logistic probabilities, represent the boundary between the estimated segments. It was found that the change points estimated by SFdfa for sequences A and B better reflect the physical phases involved in switch operations. In particular, SFdfa gives a better estimation of the first, second, and third change points.

### 7.3.3 Predictive accuracy

The predictive performances of SFdfa, RHLP-PCA, and RHLP-DFA have also been compared using the RMSE and MAPE criteria defined by Eqs. (61) and (62). These criteria have been calculated over the last third of sequences A and B, which corresponds to ( $T = 582, v = 290$ ) for sequence A and ( $T = 94, v = 46$ ) for sequence B. They are based on the parameters and factors estimated by SFdfa, RHLP-PCA, and RHLP-DFA with ( $K = 5, p = 3, q = 2$ ) for sequence A and ( $K = 5, p = 3, q = 4$ ) for sequence B, as previously



**Fig. 9** Curves estimated by SFDFA (top), RHLP-PCA (middle), and RHLP-DFA (bottom) for the sequences A (left) and B (right). (Color figure online)

**Table 6** Predictive RMSE and MAPE criteria obtained for the real data using SFDFA, RHLP-PCA, and RHLP-DFA

Method	RMSE		MAPE	
	Seq. A	Seq. B	Seq. A	Seq. B
SFDFA	7.01	46.84	1.66	6.90
RHLP-PCA	10.05	71.92	2.38	10.96
RHLP-DFA	10.05	68.70	2.17	9.83

shown. As reported in Table 6, the best performances are obtained for the SFDA approach. As for the descriptive study, the RHLP-PCA approach performs poorly, which can be attributed to the fact that it does not take into account the dynamic between curves.

#### 7.3.4 Online defect detection

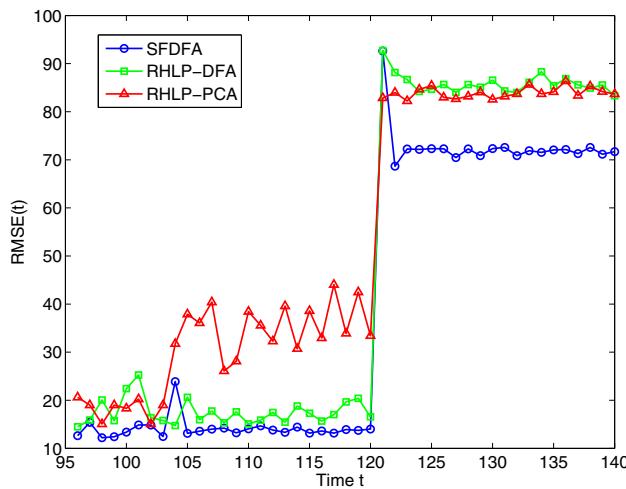
The task of online change-point detection has also been investigated using sequence B which includes three levels of degradation. For this purpose, SFDA, RHLP-PCA, and RHLP-DFA have first been estimated on the initial two-thirds

of sequence B ( $t = 1, \dots, 94$ ) with ( $K = 5, p = 3, q = 4$ ), and then the instantaneous forecasting error

$$\text{RMSE}(t) = \sqrt{\frac{1}{S} \sum_{s=1}^S (x_{t,s} - \hat{x}_{t,s|t-1})^2} \quad (63)$$

has been computed online using the last third of the sequence ( $t = 95, \dots, 140$ ). It should be noticed that the last third of sequence B only contains one change point ( $t^* = 121$ ). Figure 10 displays the  $\text{RMSE}(t)$  criterion obtained with the three compared approaches. For each of the three methods, a jump is observed at time  $t = 121$ . This one can easily be detected using an adequate threshold. As it can also be seen, the lowest errors are obtained with SFDA before and after the change point.

It should also be pointed out that a jump in the temporal evolution of  $\text{RMSE}(t)$  alerts that the model is no more adapted to the data. In this case, a re-estimation of the model parameters using more recent data is required.



**Fig. 10** Forecasting error—RMSE( $t$ )—as a function of time  $t$ . (Color figure online)

## 8 Conclusion

A dynamic factor model was introduced in this paper for modeling, through a few factors, specific time series of curves where each curve has the peculiarity of being subject to changes in regime. This dynamic regression model can be formulated as a state-space model involving discrete latent variables (to represent the regimes within curves) and continuous latent variables (to represent the dynamic across the curves). Compared to other factor analytic approaches such as dynamic factor analysis and functional principal component analysis, the proposed approach, called SF DFA, requires much less parameters due to its latent segmentation.

