Adv Data Anal Classif
DOI 10.1007/s11634-011-0096-5

REGULAR ARTICLE

Model-based clustering and segmentation of time series with changes in regime

Allou Samé · Faicel Chamroukhi · Gérard Govaert · Patrice Aknin

Received: 26 January 2011 / Revised: 31 July 2011 / Accepted: 21 September 2011

Abstract Mixture model-based clustering, usually applied to multidimensional data, has become a popular approach in many data analysis problems, both for its good statistical properties and for the simplicity of implementation of the Expectation–Maximization (EM) algorithm. Within the context of a railway application, this paper introduces a novel mixture model for dealing with time series that are subject to changes in regime. The proposed approach, called ClustSeg, consists in modeling each cluster by a regression model in which the polynomial coefficients vary according to a discrete hidden process. In particular, this approach makes use of logistic functions to model the (smooth or abrupt) transitions between regimes. The model parameters are estimated by the maximum likelihood method solved by an EM algorithm. This approach can also be regarded as a clustering approach which operates by finding groups of time series having common changes in regime. In addition to providing a time series partition, it therefore provides a time series segmentation. The problem of selecting the optimal numbers of clusters and segments is solved by means of the Bayesian Information Criterion. The ClustSeg approach is shown to be efficient using a variety of simulated time series and real-world time series of electrical power consumption from rail switching operations.

Keywords Clustering · Time series · Change in regime · Mixture model · Regression mixture · Hidden logistic process · EM algorithm

Mathematics Subject Classification (2010) 62-07 · 62M10 · 62H30

A. Samé (✉) · F. Chamroukhi · P. Aknin
Université Paris-Est, IFSTTAR, GRETTIA, 93160 Noisy-le-Grand, France
e-mail: allou.same@ifsttar.fr

G. Govaert
Université de Technologie de Compiègne, UMR CNRS 6599,
HEUDIASYC, BP 20529, 60205 Compiègne, France
e-mail: gerard.govaert@utc.fr

Published online: 12 October 2011

1 Introduction

The application which gave rise to this study concerns the diagnosis of railway switches, that is to say the mechanisms which enable trains to change tracks at junctions. One preliminary task in the diagnostic process is identifying groups of switching operations that show similar dynamic behaviour, and this is accomplished by performing clustering on the time series of electrical power consumption, acquired during various switching operations. This kind of data is referred to in other contexts as longitudinal data (Chiou and Li 2007), signals, curves (Gaffney and Smyth 1999; Shi and Wang 2008), or functional data (Ramsay and Silverman 1997). In particular, the power consumption time series studied in this paper are subject to various shifts (see Fig. 10) as a result of successive mechanical movements of the physical components associated with the switch mechanism. In this context, the time period between two successive shifting times is called a regime.

The approach adopted in this paper is mixture model-based clustering (Banfield and Raftery 1993; Celeux and Govaert 1995), which has successfully been applied in numerous domains (McLachlan and Peel 2000), and which provides, by means of the Expectation–Maximization algorithm (Dempster et al. 1977; McLachlan and Krishnan 2008), an efficient implementation framework. Typical extensions of mixture models for time series include regression mixture models (Gaffney and Smyth 1999) and random effect regression mixture models (Gaffney and Smyth 2003; James and Sugar 2003; Ng et al. 2006; Liu and Yang 2009). More recently, Coke and Tsao (2010) proposed a specific random effect mixture model for electrical load series clustering. These approaches are based on a projection of the original time series into a space with fewer dimensions, defined by polynomial or spline basis functions. Other approaches that combine mixtures of autoregressive models and the Expectation–Maximization algorithm (Wong and Li 2000), or Autoregressive Moving Average (ARMA) models and the Expectation–Maximization algorithm (Xiong and Yeung 2004) have also been proposed. Although these approaches can be seen as an efficient way of classifying time series, all of them use only one global model (regressive or autoregressive) within each cluster.

Within the particular context of time series with changes in regime, a specific regression model has been proposed in Chamroukhi et al. (2010) to model time series with changes in regime. The model in question is a regression model in which the polynomial coefficients may vary according to a discrete hidden process, and which uses logistic functions to model the (smooth or abrupt) transitions between regimes. However, the latter model did not deal with time series clustering but only with the modeling of a set of homogeneous time series with common changes in regime. Therefore, this paper extends this concept to a more general model, named ClustSeg, applied to the clustering and segmentation of heterogeneous time series with changes in regime.

This paper is organized as follows. We first present a brief review of the regression mixture model for time series clustering. Then, we detail the ClustSeg model and its parameters estimation via the Expectation–Maximization algorithm (Dempster et al. 1977; McLachlan and Krishnan 2008). Based on simulated examples and real-world time series from an application in the railway sector, an experimental study illustrates the performance of the proposed approach.

The n time series to be classified will be denoted as $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$, where each series $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{im})$ consists of m real values observed over a deterministic prespecified time grid $\boldsymbol{t} = (t_1, \ldots, t_m)$, with $t_1 < t_2 < \cdots < t_m$. They are supposed to be the realized values of n independent and identically distributed random variables $\mathbf{X}_1, \ldots, \mathbf{X}_n$, where $\mathbf{X}_i = (X_{i1}, \ldots, X_{im})$. The unobserved class labels corresponding to the time series $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$, which are denoted as $(z_1, \ldots, z_n)$, with $z_i \in \{1, \ldots, K\}$, are taken to constitute instances of n independent random variables $(Z_1, \ldots, Z_n)$ with parent variable Z.

2 Regression mixture model for time series clustering

This section briefly recalls the regression mixture model called here the RegMix model, as formulated by Gaffney and Smyth (1999), in the context of times series clustering.

2.1 Definition of the regression mixture model

Unlike standard vector-based mixture models, the density of each component of the regression mixture is represented by a polynomial prototype series parameterized by a vector of regression coefficients and a noise variance. These prototype series or functions represent the class conditional expectations of variables $\mathbf{X}_i$. The regression mixture model therefore assumes that each series $\mathbf{X}_i$, given the time grid $\boldsymbol{t}$, is distributed according the density

$$f(\boldsymbol{x}_i|\boldsymbol{t};\boldsymbol{\theta}) = \sum_{k=1}^{K} \pi_k \mathcal{N}_m(\boldsymbol{x}_i; \mathbf{T}\boldsymbol{\beta}_k, \sigma_k^2\mathbf{I}), \tag{1}$$

where $\boldsymbol{\theta} = (\pi_1, \ldots, \pi_K, \boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_K, \sigma_1^2, \ldots, \sigma_K^2)$ is the complete parameter vector, the π_k are the proportions of the mixture satisfying $0 \leq \pi_k \leq 1\ \forall k$ and $\sum_{k=1}^{K} \pi_k = 1$, $\boldsymbol{\beta}_k \in \mathbb{R}^{p+1}$ and $\sigma_k^2 > 0$ are respectively the coefficient vector of the kth regression model and the associated noise variance. The matrix $\mathbf{T} = (T_{ju})$ is an $m \times (p+1)$ Vandermonde matrix verifying $T_{ju} = t_j^{u-1}$ for all $1 \leq j \leq m$ and $1 \leq u \leq (p+1)$, and $\mathcal{N}_m(\cdot; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the Gaussian density in $\mathbb{R}^m$ with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. This corresponds to the class-specific polynomial prototype functions $g_k(t) = \sum_{u=1}^{p+1} \boldsymbol{\beta}_{ku} t^{u-1}$. Figure 1 gives an illustration of time series generated according to the mixture model defined by Eq. (1).

2.2 Fitting the model

Assuming that the observed time series $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ are independent, the parameter vector $\boldsymbol{\theta}$ is estimated by maximizing the log-likelihood

$$\mathcal{L}(\boldsymbol{\theta}) = \log \prod_{i=1}^{n} f(\boldsymbol{x}_i|\boldsymbol{t};\boldsymbol{\theta}) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} \pi_k \mathcal{N}_m(\boldsymbol{x}_i; \mathbf{T}\boldsymbol{\beta}_k, \sigma_k^2\mathbf{I}) \tag{2}$$

via the Expectation–Maximization algorithm initiated by Dempster et al. (1977).

