



Biclustering multivariate discrete longitudinal data

M. Alfó¹ · M. F. Marino² · F. Martella¹

Received: 17 October 2022 / Accepted: 25 August 2023
© The Author(s) 2023

Abstract

A model-based biclustering method for multivariate discrete longitudinal data is proposed. We consider a finite mixture of generalized linear models to cluster units and, within each mixture component, we adopt a flexible and parsimonious parameterization of the component-specific canonical parameter to define subsets of variables (segments) sharing common dynamics over time. We develop an Expectation-Maximization-type algorithm for maximum likelihood estimation of model parameters. The performance of the proposed model is evaluated on a large scale simulation study, where we consider different choices for the sample size, the number of measurement occasions, the number of components and segments. The proposal is applied to Italian crime data (font ISTAT) with the aim to detect areas sharing common longitudinal trajectories for specific subsets of crime types. The identification of such biclusters may potentially be helpful for policymakers to make decisions on safety.

Keywords Finite mixtures · Model-based clustering · Three-way data · Generalized linear models · EM algorithm

1 Introduction

Clustering (Kaufman and Rousseeuw 2009; Everitt et al. 2011; Hennig et al. 2015; Wierzchoń and Kłopotek 2018) identifies a family of methods aiming at discovering “meaningful” subsets in the observed sample. Given a data matrix with n rows (units) and p columns (variables), the traditional goal of clustering is to identify subsets such that units are as similar as possible within and as different as possible between clusters. Biclustering (Good 1965; Hartigan 1972, 1975; Bock 1979) represents an extension of the standard clustering approach defined to jointly partition the set of units and variables of a data matrix into homogeneous blocks, the biclusters. Referred to as block clustering, bidimensional clustering, two-way clustering, two-mode or two-side clustering, direct clustering, cross-clustering, etc., during the past decades, biclustering has been applied in several scientific

fields to analyze large data matrices where the role of the two modes of the matrix (rows and columns) may be thought of as being symmetric. Just to give a few examples, biclustering techniques have been used in text mining, webmining, bioinformatics, marketing, ecology, computer science. Literature on biclustering is now quite extensive and is usually separated into exploratory and model-based approaches. For more details, the interested reader may refer to reviews by Madeira and Oliveira (2004) and Brault and Lomet (2015). Martella and Alfò (2017) discussed available software implementations.

Looking at model-based approaches, only in recent times biclustering methods have been proposed to deal with discrete data. Examples include Arnold et al. (2010) and Pledger and Arnold (2014) for binary and count data, respectively. The latent block mixture model for binary data and its extension to contingency table by Govaert and Nadif (2003, 2008) and Govaert and Nadif (2010) are further examples, together with the corresponding Bayesian version by Wyse and Friel (2012). Priam et al. (2008, 2014) proposed instead a combination of the Bernoulli block mixture models with probabilistic self-organizing maps to analyze high-dimensional binary data. Li and Zha (2006) developed a two-way Poisson mixture model in the context of text analysis, while Lee and Huang (2014) proposed an algorithm for binary data based on a penalized Bernoulli likelihood. Vicari

✉ M. F. Marino
mariafrancesca.marino@uniroma1.it

M. Alfó
marco.alfó@uniroma1.it

F. Martella
francesca.martella@uniroma1.it

¹ Sapienza, University of Rome, Rome, Italy

² University of Florence, Florence, Italy

and Alfó (2014) defined a mixture model for the joint clustering of customers and products, while Martella and Alfó (2017) introduced a re-parameterization of the mixture of factor analyzers (see Ghahramani et al. 1996) to deal with discrete data. Further recent advances on model-based biclustering for discrete data can also be found in Fernández et al. (2019). Some methods, such as those based on the latent block mixture model, assume that the partition of rows and of columns are independent. That is, the segments (clusters of columns) are constant across components (clusters of rows). In a customer by product context, this means that the groups of products the costumers tend to associate are always the same. When this hypothesis does not hold, potential dependence between the two partitions should be considered; see Vicari and Alfó (2014) and Martella and Alfó (2017).