Since the derivation of an exact EM algorithm maximizing the likelihood is computationally intractable, we propose a variational EM algorithm maximizing a lower bound of the log-likelihood. The E-step of the latter algorithm requires Forward–Backward recursions to estimate the latent common factors, the estimation of the parameters governing the joint segmentation (in the M-step) being performed through the IRLS algorithm.

The experimental study conducted on simulated data and real time series of curves has shown encouraging results in terms of functional time series modeling. The extracted factors provide a descriptive tool for visualizing more easily the dynamic behind the curves. The experimental study has also revealed that the proposed model is exploitable for short-term curve forecasting. In this perspective, the using of more general models such as ARMA may be useful to more accurately describe the latent factor dynamic.

Although the present research was focused on curves observed over the same time grid, it can easily be extended to curves with potentially different temporal discretization. This extension simply requires replacing covariates  $\mathbf{u}_s$  ( $s = 1, \dots, S$ ) by curve-specific covariates  $\mathbf{u}_{ts}$  ( $s$  =

$1, \dots, S_t$ ,  $t = 1, \dots, T$ ),  $S_t$  being the number of points of the curve  $\mathbf{x}_t$ .

**Acknowledgements** The authors wish to thank M. Marc Antoni of SNCF for the data he provided and for the support provided.

## Appendix 1: Derivation of the variational criterion $\mathcal{F}$

The variational criterion to be maximized is defined by

$$\mathcal{F}(Q, \boldsymbol{\theta}) = \mathbb{E}_Q [\log P(\mathbf{x}, \mathbf{f}, \mathbf{z}; \boldsymbol{\theta})] - \mathbb{E}_Q [\log Q(\mathbf{z}, \mathbf{f})], \quad (64)$$

where the log-likelihood  $\log P(\mathbf{x}, \mathbf{z}, \mathbf{f}; \boldsymbol{\theta})$  associated to the complete data is given by

$$\begin{aligned} \log P(\mathbf{x}, \mathbf{z}, \mathbf{f}; \boldsymbol{\theta}) &= \log \left( P(\mathbf{z}; \boldsymbol{\theta}) P(\mathbf{f}; \boldsymbol{\theta}) P(\mathbf{x} | \mathbf{z}, \mathbf{f}; \boldsymbol{\theta}) \right) \\ &= \sum_{t,s,k} z_{tsk} \log \pi_k(s; \boldsymbol{\alpha}) + \sum_{t,s,k} z_{tsk} \\ &\quad \times \log \mathcal{N}(x_{ts}; \mathbf{u}'_s (\mathbf{A}_k \mathbf{f}_t + \mathbf{b}_k), \sigma^2) \\ &\quad + \sum_t \log \mathcal{N}(\mathbf{f}_t; \mathbf{f}_{t-1}, \mathbf{I}_q), \end{aligned} \quad (65)$$

and the logarithm of the distribution  $Q$  is given by

$$\begin{aligned} \log Q(\mathbf{z}, \mathbf{f}) &= \log \left( \prod_{t,s,k} (\tau_{tsk})^{z_{tsk}} \prod_t \mathcal{N}(\mathbf{f}_t; \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t) \right) \\ &= \sum_{t,s,k} z_{tsk} \log \tau_{tsk} + \sum_t \log \mathcal{N}(\mathbf{f}_t; \mathbf{f}_{t-1}, \mathbf{I}_q). \end{aligned} \quad (66)$$

By observing that

$$\begin{aligned} \mathbb{E}_Q \left[ (x_{ts} - \mathbf{u}'_s (\mathbf{A}_k \mathbf{f}_t + \mathbf{b}_k))^2 \right] \\ = (x_{ts} - \mathbf{u}'_s (\mathbf{A}_k \boldsymbol{\mu}_t + \mathbf{b}_k))^2 + \mathbf{u}'_s \mathbf{A}_k \boldsymbol{\Sigma}_t \mathbf{A}'_k \mathbf{u}_s \quad \forall t, s, k, \end{aligned}$$