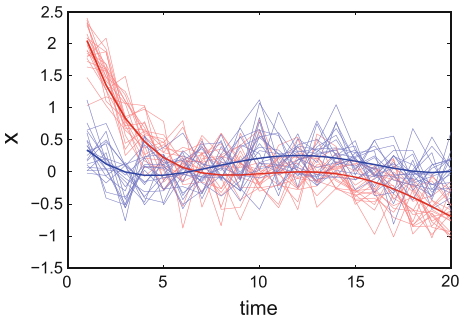

Fig. 1 Example of 40 time series simulated according to a polynomial regression mixture model, with $K = 2$ and $p = 4$

Once the parameters have been estimated, a time series partition is obtained by assigning each series $\boldsymbol{x}_i$ to the cluster having the highest posterior probability

$$P(Z_i = k|\boldsymbol{t}, \boldsymbol{x}_i; \boldsymbol{\theta}) = \frac{\pi_k \mathcal{N}_m(\boldsymbol{x}_i; \mathbf{T}\boldsymbol{\beta}_k, \sigma_k^2 \mathbf{I})}{\sum_{h=1}^{K} \pi_h \mathcal{N}_m(\boldsymbol{x}_i; \mathbf{T}\boldsymbol{\beta}_h, \sigma_h^2 \mathbf{I})}. \tag{3}$$

3 Clustering time series with changes in regime

3.1 The global mixture model

As with the standard regression mixture model, the mixture model introduced for clustering time series with changes in regime assumes that, given the time grid $\boldsymbol{t}$, the variables $\mathbf{X}_i$ are independently generated according to the global mixture model

$$f(\boldsymbol{x}_i|\boldsymbol{t}; \boldsymbol{\theta}) = \sum_{k=1}^{K} \pi_k f_k(\boldsymbol{x}_i|\boldsymbol{t}; \boldsymbol{\theta}_k), \tag{4}$$

where $\boldsymbol{\theta} = (\pi_1, \ldots, \pi_K, \boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K)$, $\pi_1, \ldots, \pi_K$ denote the proportions of the mixture, and $\boldsymbol{\theta}_k$ the parameters of the different component densities f_k. The main difference between the model proposed here and the RegMix model (Gaffney and Smyth 1999) lies in the definition of the component densities f_k, described in the following section.

3.2 Structure of the mixture components

We assume that each time series $\mathbf{X}_i = (X_{i1}, \ldots, X_{im})$ which originates from the kth mixture component f_k is generated as follows. At each time point t_j, the variable X_{ij} follows one of L (class-specific) pth order polynomial regression models (see Sect. 2.1). In this way, the individual variables X_{ij} of a time series $\mathbf{X}_i$ may switch from one regression model to another one in the course of time and the switching times might be different for different $\mathbf{X}_i$.

The assignment of the X_{ij}'s to the different (sub) regression models is specified by a hidden random process denoted by $\mathbf{W}_i = (W_{i1}, \ldots, W_{im})$, where $W_{ij} \in \{1, \ldots, L\}$

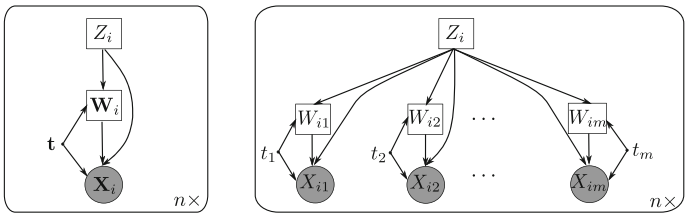

Fig. 2 Graphical representation of the variables involved in the regression mixture model for clustering and segmentation; the nodes in *gray circles* represent the observed variables

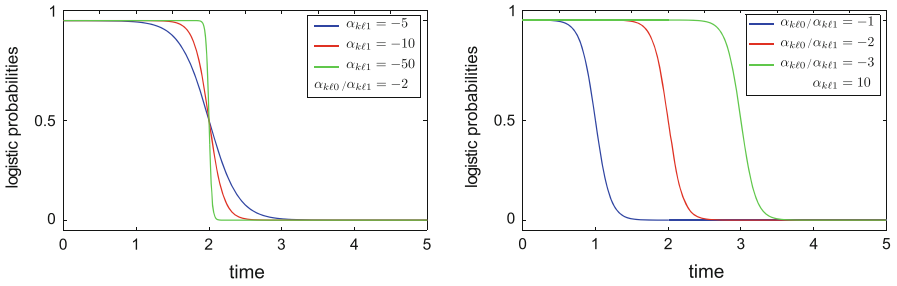

Fig. 3 Dynamical behavior of the logistic probabilities $\kappa_{k\ell}$ as a function of the parameters $\alpha_{k\ell 1}$ and $\alpha_{k\ell 0}/\alpha_{k\ell 1}$

is the label of the polynomial regression model that control the time series $\mathbf{X}_i$ at time t_j. Thus, given the class label $Z_i = k$ and the assignment labels in $\mathbf{W}_i$, the individual observations X_{ij} of a series $\mathbf{X}_i$ are given by:

$$X_{ij} = \sum_{\ell=1}^{L} W_{ij\ell}(\mathbf{T}'_j \boldsymbol{\beta}_{k\ell} + \sigma_{k\ell}\varepsilon_{ij}) \quad \text{for } j = 1, \ldots, m, \tag{5}$$

where $\varepsilon_{ij} \sim \mathcal{N}_1(0, 1)$ is a random noise and $W_{ij\ell} = 1$ if $W_{ij} = \ell$ and 0 otherwise. The parameters $\sigma_{k\ell} > 0$ and $\boldsymbol{\beta}_{k\ell} \in \mathbb{R}^{p+1}$ are respectively the noise standard deviation and the coefficient vector of the ℓth regression model of the kth cluster. $\mathbf{T}'_j$ denotes the transpose of the vector $\mathbf{T}_j = (1, t_j, \ldots, t_j^p)'$. Since the model defined by Eqs. (4) and (5) is related to time series clustering and segmentation, we shall call it "ClustSeg". The graphical model associated to the ClustSeg model is displayed in Fig. 2.

The random variables $(W_{i1}, \ldots, W_{im})$ associated to the regression model labels of the time series $\mathbf{X}_i$ are assumed to be generated according to the multinomial distribution $\mathcal{M}(1, \kappa_{k1}(t_j; \boldsymbol{\alpha}_k), \ldots, \kappa_{kL}(t_j; \boldsymbol{\alpha}_k))$, where

$$\begin{aligned} \kappa_{k\ell}(t_j; \boldsymbol{\alpha}_k) &= P(W_{ij} = \ell | Z_i = k) \\ &= \frac{\exp(\boldsymbol{\alpha}_{k\ell 1} t_j + \boldsymbol{\alpha}_{k\ell 0})}{\sum_{h=1}^{L} \exp(\boldsymbol{\alpha}_{kh1} t_j + \boldsymbol{\alpha}_{kh0})} \quad \text{for all } j, \ell, k \end{aligned} \tag{6}$$

is a logistic function with parameter vector $\boldsymbol{\alpha}_k = (\boldsymbol{\alpha}_{k1}, \ldots, \alpha_{kL})$ and $\boldsymbol{\alpha}_{k\ell} = (\boldsymbol{\alpha}_{k\ell 0}, \boldsymbol{\alpha}_{k\ell 1})$. As shown in Fig. 3, the logistic function defined in this way

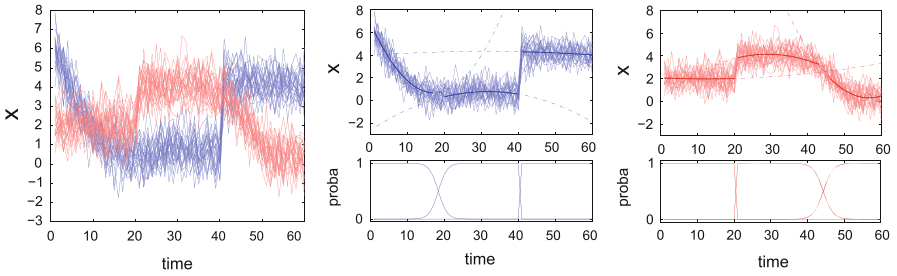

Fig. 4 (*Left*) example of 50 time series generated according to the ClustSeg model with $K = 2$, $L = 3$ and $p = 2$; (*Middle top and right top*) Classes with their associated polynomials; (*Middle bottom and right bottom*) Corresponding logistic probabilities

controls, through the parameters $\frac{\boldsymbol{\alpha}_{k\ell 0}}{\boldsymbol{\alpha}_{k\ell 1}}$ and $\boldsymbol{\alpha}_{k\ell 1}$, the transition time points and the type of transition between the different polynomial regimes involved in the generation process of the time series. Thus, given $Z_i = k$, the m individual variables X_{ij} of a series $\mathbf{X}_i$ at times $t_1, \ldots, t_j$ are independently distributed according to the mixture model given by the density

$$p(x_{ij}|t_j; \boldsymbol{\theta}_k) = \sum_{\ell=1}^{L} \kappa_{k\ell}(t_j; \boldsymbol{\alpha}_k) \mathcal{N}_1(x_{ij}; \boldsymbol{\beta}_{k\ell}^T \mathbf{T}_j, \sigma_{k\ell}^2). \tag{7}$$

The class specific density f_k can thus be written as

$$f_k(\boldsymbol{x}_i|\boldsymbol{t}; \boldsymbol{\theta}_k) = \prod_{j=1}^{m} \sum_{\ell=1}^{L} \kappa_{k\ell}(t_j; \boldsymbol{\alpha}_k) \mathcal{N}_1(x_{ij}; \boldsymbol{\beta}_{k\ell}^T \mathbf{T}_j, \sigma_{k\ell}^2). \tag{8}$$

Figure 4 shows an example of 50 time series generated according to the ClustSeg model.