All these methods deal with multivariate data observed at a single occasion. In this paper, we extend model-based biclustering to multivariate discrete data that are repeatedly observed over time. As these represent a specific three-way data example, we feel that Table 1, taken from Viroli (2011b), may help. Depending on the nature of the three modes, different data structures may, in fact, be considered. In the case of multivariate longitudinal data, the three modes (rows, columns, and layers) are represented by units, variables, and times, respectively.

A number of techniques have been developed for clustering three-way data ranging from sequential to simultaneous clustering and data reduction procedures. Examples are Gordon and Vichi (1998), Vichi (1999), and Vichi et al. (2007). A first attempt to apply model-based techniques can be found in Basford and McLachlan (1985) while, more recently, we can mention, Hunt and Basford (1999), Vermunt (2007), Viroli (2011a, b), and Bruckers et al. (2016). Up to our knowledge, only few methods for biclustering three-way data have been proposed. Turner et al. (2005) extended the plaid model, proposed by Lazzeroni and Owen (2002), to longitudinal data, while Mankad and Michailidis (2014) generalized this proposal to handle data observed in different experimental conditions or time occasions, and discussed interpolation at unobserved points. In the recent years, biclustering has appeared also in the context of functional data; see Zhao et al. (2004). Examples include the co-clustering approach proposed by Slimen et al. (2018) and Bouveyron et al. (2018), which extends the proposal by Govaert and Nadif (2013) to functional data. Both these semi-parametric models hold the typical feature of standard latent block models: unit- and variable-specific partitions are independent. That is, these methods provide a *grid* clustering where the partition of variables is constant for each component of the finite mixture. The `funLBM` R package (Bouveyron et al. 2022) was recently released to estimate the parameters of the model proposed by Bouveyron et al. (2018). As stated above, such

a grid clustering may not be appropriate in situations where the two partitions are somewhat dependent. To answer to such an issue, Galvani et al. (2021) proposed a method that adapts the Cheng and Church algorithm (Cheng and Church 2000) to deal with functional data. The novelty of such a non-parametric biclustering approach is that it allows for data misalignment and for the existence of curves that do not belong to any block (i.e. non-exhaustive biclustering). The method is implemented in the `FunCCR` package (Torti et al. 2020). Both these methods were discussed, and likely derived for, continuous data.

In this paper, we propose a model-based biclustering approach for multivariate discrete longitudinal data, where clusters of units are defined to share common longitudinal trajectories for specific subsets of variables. The latter may vary with row clusters. The method is designed to deal with longitudinal data, originating from a limited number of measurement occasions in discrete time, rather than for longer sequences observed in a (quasi-) continuous time, i.e., functional data. Within each mixture component, the canonical parameter is modeled by a suitable parametrization that helps identify subsets/groups of variables evolving over time in a similar manner for units in that cluster. Figure 1 may help understand the basic features of the proposal. It represents a toy example where 100 units are partitioned into 3 components and, within each of them, 4 variables are partitioned into 2 segments that show a similar longitudinal profile over 4 time points. In detail, variables allocated to the first segment remain rather constant over time, with a slight increase at the last occasion; on the other side, variables belonging to the second segment increase exponentially over time. These trends become more pronounced when moving from the first to the last component of the finite mixture. In this sense, the proposed model can be seen as a longitudinal extension of the model proposed by Martella and Alfó (2017). The paper is structured as follows. In Sect. 2, we briefly recall the approach introduced by Martella and Alfó (2017). The extension to longitudinal data and a discussion on model identifiability are illustrated in Sect. 3. Section 4 describes the EM algorithm for maximum likelihood estimation of model parameters; in Sect. 5, initialization, convergence of the EM algorithm, and model selection are discussed. Section 6 presents a large scale simulation study where we analyze the ability of the proposed approach in recovering the true partitions and the true values of model parameters in several controlled settings. In Sect. 7, we apply the proposed model to a real-life dataset describing crime events in Italy during the period 2012–2019 (source: ISTAT). After an exploratory analysis, we present the main results obtained by the proposed model and highlight its potentialities. Final conclusions are drawn in Sect. 8.