and

$$\begin{aligned} \mathbb{E}_Q \left[ (\mathbf{f}_t - \mathbf{f}_{t-1})' (\mathbf{f}_t - \mathbf{f}_{t-1}) \right] \\ = (\boldsymbol{\mu}_t - \boldsymbol{\mu}_{t-1})' (\boldsymbol{\mu}_t - \boldsymbol{\mu}_{t-1}) + \text{Tr}(\boldsymbol{\Sigma}_t) + \text{Tr}(\boldsymbol{\Sigma}_{t-1}) \quad \forall t, \end{aligned}$$

it can be deduced that

$$\begin{aligned} \mathcal{F}(\boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\theta}) &= \sum_{t,s,k} \tau_{tsk} \mathcal{A}_k(x_{ts}, \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t, \boldsymbol{\theta}) \\ &\quad + \sum_t \mathcal{B}(\boldsymbol{\mu}_{t-1}, \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_{t-1}, \boldsymbol{\Sigma}_t) \\ &\quad - \sum_{t,s,k} \tau_{tsk} \log \tau_{tsk} + \frac{1}{2} \sum_t \log \det(\boldsymbol{\Sigma}_t), \end{aligned}$$

with

$$\begin{aligned}\mathcal{A}_k(x_{ts}, \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t, \boldsymbol{\theta}) \\ = \log \left( \pi_k(s; \boldsymbol{\alpha}) \mathcal{N}(x_{ts}; \mathbf{u}'_s (\mathbf{A}_k \boldsymbol{\mu}_t + \mathbf{b}_k), \sigma^2) \right) \\ - \frac{1}{2\sigma^2} (\mathbf{u}'_s \mathbf{A}_k \boldsymbol{\Sigma}_t \mathbf{A}'_k \mathbf{u}_s) \quad \forall t, s, k,\end{aligned}$$

and

$$\begin{aligned}\mathcal{B}(\boldsymbol{\mu}_{t-1}, \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_{t-1}, \boldsymbol{\Sigma}_t) = \log \mathcal{N}(\boldsymbol{\mu}_t; \boldsymbol{\mu}_{t-1}, \mathbf{I}_q) \\ - \frac{1}{2} \left( \text{Tr}(\boldsymbol{\Sigma}_t) + \text{Tr}(\boldsymbol{\Sigma}_{t-1}) \right) \quad \forall t,\end{aligned}$$

where the convention  $\boldsymbol{\mu}_0 = \mathbf{f}_0$  and  $\boldsymbol{\Sigma}_0 = \mathbf{0}$  is used.

## Appendix 2: E-step: derivation of the Forward–Backward recursions

By canceling the partial derivatives of  $\mathcal{F}$  with respect to each of the vectors  $\boldsymbol{\mu}_t$ , it can be verified that the sequence  $\boldsymbol{\mu}^{(c+1)}$  defined by Eq. (27) is the solution of the following system of linear equations:

$$\left\{ \begin{array}{l} \left( \boldsymbol{\Sigma}_1^{(c+1)} \right)^{-1} \boldsymbol{\mu}_1^{(c+1)} - \boldsymbol{\mu}_2^{(c+1)} = \mathbf{d}_1^{(c+1)} + \mathbf{f}_0^{(c)} \\ \left( \boldsymbol{\Sigma}_2^{(c+1)} \right)^{-1} \boldsymbol{\mu}_2^{(c+1)} - \boldsymbol{\mu}_1^{(c+1)} - \boldsymbol{\mu}_3^{(c+1)} = \mathbf{d}_2^{(c+1)} \\ \vdots \\ \left( \boldsymbol{\Sigma}_{T-1}^{(c+1)} \right)^{-1} \boldsymbol{\mu}_{T-1}^{(c+1)} - \boldsymbol{\mu}_{T-2}^{(c+1)} - \boldsymbol{\mu}_T^{(c+1)} = \mathbf{d}_{T-1}^{(c+1)} \\ \left( \boldsymbol{\Sigma}_T^{(c+1)} \right)^{-1} \boldsymbol{\mu}_T^{(c+1)} - \boldsymbol{\mu}_{T-1}^{(c+1)} = \mathbf{d}_T^{(c+1)}, \end{array} \right. \quad (67)$$

with

$$\begin{aligned}\mathbf{d}_t^{(c+1)} &= \frac{1}{\sigma^{2(c)}} \sum_{s,k} \tau_{tsk}^{(c+1)} \mathbf{A}_k^{(c)'} \mathbf{u}_s (x_{ts} - \mathbf{u}'_s \mathbf{b}_k^{(c)}), \\ \mathbf{D}_t^{(c+1)} &= \frac{1}{\sigma^{2(c)}} \sum_{s,k} \tau_{tsk}^{(c+1)} \mathbf{A}_k^{(c)'} \mathbf{u}_s \mathbf{u}'_s \mathbf{A}_k^{(c)}.\end{aligned}$$