3.3 A combined clustering-segmentation model

Let $E_{k\ell}$ be the subset of $[t_1; t_m]$ defined by

$$E_{k\ell} = \left\{ t \in [t_1; t_m] \quad | \quad \kappa_{k\ell}(t; \boldsymbol{\alpha}_k) = \max_{1 \leq h \leq L} \kappa_{kh}(t; \boldsymbol{\alpha}_k) \right\}. \tag{9}$$

It can be easily verified that, for each class k, $(E_{k1}, \ldots, E_{kL})$ is a partition of the set of times $[t_1; t_m]$. Moreover, it can be proved that $E_{k\ell}$ is convex as the intersection of convex parts of $[t_1; t_m]$ (see Appendix A). As some of the subsets $E_{k\ell}$ can be empty, the proposed model leads to a cluster-specific segmentation $\mathbf{E}_k = (E_{k1}, \ldots, E_{kL'})$ $(L' \leq L)$ of the times $[t_1; t_m]$, into contiguous parts.

3.4 Parameter estimation via the EM algorithm

The parameters of the proposed model are estimated by maximizing the log-likelihood defined by

$$\mathcal{L}(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log f(\boldsymbol{x}_i|\boldsymbol{t};\boldsymbol{\theta}) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} \pi_k \left(\prod_{j=1}^{m} \sum_{\ell=1}^{L} \kappa_{k\ell}(t_j;\boldsymbol{\alpha}_k) \mathcal{N}_1\left(x_{ij};\boldsymbol{\beta}'_{k\ell}\mathbf{T}_j,\sigma^2_{k\ell}\right)\right). \quad (10)$$

The Expectation–Maximization algorithm (Dempster et al. 1977) is used for the maximization of this log-likelihood, a problem which cannot be solved analytically. Let us recall that the EM algorithm requires a complete data specification, whose log-likelihood can be maximized more easily than the observed marginal data log-likelihood. Here, the "complete data" are obtained by joining to each series $\boldsymbol{x}_i$ its latent class membership label z_i and its unobservable assignment indicator $\boldsymbol{w}_i = (w_{i1}, \ldots, w_{im})$ to the different sub-regression models. Using the binary coding of z_i and $\boldsymbol{w}_{ij}$,

$$z_{ik} = \begin{cases} 1 & \text{if } z_i = k \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \quad w_{ij\ell} = \begin{cases} 1 & \text{if } w_{ij} = \ell \\ 0 & \text{otherwise,} \end{cases}$$

the complete data log-likelihood can be written as

$$\mathcal{L}_c(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log p(\boldsymbol{x}_i, z_i, \boldsymbol{w}_i|\boldsymbol{t};\boldsymbol{\theta}) = \sum_{i=1}^{n}\sum_{k=1}^{K} z_{ik} \log \pi_k + \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{K}\sum_{\ell=1}^{L} z_{ik} w_{ij\ell} \log\left(\kappa_{k\ell}(t_j;\boldsymbol{\alpha}_k)\mathcal{N}_1\left(x_{ij};\boldsymbol{\beta}'_{k\ell}\mathbf{T}_j,\sigma^2_{k\ell}\right)\right). \quad (11)$$

Given an initial value $\boldsymbol{\theta}^{(0)}$ of the parameter vector, the EM algorithm alternates the following two steps until convergence.

E-Step (Expectation)

This step consists in evaluating the expectation of the complete data log-likelihood conditionally on the observed data and the current parameter vector $\boldsymbol{\theta}^{(q)}$, q denoting the current iteration:

$$Q(\boldsymbol{\theta},\boldsymbol{\theta}^{(q)}) = E[\mathcal{L}_c(\boldsymbol{\theta})|\boldsymbol{t},\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n;\boldsymbol{\theta}^{(q)}] = \sum_{i=1}^{n}\sum_{k=1}^{K} r_{ik}^{(q)} \log \pi_k + \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{K}\sum_{\ell=1}^{L} \lambda_{ijk\ell}^{(q)} \log(\kappa_{k\ell}(t_j;\boldsymbol{\alpha}_k)\mathcal{N}_1(x_{ij};\boldsymbol{\beta}'_{k\ell}\mathbf{T}_j,\sigma^2_{k\ell})), \quad (12)$$

where

$$r_{ik}^{(q)} = E[z_{ik}|\boldsymbol{t},\boldsymbol{x}_i;\boldsymbol{\theta}^{(q)}] = \frac{\pi_k^{(q)} f_k(\boldsymbol{x}_i|\boldsymbol{t};\boldsymbol{\theta}_k^{(q)})}{\sum_{h=1}^{K} \pi_h^{(q)} f_h(\boldsymbol{x}_i|\boldsymbol{t};\boldsymbol{\theta}_h^{(q)})} \quad (13)$$

is the posterior probability that time series $\boldsymbol{x}_i$ originates from cluster k, and

$$\begin{aligned}\lambda_{ijk\ell}^{(q)} &= E[z_{ik}\, w_{ij\ell}|\boldsymbol{t}, \boldsymbol{x}_i; \boldsymbol{\theta}^{(q)}] \\ &= r_{ik}^{(q)} \times \frac{\kappa_{k\ell}(t_j; \boldsymbol{\alpha}_k^{(q)})\, \mathcal{N}_1(x_{ij}; \boldsymbol{\beta}_{k\ell}^{(q)'}\mathbf{T}_j, (\sigma_{k\ell}^{(q)})^2)}{\sum_{h=1}^{L} \kappa_{kh}(t_j; \boldsymbol{\alpha}_k^{(q)})\, \mathcal{N}_1(x_{ij}; \boldsymbol{\beta}_{kh}^{(q)'}\mathbf{T}_j, (\sigma_{kh}^{(q)})^2)} \end{aligned} \tag{14}$$

is the posterior probability that the jth value of $\boldsymbol{x}_i$, i.e., the observation x_{ij} at time t_j, originates from the ℓth sub-regression model of cluster k.

M-Step (Maximization)

This step consists in computing the parameter vector $\boldsymbol{\theta}^{(q+1)}$ that maximizes the quantity $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(q)})$ with respect to $\boldsymbol{\theta}$. For our purposes this quantity can be written as

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(q)}) = Q_1((\pi_k)) + Q_2((\boldsymbol{\alpha}_k)) + Q_3((\beta_{k\ell}, \sigma_{k\ell}^2)),$$

where

$$Q_1((\pi_k)) = \sum_{i=1}^{n}\sum_{k=1}^{K} r_{ik}^{(q)} \log \pi_k, \tag{15}$$

$$Q_2((\boldsymbol{\alpha}_k)) = \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{K}\sum_{\ell=1}^{L} \lambda_{ijk\ell}^{(q)} \log(\kappa_{k\ell}(t_j; \boldsymbol{\alpha}_k)), \tag{16}$$

$$Q_3((\beta_{k\ell}, \sigma_{k\ell}^2)) = \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{K}\sum_{\ell=1}^{L} \lambda_{ijk\ell}^{(q)} \log(\mathcal{N}_1(x_{ij}; \boldsymbol{\beta}'_{k\ell}\mathbf{T}_j, \sigma_{k\ell}^2)). \tag{17}$$