Table 1 Some common three-way data structures

Three-way structure	Mode		
	Rows	Columns	Layers
Multivariate longitudinal data	Units	Variables	Times
Multivariate repeated measures	Units	Variables	Situations
Multivariate spatial data	Units	Variables	Locations
Multivariate time-series	Units/locations	Times	Variables
Multivariate spatio-temporal data	Locations	Variables	Times
...

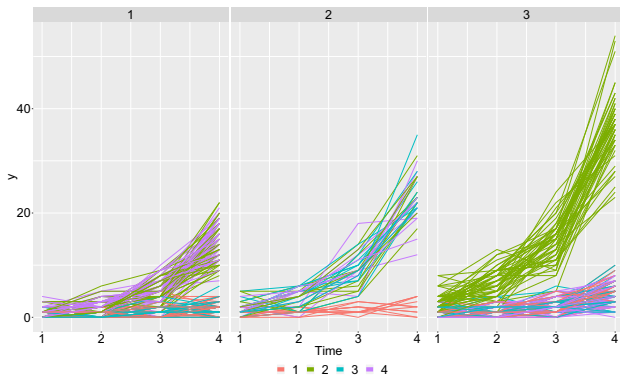


Fig. 1 An example of biclustering with 100 units, 4 variables, 3 components, 2 segments, and 4 time occasions

2 Biclustering multivariate discrete data

Martella and Alfò (2017) proposed a model for biclustering discrete outcomes, starting from a specific version of a finite mixture of factor analyzers. To identify homogeneous subsets of variables they used a suitable choice of the factor loading matrix to represent segment membership. In detail, let n and p denote the number of observed rows (units) and columns (variables) of the data matrix, respectively. For a given unit, the vector $\mathbf{y}_i = (y_{i1}, \dots, y_{ip})'$ is observed; its generic element y_{ij} represents the value of the j -th variable for the i -th unit, $i = 1, \dots, n, j = 1, \dots, p$. As it is usual in model-based clustering, we assume that \mathbf{y}_i is drawn from a population \mathcal{P} formed by K non-overlapping subpopulations $\mathcal{P}_k, k = 1, \dots, K$, identified by a unit-specific component indicator $\mathbf{Z}_i = (Z_{i1}, \dots, Z_{iK})'$, with $Z_{ik} = 1$ if unit i comes from the k -th subpopulation and $Z_{ik} = 0$ otherwise. The prior probability for the i -th unit, $i = 1, \dots, n$, to come from subpopulation \mathcal{P}_k is denoted by $\pi_k = \Pr(Z_{ik} = 1)$, where the usual constraints $0 < \pi_k \leq 1, k = 1, \dots, K$, and $\sum_{k=1}^K \pi_k = 1$ hold. Without loss of generality, let us consider that responses y_{ij} are realizations of independent random variables with density in the Exponential Family (EF). The joint (conditional) density of the vector \mathbf{y}_i may be expressed

as follows:

$$\begin{aligned}
 f(\mathbf{y}_i | \mathbf{Z}_{ik} = 1) &= \prod_{j=1}^p f(y_{ij} | \theta_{j(k)}, \sigma_{j(k)}) \\
 &= \prod_{j=1}^p \exp \left\{ \frac{y_{ij} \theta_{j(k)} - c(\theta_{j(k)})}{b(\sigma_{j(k)})} + d(y_{ij}; \sigma_{j(k)}) \right\}. \quad (1)
 \end{aligned}$$

Here, $f(y_{ij} | \theta_{j(k)}, \sigma_{j(k)})$ represents a generic density in the EF for the j -th response in the k -th component; this is characterized by the canonical parameter $\theta_{j(k)}$ and the dispersion parameter $\sigma_{j(k)}$. Further, $b(\cdot), c(\cdot), d(\cdot)$ denote known functions. As it can be observed, heterogeneity induces marginal dependence; that is, the multivariate model is defined as a finite mixture of conditional multivariate densities, in turn defined as the product of conditionally independent univariate densities. Even if this assumption may be overly simple, as the corresponding dependence structure is exchangeable at the linear predictor scale, it still gives a solution to the lack of simple multivariate models for discrete data and gives rise to a model which is simple to be interpreted and handled. Martella and Alfò (2017) focused on Binomial, Poisson, and Negative Binomial distributions as component-specific models.