Solving such a linear system requires the inversion of a block-tridiagonal matrix (Varah 1972) whose diagonal terms are the matrices  $(\boldsymbol{\Sigma}_t^{(c+1)})^{-1}$  and off-diagonal terms are equal to the matrix  $-\mathbf{I}_q$ . The solution commonly adopted is based on LU matrix decomposition (Thomas 1949), which results in the following Forward–Backward recursions:

- Forward: starting from  $\tilde{\boldsymbol{\Sigma}}_0 = \mathbf{0}$  and  $\tilde{\boldsymbol{\mu}}_0 = \mathbf{f}_0^{(c)}$ , compute, for  $t = 1, \dots, T$ ,

$$\begin{aligned}\tilde{\boldsymbol{\Sigma}}_t &= \left[ \mathbf{D}_t^{(c+1)} + (\mathbf{I}_q + \tilde{\boldsymbol{\Sigma}}_{t-1})^{-1} \right]^{-1}, \\ \tilde{\boldsymbol{\mu}}_t &= \tilde{\boldsymbol{\mu}}_{t-1} + \tilde{\boldsymbol{\Sigma}}_t \left( \mathbf{d}_t^{(c+1)} - \mathbf{D}_t^{(c+1)} \tilde{\boldsymbol{\mu}}_{t-1} \right),\end{aligned}$$

with

$$\begin{aligned}\mathbf{d}_t^{(c+1)} &= \frac{1}{\sigma^{2(c)}} \sum_{s,k} \tau_{tsk}^{(c+1)} \mathbf{A}_k^{(c)'} \mathbf{u}_s (x_{ts} - \mathbf{u}'_s \mathbf{b}_k^{(c)}), \\ \mathbf{D}_t^{(c+1)} &= \frac{1}{\sigma^{2(c)}} \sum_{s,k} \tau_{tsk}^{(c+1)} \mathbf{A}_k^{(c)'} \mathbf{u}_s \mathbf{u}'_s \mathbf{A}_k^{(c)}.\end{aligned}$$

- Backward: starting from  $\boldsymbol{\mu}_T^{(c+1)} = \tilde{\boldsymbol{\mu}}_T$ , compute retrospectively, for  $t = T-1, \dots, 1$ ,

$$\boldsymbol{\mu}_t^{(c+1)} = \tilde{\boldsymbol{\mu}}_t + \mathbf{J}_t \left( \boldsymbol{\mu}_{t+1}^{(c+1)} - \tilde{\boldsymbol{\mu}}_t \right),$$

$$\text{with } \mathbf{J}_t = \tilde{\boldsymbol{\Sigma}}_t (\mathbf{I}_q + \tilde{\boldsymbol{\Sigma}}_t)^{-1}.$$

## Appendix 3: M-step

### Maximization of $\mathcal{F}$ with respect to $\mathbf{f}_0$

By canceling the partial derivative of  $\mathcal{F}_1$  with respect to  $\mathbf{f}_0$ , we get

$$\mathbf{f}_0^{(c+1)} = \boldsymbol{\mu}_1^{(c+1)}. \quad (68)$$

### Maximization of $\mathcal{F}$ with respect to $\boldsymbol{\alpha}$

Maximizing  $\mathcal{F}_1$  with respect to  $\boldsymbol{\alpha}$  is equivalent to solving the logistic regression problem

$$\boldsymbol{\alpha}^{(c+1)} = \arg \max_{\boldsymbol{\alpha}} \sum_{t,s,k} \tau_{tsk}^{(c+1)} \log \pi_k(s; \boldsymbol{\alpha}). \quad (69)$$