Q can thus be maximized by separately maximizing the quantities Q_1 w.r.t. $(\pi_1, \ldots, \pi_K)$, Q_2 w.r.t. $(\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_K)$ and Q_3 w.r.t. $((\beta_{k\ell}, \sigma_{k\ell}^2)_{k,\ell})$. As in the classical Gaussian mixture model, it can easily be shown that the proportions π_k that maximize Q_1 under the constraint $\sum_{k=1}^{K} \pi_k = 1$ are given by

$$\pi_k^{(q+1)} = \frac{\sum_{i=1}^{n} r_{ik}^{(q)}}{n}. \tag{18}$$

Q_2 can be maximized with respect to the $\boldsymbol{\alpha}_k$ by separately solving K weighted logistic regression problems:

$$\boldsymbol{\alpha}_k^{(q+1)} = \arg\max_{\boldsymbol{\alpha}_k} \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\ell=1}^{L} \lambda_{ijk\ell}^{(q)} \log(\kappa_{k\ell}(t_j; \boldsymbol{\alpha}_k)) \tag{19}$$

through the well known Iteratively Reweighted Least Squares (IRLS) algorithm (Green 1984; Chamroukhi et al. 2010). Let us recall that the IRLS algorithm, which is generally

used to estimate the parameters of a logistic regression model, is equivalent to the following Newton Raphson algorithm (Green 1984; Chamroukhi et al. 2010):

$$\boldsymbol{\alpha}_k^{(v+1)} = \boldsymbol{\alpha}_k^{(v)} - \left[\frac{\partial^2 Q_{2k}}{\partial \boldsymbol{\alpha}_k \partial \boldsymbol{\alpha}_k^T}\right]^{-1}_{\boldsymbol{\alpha}_k=\boldsymbol{\alpha}_k^{(v)}} \left[\frac{\partial Q_{2k}}{\partial \boldsymbol{\alpha}_k}\right]_{\boldsymbol{\alpha}_k=\boldsymbol{\alpha}_k^{(v)}}, \quad \text{for } v = 0, 1, \ldots \tag{20}$$

where $Q_{2k} = \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\ell=1}^{L} \lambda_{ijk\ell}^{(q)} \log \kappa_{k\ell}(t_j; \boldsymbol{\alpha}_k)$.

Maximizing Q_3 with respect to $(\boldsymbol{\beta}_{k\ell})_{k,\ell}$ consists in analytically solving $K \times L$ weighted least-squares problems. It can be shown that

$$\boldsymbol{\beta}_{k\ell}^{(q+1)} = \left[\mathbf{T}'\left(\sum_{i=1}^{n} \Lambda_{ik\ell}^{(q)}\right)\mathbf{T}\right]^{-1}\left[\mathbf{T}\left(\sum_{i=1}^{n} \Lambda_{ik\ell}^{(q)} \boldsymbol{x}_i\right)\right] \quad \forall k, \ell, \tag{21}$$

where $\Lambda_{ik\ell}^{(q)}$ is the $m \times m$ diagonal matrix whose diagonal elements are $\{\lambda_{ijk\ell}^{(q)} \; ; \; j = 1, \ldots, m\}$. The maximization of Q_3 with respect to $(\sigma_{k\ell}^2)_{k,\ell}$ gives

$$(\sigma_{k\ell}^{(q+1)})^2 = \frac{\sum_{i=1}^{n} \left\| \sqrt{\Lambda_{ik\ell}^{(q)}}(\boldsymbol{x}_i - \mathbf{T}\boldsymbol{\beta}_{k\ell}^{(q+1)}) \right\|^2}{\sum_{i=1}^{n} \text{trace}(\Lambda_{ik\ell}^{(q)})} \quad \forall k, \ell, \tag{22}$$

where $\sqrt{\Lambda_{ik\ell}^{(q)}}$ is the $m \times m$ diagonal matrix with diagonal elements $\sqrt{\lambda_{ijk\ell}^{(q)}}$ for $j = 1, \ldots, m$ and $\|\cdot\|$ is the norm corresponding to the Euclidean distance.

M-step for three parsimonious clustering-segmentation models

Common segmentation of time axis for all clusters In certain situations, the segmentation defined by the $\boldsymbol{\alpha}_k$ $(k = 1, \ldots, K)$ may be constrained to be common for each cluster, that is $\boldsymbol{\alpha}_k = \boldsymbol{\alpha}$ $\forall k$. In that case, the quantity Q_2 to be maximized can be rewritten as:

$$Q_2(\boldsymbol{\alpha}) = \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\ell=1}^{L} \lambda_{ij\cdot\ell}^{(q)} \log(\kappa_\ell(t_j; \boldsymbol{\alpha})), \tag{23}$$

where $\lambda_{ij\cdot\ell}^{(q)} = \sum_{k=1}^{K} \lambda_{ijk\ell}^{(q)}$. The IRLS algorithm can therefore be used to compute the parameter $\boldsymbol{\alpha}^{(q+1)}$, in the same way as for the unconstrained situation.

Common variance for regression models from the same cluster In other situations, it may be useful to constrain the regression model variances to be identical within a same cluster. In that case, $\sigma_{k\ell}^2 = \sigma_k^2$ $\forall \ell$. The updating formula for the variance can thus be written as:

$$(\sigma_k^{(q+1)})^2 = \frac{\sum_{i=1}^{n}\sum_{\ell=1}^{L} \left\| \sqrt{\Lambda_{ik\ell}^{(q)}}\left(\boldsymbol{x}_i - \mathbf{T}\boldsymbol{\beta}_{k\ell}^{(q+1)}\right) \right\|^2}{\sum_{i=1}^{n}\sum_{\ell=1}^{L} \text{trace}(\Lambda_{ik\ell}^{(q)})}. \tag{24}$$

Common variance for all regression models If the model variances are constrained to be identical for all regression models, we have $\sigma_{k\ell}^2 = \sigma^2 \; \forall k, \ell$. The updating formula for the variance takes the form:

$$\left(\sigma^{(q+1)}\right)^2 = \frac{\sum_{i=1}^{n}\sum_{k=1}^{K}\sum_{\ell=1}^{L}\left\|\sqrt{\Lambda_{ik\ell}^{(q)}}(\boldsymbol{x}_i - \mathbf{T}\boldsymbol{\beta}_{k\ell}^{(q+1)})\right\|^2}{n \times m}. \tag{25}$$

3.5 Complexity analysis of the clustering-segmentation EM algorithm

The algorithmic complexity of the proposed EM algorithm depends on the computation costs of the E and M steps. The complexity of the E step is $O(KLnmp)$, which mainly comprises the calculation of the probabilities $\kappa_{k\ell}(t_j;\boldsymbol{\alpha}_k)$ and densities $\mathcal{N}_1(x_{ij};\boldsymbol{\beta}_{k\ell}'\mathbf{T}_j,\sigma_{k\ell}^2)$, for all k, ℓ, i, j. For each k and ℓ, the regression coefficients update equation (21) requires the computation and inversion of a $(p+1)\times(p+1)$ matrix which can be done in $O(nmp^3)$, and the variance update equation (22) is computed in $O(nmp)$. Each iteration of the IRLS algorithm requires a $2(L-1)\times 2(L-1)$ Hessian matrix to be computed and inverted, which is done in $O(L^3nm)$. From the computation costs of the regression coefficients, the variances and the logistic functions coefficients, it can be deduced that the M step has complexity $O(KLnmp^3)$. Consequently, the computational complexity of the proposed EM algorithm is $O(I_{EM}I_{IRLS}KL^3nmp^3)$, where I_{EM} is the number of iteration of the EM algorithm and I_{IRLS} is the maximum number of iterations of the inner IRLS loops. The parsimonious models described above, while requiring less calculations than the general model, have their complexity also limited by $O(I_{EM}I_{IRLS}KL^3nmp^3)$.

As a particular case, it can be easily verified that the complexity of the RegMix EM algorithm, described in Sect. 2, is $O(I_{EM}Knmp^3)$.

Compared to other clustering and segmentation algorithms such as the k-means type algorithm based on piecewise polynomial regression (Hébrail et al. 2010), whose complexity is $O(I_{KM}KLnm^2p^3)$ where I_{KM} is the number of iterations of the algorithm, our EM algorithm is computationally attractive for large values of m and small values of L.

3.6 Clustering, approximation and segmentation of the time series

From the parameters estimated by the EM algorithm, a partition of the n time series into K clusters can easily be obtained by applying the maximum a posteriori (MAP) rule to the expected membership indicator Z_{ik}, that is the values $\widehat{r}_{ik} = E[Z_{ik}|\boldsymbol{t},\boldsymbol{x}_i;\widehat{\boldsymbol{\theta}}]$, where $\widehat{\boldsymbol{\theta}}$ is the parameter estimated by the EM algorithm:

$$\widehat{z}_i = \arg\max_k \widehat{r}_{ik} \quad \text{for } i = 1,\ldots,n. \tag{26}$$

The class-specific prototype function can be approximated by the estimate $\widehat{\mathbf{c}}_k = (\widehat{c}_{k1},\ldots,\widehat{c}_{km})$, with

$$\widehat{c}_{kj} = E\left[x_{ij}|t_j, z_i = k;\widehat{\boldsymbol{\theta}}\right] = \sum_{\ell=1}^{L}\kappa_{k\ell}(t_j;\widehat{\boldsymbol{\alpha}}_k)\mathbf{T}_j'\widehat{\boldsymbol{\beta}}_{k\ell}. \tag{27}$$

Moreover, a segmentation $\mathbf{E}_k = (E_{k\ell})_{\ell=1,\ldots,L'}$ of the time series originating from the kth cluster can be derived from the estimated parameters by computing $E_{k\ell}$ as defined in Eq. (9).