To introduce partitioning of variables for units in the k -th component of the finite mixture (i.e., to identify segments), we assume that the parameters $\theta_{j(k)}, j = 1, \dots, J, k = 1, \dots, K$, are described by the following equation

$$\theta_{j(k)} = \phi_k + \mathbf{a}'_{jk} \boldsymbol{\beta}.$$

Here, ϕ_k is a component-specific intercept, $\boldsymbol{\beta}$ is a Q -dimensional vector of effects with elements β_1, \dots, β_Q , while \mathbf{a}_{jk} is a Q -dimensional ($Q \leq p$) membership indicator for the j -th variable in the k -th component. In detail, the terms $\beta_q, q = 1, \dots, Q$, are fixed effects associated to subsets of variables. That is, if two variables share the same effect β_q , they belong to the same segment (within the k -th component of the finite mixture). On the other hand, the

indicator variables in \mathbf{a}_{jk} are defined as

$$a_{jkq} = \begin{cases} 1 & \text{if variable } j \text{ is in segment } q, \\ 0 & \text{otherwise,} \end{cases}$$

for $j = 1, \dots, J, k = 1, \dots, K, q = 1, \dots, Q$. This is the main difference with respect to standard grid-clustering (co-clustering) techniques; there, a *rigid* partition built up by clusters of rows and columns is assumed. Here, instead, we allow for a different number of variable-specific segments in each mixture component. For example, a subset of columns may be empty in a given component and non-empty in another one. That is, for some components $k = 1, \dots, K$, a specific segment $q = 1, \dots, Q$, may be empty and the corresponding element a_{jkq} be null. So, for that component the number of segments Q reduces to $Q_{(k)} \leq Q$. Further, we do not need to assume independence between the unit and the variable-specific partitions. The model is essentially based on projecting, conditional on the cluster of units, the p observed variables onto a reduced space of dimension $Q \leq p$.

3 The extension to longitudinal data

Let us suppose we observe for a sample of n units a p -dimensional set of variables at T measurement occasions. Let y_{ijt} denote the value of the j -th variable at occasion t for unit $i, i = 1, \dots, n, j = 1, \dots, p, t = 1, \dots, T$. The vector $\mathbf{y}_{it} = (y_{i1t}, \dots, y_{ipt})'$ is the i -th unit vector of variables recorded at occasion t ($i = 1, \dots, n, t = 1, \dots, T$), while $\mathbf{y}_i = (\mathbf{y}_{i1}, \dots, \mathbf{y}_{iT})$ denotes the matrix of values recorded over all occasions, $i = 1, \dots, n$. Here, for simplicity, we assume that all units are observed at all the intended measurement occasions, as in *balanced* longitudinal studies. The generalization to the unbalanced case is straightforward under a Missing at Random (Rubin 1976) assumption.

To extend the model introduced in the previous section to the longitudinal framework, we assume that \mathbf{y}_i is drawn from a population \mathcal{P} formed by K subpopulations $\mathcal{P}_k, k = 1, \dots, K$, identified by a unit-specific component indicator $\mathbf{Z}_i = (Z_{i1}, \dots, Z_{iK})'$, with $Z_{ik} = 1$ if unit i comes from the k -th subpopulation and $Z_{ik} = 0$ otherwise; the prior probabilities $\pi_k = \Pr(Z_{ik} = 1), k = 1, \dots, K$, follow the usual constraints $0 < \pi_k \leq 1, k = 1, \dots, K$, and $\sum_{k=1}^K \pi_k = 1$. As it can be noticed, units' partition is fixed over time. We assume local independence over variables and times; according to such an assumption, conditional on $Z_{ik} = 1$, the p variables are independent, both within (same variable over time) and between (different variables at a given occasion). That is, we assume that the unit-specific component indicator \mathbf{Z}_i captures the dependence both between and within variables recorded at different occasions from the i -th statistical unit. As before, even if this assumption may be overly restrictive, it is due to the lack of a proper multivariate model