This maximization is therefore solved through the well-known Iteratively Reweighted Least Squares (IRLS) algorithm (Green 1984) which is equivalent to a Newton–Raphson algorithm. In order to describe this algorithm, let

$$\mathcal{G}(\boldsymbol{\alpha}) = \sum_{t,s,k} \tau_{tsk}^{(c+1)} \log \pi_k(s; \boldsymbol{\alpha})$$

denote the quantity to be maximized. As shown in (Chamroukhi et al. 2009a,b), the first and second derivatives of  $\mathcal{G}$  can be written as

$$\mathbf{g}(\boldsymbol{\alpha}) = (\mathbf{g}_k(\boldsymbol{\alpha}))_{1 \leq k \leq K-1}, \quad (70)$$

$$\mathbf{H}(\boldsymbol{\alpha}) = (\mathbf{H}_{kl}(\boldsymbol{\alpha}))_{1 \leq k, l \leq K-1}, \quad (71)$$

with

$$\mathbf{g}_k(\boldsymbol{\alpha}) = \frac{\partial \mathcal{G}(\boldsymbol{\alpha})}{\partial \alpha_k} = \sum_{t,s} (\tau_{tsk}^{(c+1)} - \pi_k(s; \boldsymbol{\alpha})) \mathbf{u}_s, \quad (72)$$

$$\begin{aligned} \mathbf{H}_{k\ell}(\boldsymbol{\alpha}) &= \frac{\partial^2 \mathcal{G}(\boldsymbol{\alpha})}{\partial \boldsymbol{\alpha}_k \partial \boldsymbol{\alpha}'_\ell} \\ &= -T \sum_s \left[ \pi_k(s; \boldsymbol{\alpha}) (\delta_{k\ell} - \pi_\ell(s; \boldsymbol{\alpha})) \right] \mathbf{u}_s \mathbf{u}'_s, \end{aligned} \quad (73)$$

where  $\delta_{k\ell} = 1$  if  $k = \ell$  and 0 otherwise.

Using these derivatives, the pseudo-code of the IRLS procedure can therefore be summarized by Algorithm 3.

### Algorithm 3: IRLS

**Input:** posterior probabilities  $(\tau_{tsk}^{(c+1)})$ , threshold  $\varepsilon$

$\boldsymbol{\alpha} \leftarrow \mathbf{0}$

**repeat**

$$\begin{cases} \mathbf{g}_k \leftarrow \sum_{t,s} (\tau_{tsk}^{(c+1)} - \pi_k(s; \boldsymbol{\alpha})) \mathbf{u}_s \\ \mathbf{H}_{k\ell} \leftarrow -T \sum_s \pi_k(s; \boldsymbol{\alpha}) (\delta_{k\ell} - \pi_\ell(s; \boldsymbol{\alpha})) \mathbf{u}_s \mathbf{u}'_s \\ \boldsymbol{\alpha} \leftarrow \boldsymbol{\alpha} - \mathbf{H}^{-1} \mathbf{g} \end{cases}$$

**until**  $\|\mathbf{g}\| < \varepsilon$ ;

$$\boldsymbol{\alpha}^{(c+1)} \leftarrow \boldsymbol{\alpha}$$

**Output:** logistic regression parameter  $\boldsymbol{\alpha}^{(c+1)}$

### Maximization of $\mathcal{F}$ with respect to $(A_k, b_k)$

This maximization problem cannot be solved in a closed form but conditional maximizations, with respect to  $\mathbf{b}_k$  and  $\mathbf{A}_k$ , can be performed analytically. This kind of approach, which just increases  $\mathcal{F}_1$ , results in a Generalized EM algorithm (McLachlan and Krishnan 2008), whose convergence properties are the same as those of the EM algorithm. The resulting updating formulae are the following:

$$\begin{aligned} \mathbf{b}_k^{(c+1)} &= \left( \sum_{t,s} \tau_{tsk}^{(c+1)} \mathbf{u}_s \mathbf{u}'_s \right)^{-1} \\ &\times \sum_{t,s} \tau_{tsk}^{(c+1)} \mathbf{u}_s \left( x_{ts} - \mathbf{u}'_s \mathbf{A}_k^{(c)} \boldsymbol{\mu}_t^{(c+1)} \right), \end{aligned} \quad (74)$$

$$\begin{aligned} \text{vec} \left( \mathbf{A}_k^{(c+1)} \right) \\ = \left( \sum_t \tau_{tsk}^{(c+1)} \left( \boldsymbol{\Sigma}_t^{(c+1)} + \boldsymbol{\mu}_t^{(c+1)} \boldsymbol{\mu}_t^{(c+1)'} \right) \otimes (\mathbf{u}_s \mathbf{u}'_s) \right)^{-1} \\ \times \left( \sum_{t,s} \tau_{tsk}^{(c+1)} \mathbf{u}_s (x_{ts} - \mathbf{u}'_s \mathbf{b}_k^{(c+1)}) \boldsymbol{\mu}_t^{(c+1)'} \right). \end{aligned} \quad (75)$$