3.7 Assessing the number of clusters, segments and the regression order

In the context of mixture models and the EM algorithm, the natural criterion for model selection is the Bayesian Information Criterion (BIC) (Schwarz 1978). Unlike for the classical regression mixture model, three parameters need to be tuned: the number of clusters K, the number of segments L and the degree p of the polynomials. The BIC criterion, in this case, can be defined by:

$$BIC(K, L, p) = \mathcal{L}(\widehat{\boldsymbol{\theta}}) - \frac{\nu(K, L, p)}{2} \log(n), \tag{28}$$

where $\widehat{\boldsymbol{\theta}}$ is the parameter vector estimated by the EM algorithm, and $\nu(K, L, p)$ is the number of free parameters of the model. In the ClustSeg model, the number of free parameters

$$\nu(K, L, p) = (K - 1) + 2\,K(L - 1) + L\,K(p + 1) + L\,K \tag{29}$$

comprizes the mixture proportions, the logistic functions parameters, the polynomial coefficients and the variances.

From a practical point of view, the maximum values K_{max}, L_{max} and p_{max} have first to be specified. Then, the EM algorithm is run for $K \in \{1, \ldots, K_{max}\}$, $L \in \{1, \ldots, L_{max}\}$ and $p \in \{1, \ldots, p_{max}\}$, and the BIC criterion is computed. The set (K, L, p) with the highest value of BIC is taken to be right solution. In contrast to more classical mixture model situations where only K_{max} computations of BIC are required to estimate the number of classes of a data set, our situation requires $K_{max} \times L_{max} \times p_{max}$ computations of BIC to estimate the number of classes, segments and polynomial order of a data set, which can be computationally more expensive.

4 Experimental study

This section is devoted to an evaluation of the clustering accuracy of the proposed EM algorithm for time series and segmentation, carried out using simulated time series and real-world time series from a railway application. Results yielded by the ClustSeg model are compared with those provided by the RegMix model described in Sect. 2, and the k-means type clustering and segmentation algorithm based on polynomial piecewise regression (Hébrail et al. 2010), called PWR in the following. Starting from a randomly initialized partition $\mathbf{G} = (G_1, \ldots, G_K)$ of the n time series, the PWR algorithm iteratively maximizes the criterion

$$C(\mathbf{G}, \mathbf{I}, \boldsymbol{\beta}) = \sum_{k=1}^{K} \sum_{\ell=1}^{L} \sum_{i \in G_k} \sum_{j \in I_{k\ell}} (x_{ij} - \mathbf{T}'_j \boldsymbol{\beta}_{k\ell})^2, \tag{30}$$

where $\mathbf{I} = (\mathbf{I}_k)_{k=1,\ldots,K}$ represents K segmentations of $\{t_1, \ldots, t_n\}$ into L intervals and $\mathbf{I}_k = (I_{k\ell})_{\ell=1,\ldots,L}$ is the segmentation corresponding to the kth cluster, by alternating the following two steps until the partition stabilizes:

(a) estimate an L-segment piecewise polynomial function for each cluster, using a dynamic programming procedure;
(b) assign each time series $\boldsymbol{x}_i$ to the closest piecewise polynomial function in the sense of the L_2 distance.

To measure the clustering accuracy, two criteria were used: the percentage of misclassifications between the true partition of the time series and the estimated partition, and the intra-cluster inertia defined by

$$\sum_{k=1}^{K}\sum_{i=1}^{n}\widehat{z}_{ik}\|\boldsymbol{x}_i - \widehat{\mathbf{c}}_k\|^2, \tag{31}$$

where $(\widehat{z}_{ik})$ and $\widehat{\mathbf{c}}_k = (\widehat{c}_{k1}, \ldots, \widehat{c}_{km})$ represent, respectively, the binary partition matrix and the kth prototype series estimated by each of the three compared algorithms, with:

- $\widehat{c}_{kj} = \sum_{\ell=1}^{L} \kappa_{k\ell}(t_j; \widehat{\boldsymbol{\alpha}}_k)\mathbf{T}'_j\widehat{\boldsymbol{\beta}}_{k\ell}$ for the ClustSeg model,
- $\widehat{c}_{kj} = \mathbf{T}'_j\widehat{\boldsymbol{\beta}}_k$ for the RegMix model,
- $\widehat{c}_{kj} = \sum_{\ell=1}^{L} \mathbb{1}_{\widehat{I}_{k\ell}}(t_j)\mathbf{T}'_j\widehat{\boldsymbol{\beta}}_{k\ell}$ for the PWR algorithm, where $\mathbb{1}_{I_{k\ell}}(t)$ denotes the indicator function of the segment $I_{k\ell}$.

4.1 Experiments using simulated data

4.1.1 Simulation protocol and algorithms tuning

Quite generally, we simulate n time series with m discrete time points $t_1 = 1, \ldots, t_m = m$ according to a mixture of K classes whose prototypes functions are nonlinear. In the ClustSeg and PWR approaches, the polynomial coefficients and variances are initialized as follows: K series are randomly selected and segmented into L segments of equal length; the initial polynomial regression parameters are derived from a pth order regression on each segment. In the initial iteration of the EM algorithm, the logistic regression parameters are initialized to zero. The initial polynomial coefficients and variances of the regression mixture approach are obtained by performing a pth-order regression on K randomly drawn series. The initial proportions of the latent classes are set to $\pi_k = 1/K$ for all algorithms. Each algorithm starts with 20 different initializations of $(\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K)$ and the solution with the highest log-likelihood is selected.

4.1.2 Comparisons with the RegMix and PWR approaches in terms of clustering accuracy

A first set of experiments was performed in order to compare the relative performances of the ClustSeg, RegMix and PWR approaches. Each data set, consisting of $n = 50$

Table 1 Prototype series of mixture 1

Cluster	Prototype series
$k = 1$	$c_{1j} = 20 \sin(0.12\pi j) \exp(-0.07j)$
$k = 2$	$c_{2j} = 11.1 \times \mathbb{1}_{[1;8[}(j) - 4.3 \times \mathbb{1}_{[8;18[}(j) + 3.8 \times \mathbb{1}_{[18;24[}(j) - 0.3 \times \mathbb{1}_{[24;50]}(j)$

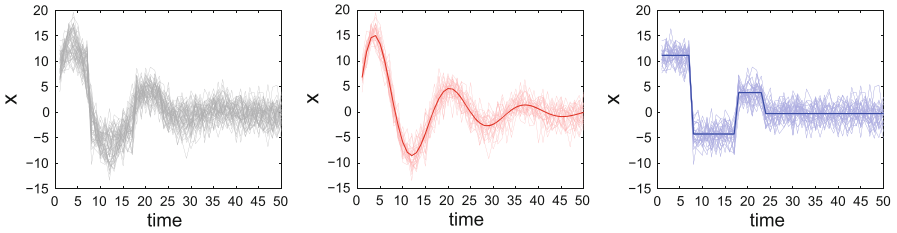

Fig. 5 Example of $n = 50$ series simulated according to mixture 1 with $\sigma = 2$ (*left*) and series corresponding to the two clusters, with their prototype curve (*middle and right*)

time series of length $m = 50$, was simulated according to a mixture of $K = 2$ classes with equal proportions ($\pi_1 = \pi_2 = 1/2$). As described in Table 1, the first class prototype function is a nonlinear function and the second one is a piecewise constant function. Within each class, the time series were generated by adding a centered Gaussian noise with variance σ^2 to a prototype series. Values of the noise variance were chosen constant within each of the time series data sets. Figure 5 provides an illustration of time series simulated according to this model, which will be called mixture 1 in the following. For each noise standard deviation σ in the set $\{0.5, 1, 1.5, 2, 2.5, 3, 3.5\}$, 25 different samples of 50 time series were generated and the results (misclassification rates and intra-cluster inertia) were averaged over these 25 random samples.