where  $\otimes$  is the Kronecker product and  $\text{vec}(\cdot)$  is the operator that concatenates the columns of a matrix into a vector. Once the matrices  $\mathbf{A}_k$  ( $k = 1, \dots, K$ ) have been estimated, the orthogonalization of the complete matrix  $\mathbf{A}$  is done as follows: we first perform the eigenvalue decomposition of the symmetric positive definite matrix  $\mathbf{A}^{(c+1)'} \mathbf{A}^{(c+1)}$ ; let  $\mathbf{P}^{(c+1)}$

denote the resulting orthogonal matrix of eigenvectors; then we set

$$\mathbf{A}^{(c+1)} \leftarrow \mathbf{A}^{(c+1)} \mathbf{P}^{(c+1)} \quad (76)$$

$$\mathbf{f}_0^{(c+1)} \leftarrow \mathbf{P}^{(c+1)'} \mathbf{f}_0^{(c+1)} \quad (77)$$

$$\boldsymbol{\mu}_t^{(c+1)} \leftarrow \mathbf{P}^{(c+1)'} \boldsymbol{\mu}_t^{(c+1)} \quad \forall t. \quad (78)$$

It can easily be verified that the resulting matrix, which satisfies the condition  $\mathbf{A}^{(c+1)'} \mathbf{A}^{(c+1)} = \mathbf{I}_q$ , is diagonal.

### Maximization of $\mathcal{F}$ with respect to $\sigma^2$

Maximizing  $\mathcal{F}_1$  with respect to the observations variance  $\sigma^2$  leads to

$$\begin{aligned} \sigma^{2(c+1)} &= \frac{1}{TS} \sum_{t,s,k} \tau_{tsk}^{(c+1)} \times \left[ \mathbf{u}'_s \mathbf{A}_k^{(c+1)} \boldsymbol{\Sigma}_t^{(c+1)} \mathbf{A}_k^{(c+1)'} \mathbf{u}_s \right. \\ &\quad \left. + \left( x_{ts} - \mathbf{u}'_s (\mathbf{A}_k^{(c+1)} \boldsymbol{\mu}_t^{(c+1)} + \mathbf{b}_k^{(c+1)}) \right)^2 \right]. \end{aligned} \quad (79)$$

### References

- Baek, J., McLachlan, G.J., Flack, L.: Mixtures of factor analyzers with common factor loadings: applications to the clustering and visualization of high-dimensional data. *IEEE Trans. Pattern Anal. Mach. Intell.* **32**, 1298–1309 (2010)
- Bai, J., Wang, P.: Identification and bayesian estimation of dynamic factor models. *J. Bus. Econ. Stat.* **33**(2), 221–240 (2015)
- Beal, M.J.: Variational algorithms for approximate bayesian inference. PhD thesis, Gatsby Computational Neuroscience Unit, University College London (2003)
- Besse, P.C., Cardot, H., Stephenson, D.B.: Autoregressive forecasting of some functional climatic variations. *Scand. J. Stat.* **27**, 673–687 (2000)
- Blei, D.M., Lafferty, J.D.: Dynamic topic models. In: Proceedings of the 23rd International Conference on Machine Learning, ICML '06, pp. 113–120. ACM, New York (2006)
- Chamroukhi, F., Samé, A., Govaert, G., Aknini, P.: A regression model with a hidden logistic process for feature extraction from time serie. In: IEEE International Joint Conference on Neural Networks (IJCNN 2009), Atlanta (2009a)
- Chamroukhi, F., Samé, A., Govaert, G., Aknini, P.: Time series modeling by a regression approach based on a latent process. *Neural Netw.* **22**, 593–602 (2009b)
- Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc. Ser. B* **39**, 1–38 (1977)
- Diebold, F.X., Li, C.: Forecasting the term structure of government bond yields. *J. Econom.* **130**(2), 337–364 (2006)
- Fearnhead, P., Liu, Z.: On-line inference for multiple changepoint problems. *J. R. Stat. Soc. B* **69**, 589–605 (2007)
- Ferraty, F., Vieu, P.: Nonparametric Functional Data Analysis. Springer, New York (2006)
- Gareth, J., Hastie, T., Sugar, C.: Principal component models for sparse functional data. *Biometrika* **87**, 587–602 (2000)
- Garnett, R., Osborne, M.A., Reece, S., Rogers, A., Roberts, S.J.: Sequential bayesian prediction in the presence of changepoints and faults. *Comput. J.* **53**(9), 1430–1446 (2010)