Since the goal of this set of experiments was to evaluate the ClustSeg approach in terms of estimation accuracy, the number of clusters has been set to $K = 2$ in the three compared algorithms. For each simulated data set, as the true number of segments and polynomial order are unknown, the ClustSeg EM algorithm is run for $L = 1, \ldots, 7$ and $p = 1, \ldots, 7$ (i.e. $L_{max} = 7$ and $p_{max} = 7$), and only the solution providing the highest BIC value is retained. The same strategy is applied for the RegMix EM algorithm by simply varying the polynomial order p from 1 to $p_{max} = 13$, as described in Sect. 3.7.

The model underlying the PWR approach can be viewed as a Gaussian mixture with hard cluster assignment, uniform cluster prior and identical variances, where the class prototype functions are piecewise polynomial functions. Since the likelihood in this situation is $\mathcal{L} = -\frac{1}{2}C$ up to a constant, the number of segments and the polynomial order of this approach were selected by maximizing the following BIC-like criterion :

$$BIC(K, L, p) = -\frac{C}{2} - \frac{\nu_{K,L,p}}{2} \log(n), \tag{32}$$

where $\nu_{K,L,p} = K(L-1) + KL(p+1)$ is the number of free parameters of the model, including the polynomial coefficients $\boldsymbol{\beta}_{k\ell}$ and the boundaries of the segments $I_{k\ell}$.

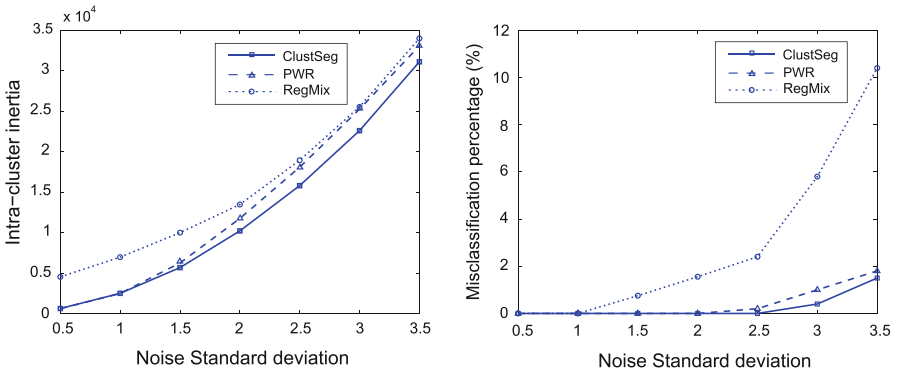

Fig. 6 (*Left*) Misclassification rate and (*right*) Intra-cluster inertia in relation to the noise standard deviation, obtained with the ClustSeg, PWR and RegMix approaches

Figure 6 shows the misclassification rate and the intra-cluster inertia obtained with the ClustSeg, RegMix and PWR approaches as a function of the noise variance. The intra-cluster inertia and misclassification percentages obtained by the three approaches naturally increases with the variance level, but the ClustSeg approach performs better than its competitor. Nevertheless, the classification results provided by the ClustSeg and PWR approach are very close to each other. Their difference in terms of intra-cluster inertia are more distinct, particularly for noise standard deviations greater than 2. The misclassification rate of the RegMix approach increases faster than that of the ClusSeg and PWR approaches, which can be attributed to the fact that a single polynomial cannot accurately model the simulated classes.

An example of the clustering results obtained with the ClustSeg approach is displayed in Fig. 7. It will be observed that the two classes obtained are well represented by four polynomials of order 3 weighted by logistic functions.

4.1.3 Comparisons of the RegMix and PWR approaches in terms of computational speed

To illustrate the algorithmic complexity of the ClustSeg, RegMix and PWR approaches described in Sect. 3.5, this second set of experiments shows how their execution time varies when the length m of the time series increases. For this purpose, we record their CPU time for m varying in the set $\{25, 50, 100, 200, 500, 1{,}000\}$ by using a 2 GHz Pentium Dual-Core. For each value of m, 25 data sets of $n = 50$ time series were generated according to mixture 1, by varying the time sampling frequency. Figure 8 displays the CPU times (averaged over the 25 data sets) for the ClustSeg, RegMix and PWR approaches with respect to the time series length m.

Not surprisingly, the RegMix approach is found to be computationally less expensive than its competitors. It can also be observed that the CPU times of the three compared approaches are almost the same until $m = 200$. For large values of m ($m \geq 300$), the CPU times of the ClustSeg approach are smaller than those provided by the PWR approach which increases considerably with m. In particular, for $m = 500$,

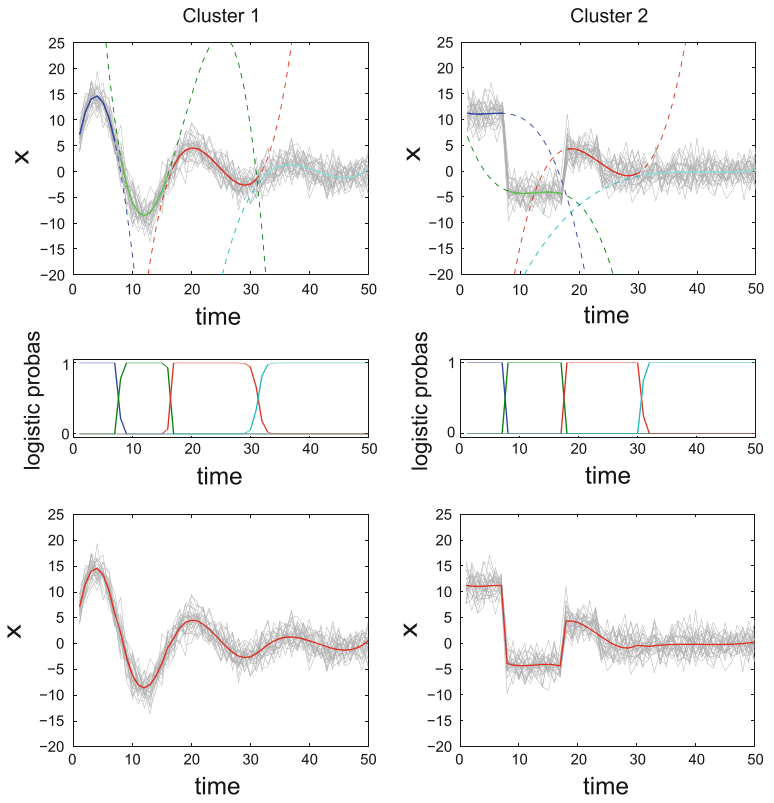

Fig. 7 Clustering results provided by the ClustSeg model applied with the BIC estimates $L = 4$ and $p = 3$: (*top*) Clusters with their estimated polynomials, (*middle*) logistic probabilities and (*bottom*) estimated prototype series

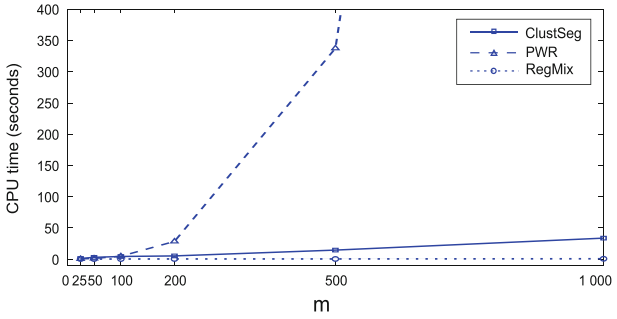

Fig. 8 CPU time (in s) obtained with the ClustSeg, RegMix and PWR approaches in relation to the time series' length m

ClustSeg is about 7 time slower than PWR. These experiments clearly show that the ClustSeg approach, while being computationally more expensive than the RegMix approach, is more efficient than the clustering-segmentation PWR approach for large values of m and small values of L.

Table 2 Prototype series of mixture 2

Cluster	Prototype functions
$k = 1$	$c_{1j} = 10\,\mathbb{1}_{[1;20[}(j) + (0.5j)\,\mathbb{1}_{[20;30[}(j) + (-0.75j + 37.5)\,\mathbb{1}_{[30;50[}(j)$ $+(0.8j - 40)\,\mathbb{1}_{[50;65[}(j) + 12\,\mathbb{1}_{[65;75]}(j)$
$k = 2$	$c_{2j} = 8\,\mathbb{1}_{[1;20[}(j) + 12\,\mathbb{1}_{[20;35[}(j) + (-0.47j + 28.3)\,\mathbb{1}_{[35;50[}(j)$ $+5\,\mathbb{1}_{[50;65[}(j) + 14\,\mathbb{1}_{[65;75]}(j)$

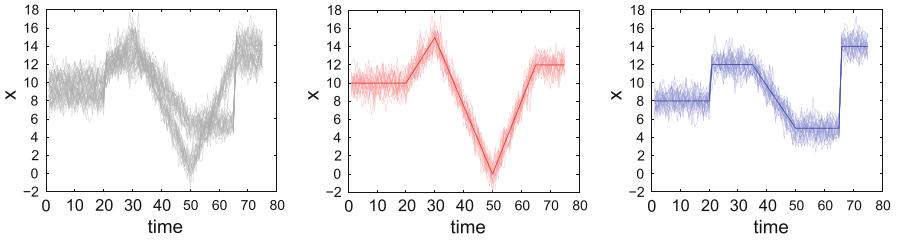

Fig. 9 Example of $n = 50$ time series simulated according to mixture 2 (*left*) and series corresponding to the two clusters, with their prototype function (*center and right*)

4.1.4 Selecting the number of clusters and segments

Our third set of experiments is designed to the evaluation of the ClustSeg approach in terms of identifying the correct number of clusters and segments. Simulations were performed to compare the number of clusters and segments estimated for the ClustSeg approach, to those estimated for the PWR approach based on hard transitions between polynomials. For each simulated sample, the number of clusters was computed by running the algorithms with K varying from 1 to $K_{max} = 3$, L varying from 1 to $L_{max} = 4$, and p varying from 0 to $p_{max} = 3$, and then selecting the triplet (K, L, p) which maximizes the BIC criterion defined by Eq. (28) (for the ClustSeg approach) or the BIC-like criterion defined by Eq. (32) (for the PWR approach). The process is repeated for 100 different random samples, each sample consisting of $n = 50$ time series of length $m = 75$ generated according to a mixture of $K = 2$ classes with $L = 4$ segments, whose prototype series are given in Table 2. The noise standard deviation was set to $\sigma = 1$ for all the data sets. In the following, this mixture will be called mixture 2. Figure 9 shows an example of times series simulated according to mixture 2.

The selection rate for each triplet (K, L, p) over the 100 random samples is displayed in Table 3 as a percentage. Not surprisingly, we observe that the selection rates associated to $K = 1$ and to the numbers of segments $L = 1$ and $L = 2$, which are not adapted to the time series simulated according to mixture 2, are equal zero. The models with the highest percentages of selection (89% and 75%) are those with $(K, L, p) = (2, 4, 3)$ for the ClustSeg approach and $(K, L, p) = (3, 4, 3)$ for the MixReg Approach. Although the polynomial order is slightly overestimated, the true numbers of classes and segments are well estimated by the ClustSeg approach. It can be seen that the number of classes is overestimated by the PWR approach: the number of clusters and segments is correctly detected for only 11% of the simulated data sets. These results are encouraging in terms of model selection.

Table 3 Percentage of selecting (K, L, p)

	ClustSeg				PWR			
	$p=0$	$p=1$	$p=2$	$p=3$	$p=0$	$p=1$	$p=2$	$p=3$
$K=1$								
$L=1$	0	0	0	0	0	0	0	0
$L=2$	0	0	0	0	0	0	0	0
$L=3$	0	0	0	0	0	0	0	0
$L=4$	0	0	0	0	0	0	0	0
$K=2$								
$L=1$	0	0	0	0	0	0	0	0
$L=2$	0	0	0	0	0	0	0	0
$L=3$	0	0	0	0	0	0	0	0
$L=4$	0	0	11	**89**	0	0	0	11
$K=3$								
$L=1$	0	0	0	0	0	0	0	0
$L=2$	0	0	0	0	0	0	0	0
$L=3$	0	0	0	0	0	0	0	1
$L=4$	0	0	0	0	0	3	10	**75**

4.2 Experiments using real world data

As mentioned in the introduction, the main motivation behind this study was diagnosing problems in the rail switches that allow trains to change tracks at junctions. A switching operation consists in moving laterally some linked tapering rails (also known as points) into one of two positions. This operation is generally operated by an electrical motor.

An important preliminary task in the diagnostic process is the automatic identification of groups of switching operations that have similar characteristics, by analyzing time series of electrical power consumption acquired during switching operations.

The specificity of the time series 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:

- the starting phase: period between the activation of the motor and the starting of the switch operation itself,
- the points unlocking: phase where the switch points are unlocked, that makes them ready for the translation,
- the points translation: phase corresponding to the translation of the points,
- the points locking: phase where the switch points are locked;
- the friction phase: phase where an additional effort is applied to ensure the locking.

We accomplished this clustering task by using our EM algorithm, designed for estimating the parameters of a mixture of hidden process regression models. We compared the proposed EM algorithm to the regression mixture EM algorithm described in Sect. 3.4, on a data set of $n = 140$ time series (see Fig. 10). This data set is composed of four

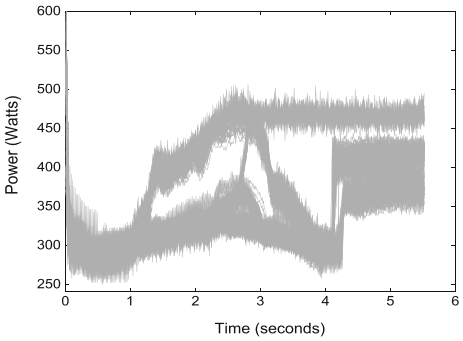

Fig. 10 Electrical power consumption time series acquired during $n = 140$ switch operations and subject to changes in regime

Table 4 BIC criterion (divided by 10^5) obtained with the ClustSeg, PWR and RegMix approaches for p varying from 1 to 6, with $K = 4$ and $L = 5$

p	1	2	3	4	5	6
ClustSeg	−3.104	−3.009	**−2.992**	−2.994	−3.029	−3.132
PWR	−3.288	−3.123	**−3.061**	−3.104	−3.106	−3.734
p	1	2	3	4	5	6
RegMix	−4.057	−3.982	−3.921	−3.863	−3.641	−3.636
p	7	8	9	10	11	
RegMix	−3.608	−3.480	−3.466	−3.474	**−3.391**	

Table 5 Misclassification percentage and intra-cluster inertia obtained for the ClustSeg, PWR and RegMix approaches

	ClustSeg	PWR	RegMix
Misclassification (%)	9.28	10.72	11.42
Intra-cluster inertia	1.1566×10^7	1.3587×10^7	2.6583×10^7

classes identified by an expert: a defect-free class (35 time series), a class with a minor defect (40 time series), a class with a type 1 critical defect (45 time series) and a class with a type 2 critical defect (20 time series).

In this section, only the selection of the polynomial order p using BIC is performed, for $K = 4$ and $L = 5$, which respectively correspond to the number of operating states to be identified in our diagnosis problem and the five electromechanical phases of a switching operation. In practice, we have observed that the BIC criterion tended to overestimate K and L for the real time series. This might suggest, as in the situation of Gaussian mixtures, that an Integrated Classification Likelihood (ICL) type criterion (Biernacki et al. 2000) which would also take into account the clustering-segmentation objective, could be interesting to analyze.

Table 4 shows the BIC criteria in relation to the polynomial order p, obtained by the ClustSeg, RegMix and PWR strategies, for $K = 4$ and $L = 5$. The maximum values of BIC are obtained for $p = 3$ with the ClustSeg and PWR approaches, and $p = 11$ with the RegMix approach. This result confirms a preliminary choice made in conjunction with the expert, which consisted in modeling each regime by a third order polynomial.