- Geweke, J.F.: The dynamic factor analysis of economic time series models. In: Aigner, D.J., Goldberger, A.S. (eds.) *Latent Variables in Socio-economic Models*, pp. 365–383. North-Holland, Amsterdam (1977)
- Green, P.: Iteratively reweighted least squares for maximum likelihood estimation, and some robust and resistant alternatives. *J. R. Stat. Soc. Ser. B* **46**(2), 149–192 (1984)
- Harvey, A.C.: *Forecasting Structural Time Series Models and the Kalman Filter*. Cambridge University Press, Cambridge (1989)
- Hays, S., Shen, H., Huang, J.Z.: Functional dynamic factor models with application to yield curve forecasting. *Ann. Appl. Stat.* **6**(3), 870–894 (2012)
- Hyndman, R.J., Ullah, S.: Robust forecasting of mortality and fertility rates: a functional data approach. *Comput. Statist. Data Anal.* **51**(10), 4942–4956 (2007)
- Jiang, W., Tanner, M.A.: On the identifiability of mixtures-of-experts. *Neural Netw.* **12**(9), 1253–1258 (1999)
- Jolliffe, I.T.: *Principal Component Analysis*, 2nd edn. Springer, New York (2002)
- Jordan, M.I., Ghahramani, Z., Jaakkola, T.S., Saul, L.K.: An introduction to variational methods for graphical models. *Mach. Learn.* **37**(2), 183–233 (1999)
- Kalman, R.E.: A new approach to linear filtering and prediction problems. *J. Basic Eng.* **82**(1), 35–45 (1960)
- Lawley, D.N.: The estimation of factor loadings by the method of maximum likelihood. *Proc. R. Soc. Edinb.* **60**(01), 64–82 (1940)
- McLachlan, G.J., Krishnan, T.: *The EM Algorithm and Extensions*. Wiley, New York (2008)
- McLachlan, G.J., Peel, D., Bean, R.W.: Modelling high-dimensional data by mixtures of factor analyzers. *Computat. Stat. Data Anal.* **41**(3–4), 379–388 (2003)
- Molaneaar, P.C.: A dynamic factor model for the analysis of multivariate time series. *Psychometrika* **50**, 181–202 (1985)
- Neal, R.M.: Probabilistic inference using Markov chain Monte Carlo methods. Technical Report CRG-TR-93-1, Dept. of Computer Science, University of Toronto (1993)
- Neal, R., Hinton, G.: A view of the EM algorithm that justifies incremental, sparse, and other variants. In: I, J.M. (ed.) *Learning in Graphical Models*. Kluwer, Dordrecht (1998)
- Pearson, K.: On lines and planes of closest fit to systems of points in space. *Philos. Mag.* **6**(2), 559–572 (1901)
- Ramsay, J.O., Silverman, B.W.: *Functional data analysis*, 2nd edn. Springer, New York (1997)
- Saatçi, Y., Turner, R., Rasmussen, C.E.: Gaussian process change point models. In: *Proceedings of the 27th International Conference on Machine Learning*, pp. 927–934, Haifa (2010)
- Sarkka, S., Nummenmaa, A.: Recursive noise adaptive kalman filtering by variational bayesian approximations. *IEEE Trans. Autom. Control* **54**(3), 596–600 (2009)
- Shumway, R.H., Stoffer, D.S.: An approach to time series smoothing and forecasting using the EM algorithm. *J. Time Ser. Anal.* **3**, 253–264 (1982)
- Shumway, R.H., Stoffer, D.S.: *Time Series Analysis and Its Applications*. Springer, New York (2011)
- Thomas, L.H.: Elliptic problems in linear differential equations over a network. Technical report, Columbia University (1949)
- Varah, J.M.: On the solution of block-tridiagonal systems arising from certain finite-difference equations. *Math. Comput.* **26**(120), 859–868 (1972)
- Zuur, A.F., Fryer, R.J., Jolliffe, I.T., Dekker, R., Beukema, J.J.: Estimating common trends in multivariate time series using dynamic factor analysis. *Environmetrics* **14**, 665–685 (2003)