Table 5 displays the misclassification error rates and the corresponding intra-cluster inertia. Although misclassification rates obtained by the three approaches are very

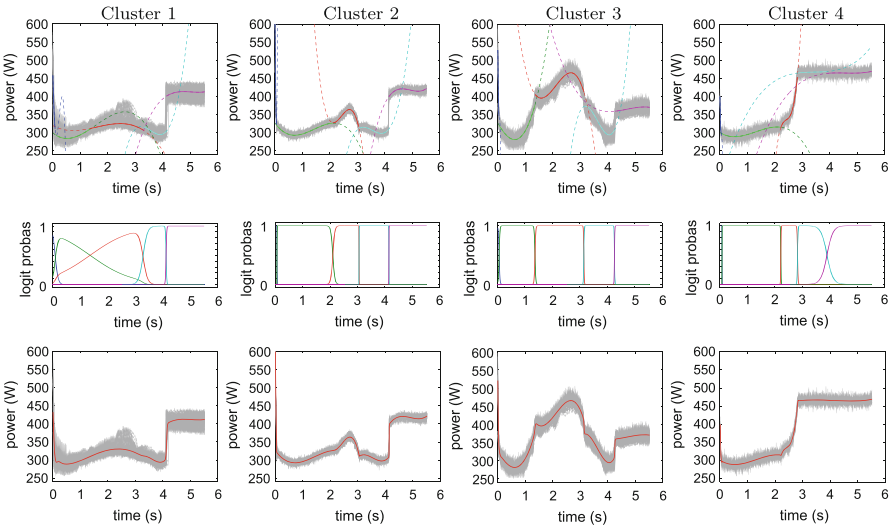

Fig. 11 Clustering results provided by the ClustSeg model applied with $K = 4$, $L = 5$ and $p = 3$: (*top*) clusters with their estimated polynomials, (*middle*) logistic probabilities and (*bottom*) estimated prototype series

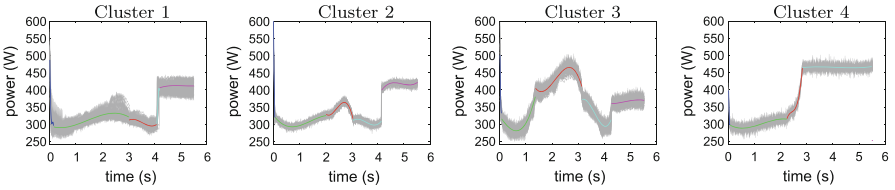

Fig. 12 Clusters and prototype series estimated by the PWR approach

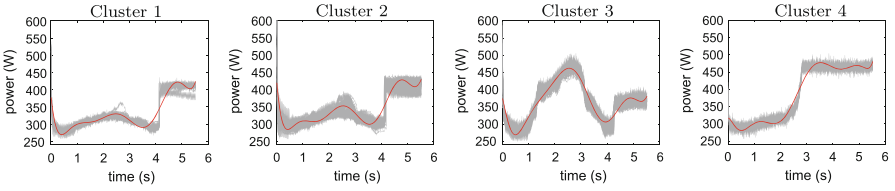

Fig. 13 Clusters and prototype series estimated by the RegMix model

close, the ClustSeg approach provides the smallest intra-cluster inertia and misclassification rate. The clusters obtained by the three compared approaches are displayed on Figs. 11, 12 and 13. The switching phases 1 and 5, which are the most abrupt ones, have been found to be well identified by the ClustSeg approach. The phases 2, 3 and 4 have been well identified only for clusters 1 and 3 and not for clusters 2 and 4, due to their specific defect nature. Nevertheless, the segmentation results on the latter clusters were found to be relevant regarding the defects localizations.

5 Conclusion

A new mixture model-based approach for the clustering of univariate time series with changes in regime has been proposed in this paper. This approach involves modeling each cluster using a particular regression model whose polynomial coefficients vary over time according to a discrete hidden process. The transition between regimes is smoothly controlled by logistic functions. The model parameters are estimated by the maximum likelihood method, solved by a dedicated Expectation–Maximization algorithm. The proposed approach can also be regarded as a clustering approach which operates by finding groups of time series having common changes in regime. The Bayesian Information Criterion (BIC) is used to determine the numbers of clusters and segments, as well as the regression order. The experimental results, both from simulated time series and from a real-world application, show that the proposed approach is an efficient means for clustering univariate time series with changes in regime. In the framework of model selection, a prospect of this work will be to derive an Integrated Classification Likelihood (ICL) type criterion (Biernacki et al. 2000) which is known to be suited to the clustering and segmentation objectives.

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

Appendix A: Convexity of the set $E_{k\ell}$

The set $E_{k\ell}$ defined by:

$$E_{k\ell} = \left\{ t \in [t_1; t_m] \quad | \quad \kappa_{k\ell}(t; \boldsymbol{\alpha}_k) = \max_{1 \leq h \leq L} \kappa_{kh}(t; \boldsymbol{\alpha}_k) \right\}.$$

is a convex set of $\mathbb{R}$. In fact, we have the following equalities:

$$\begin{aligned} E_{k\ell} &= \left\{ t \in [t_1; t_m] \quad | \quad \kappa_{k\ell}(t; \boldsymbol{\alpha}_k) = \max_{1 \leq h \leq L} \kappa_{kh}(t; \boldsymbol{\alpha}_k) \right\} \\ &= \{ t \in [t_1; t_m] \quad | \quad \kappa_{kh}(t; \boldsymbol{\alpha}_k) \leq \kappa_{k\ell}(t; \boldsymbol{\alpha}_k) \ \text{ for } h = 1, \ldots, L \} \\ &= \bigcap_{1 \leq h \leq L} \{ t \in [t_1; t_m] \quad | \quad \kappa_{kh}(t; \boldsymbol{\alpha}_k) \leq \kappa_{k\ell}(t; \boldsymbol{\alpha}_k) \} \\ &= \bigcap_{1 \leq h \leq L} \left\{ t \in [t_1; t_m] \quad | \quad \ln \frac{\kappa_{kh}(t; \boldsymbol{\alpha}_k)}{\kappa_{k\ell}(t; \boldsymbol{\alpha}_k)} \leq 0 \right\} \end{aligned}$$

From the definition of $\kappa_{k\ell}(t; \boldsymbol{\alpha}_k)$ (see Eq. 6), it can be easily verified that $\ln \frac{\kappa_{kh}(t; \boldsymbol{\alpha}_k)}{\kappa_{k\ell}(t; \boldsymbol{\alpha}_k)}$ is a linear function of t. Consequently, $E_{k\ell}$ is convex, as the intersection of convex sets of $\mathbb{R}$.

References

Banfield JD, Raftery AE (1993) Model-based gaussian and non-gaussian clustering. Biometrics 49:803–821

Biernacki C, Celeux G, Govaert G (2000) Assessing a mixture model for clustering with the integrated completed likelihood. IEEE Trans Pattern Anal Mach Intell 22(7):719–725

Celeux G, Govaert G (1995) Gaussian parsimonious clustering models. Pattern Recogn. 28(5):781–793

Chamroukhi F, Samé A, Govaert G, Aknin P (2010) A hidden process regression model for functional data description. application to curve discrimination. Neurocomputing 73:1210–1221

Chiou J, Li P (2007) Functional clustering and identifying substructures of longitudinal data. J Royal Stat Soc Ser B (Stat Methodol) 69(4):679–699

Coke G, Tsao M (2010) Random effects mixture models for clustering electrical load series. J Time Ser Anal 31(6):451–464

Dempster AP, Laird NM, Rubin DB (1977) Maximum likelihood from incomplete data via the em algorithm (with discussion). J Royal Stat Soc B 39:1–38

Gaffney S, Smyth P (1999) Trajectory clustering with mixtures of regression models. In: Proceedings of the fifth ACM SIGKDD international conference on knowledge discovery and data mining. ACM Press, San Diego, CA, USA

Gaffney S, Smyth P (2003) Curve clustering with random effects regression mixtures. In: Proceedings of the ninth international workshop on artificial intelligence and statistics, society for artificial intelligence and statistics, Key West, Florida, USA

Green P (1984) Iteratively reweighted least squares for maximum likelihood estimation, and some robust and resistant alternatives. J Royal Stat Soc B 46(2):149–192

Hébrail G, Hugueney B, Lechevallier Y, Rossi F (2010) Exploratory analysis of functional data via clustering and optimal segmentation. Neurocomputing 73(7–9):1125–1141

James G, Sugar C (2003) Clustering for sparsely sampled functional data. J Am Stat Assoc 98(462):397–408

Liu X, Yang M (2009) Simultaneous curve registration and clustering for functional data. Comput Stat Data Anal 53(4):1361–1376

McLachlan GJ, Krishnan K (2008) The EM algorithm and extension, 2nd edn. Wiley, New York

McLachlan GJ, Peel D (2000) Finite mixture models. Wiley, New York

Ng S, McLachlan G, Wang K, Ben-Tovim Jones L, Ng S (2006) A mixture model with random-effects components for clustering correlated gene-expression profiles. Bioinformatics 22(14):1745

Ramsay JO, Silverman BW (1997) Fuctional data analysis. Springer Series in Statistics, Springer, New York

Schwarz G (1978) Estimating the number of components in a finite mixture model. Ann Stat 6:461–464

Shi J, Wang B (2008) Curve prediction and clustering with mixtures of gaussian process functional regression models. Stat Comput 18(3):267–283

Wong C, Li W (2000) On a mixture autoregressive model. J Royal Stat Soc Ser B Stat Methodol 62(1):95–115

Xiong Y, Yeung D (2004) Time series clustering with arma mixtures. Pattern Recogn 37(8):1675–1689