for discrete data. Based on such assumptions, the conditional density (1) may be extended in a natural way as

$$f(\mathbf{y}_i | \mathbf{Z}_{ik} = 1) = \prod_{t=1}^T \prod_{j=1}^p f(y_{ijt} | \theta_{jt(k)}, \sigma_{j(k)}) \\ = \prod_{t=1}^T \prod_{j=1}^p \exp \left\{ \frac{y_{ijt} \theta_{jt(k)} - c(\theta_{jt(k)})}{b(\sigma_{j(k)})} + d(y_{ijt}; \sigma_{j(k)}) \right\}, \tag{2}$$

where $\theta_{jt(k)}$ represents the canonical parameter for the j -th variable at occasion t for units in the k -th component, while $\sigma_{j(k)}$ is the component-specific dispersion parameter for the j -th variable, assumed to be constant across occasions.

To introduce a partitioning of observed variables, different parameterizations can be considered. We may start noticing that the proposal by Martella and Alfò (2017) can be applied to cases with $T = 1$. For $T > 1$, several extensions can be obtained by opportunely specifying the canonical parameter $\theta_{jt(k)}$. The simplest case consists in assuming that variable partitioning remains constant over time. That is,

$$\theta_{jt(k)} = \theta_{j(k)} = \phi_k + \mathbf{a}'_{jk} \boldsymbol{\beta}, \\ j = 1, \dots, p \text{ and } k = 1, \dots, K, \text{ as in the non-longitudinal case discussed so far. A step forward is that of allowing the variable partition to vary with time:} \\ \theta_{jt(k)} = \phi_k + \mathbf{a}'_{jkt} \boldsymbol{\beta}, \tag{3}$$

$j = 1, \dots, p, k = 1, \dots, K$ and $t = 1, \dots, T$. Here, the vector \mathbf{a}_{jkt} represents the occasion-specific membership for the j -th variable and the k -th component. Therefore, partitioning of variables may vary with occasions and components. Note that, in equation (3), segment membership changes with time, while the corresponding effects in the vector $\boldsymbol{\beta}$ remain constant. While this approach is more general, it assumes that a constant vector $\boldsymbol{\beta}$ may well represent the evolution over time of the observed variables. In this sense, we may expect that a few values of $\boldsymbol{\beta}$ are associated with a subset of occasions only and therefore, a higher number of segments Q is needed in the presence of highly variable longitudinal trajectories.

An alternative approach consists in considering the following parameterization:

$$\theta_{jt(k)} = \phi_k + \mathbf{a}'_{jk} \boldsymbol{\beta}(t). \tag{4}$$

Here, membership of the p variables is constant over time within the k -th component, while the vector of segment-specific effects may vary. In detail, we consider a Q -dimensional function of time $\boldsymbol{\beta}(t) = (\beta_1(t), \dots, \beta_Q(t))'$,

where $\beta_q(t)$ is a process describing segment-specific dynamics. According to this parameterization, unit membership does not change over time, while the associated effect, say $\beta_q(t)$, may evolve over time and describe different dynamics for each segment. Clearly, various specifications may be considered for the vector $\beta(t)$ as well. In the following section, we discuss a few examples.

3.1 The time functions

Starting from the parameterization in equation (4), different time functions may be considered. While many alternatives are possible, we focus on a few that we think can properly approximate very different time behaviors.

3.1.1 Polynomial time functions

A simple way to model segment-specific functions $\beta_q(t)$ is to use a polynomial of a given degree R :

$$\beta_q(t) = \sum_{r=0}^R \lambda_{qr} t^r,$$

where $\{\lambda_{qr}, r = 0, \dots, R\}$ represent the polynomial coefficients, with $\lambda_{q0} = 0$ for identifiability reasons. In this case, equation (4) can be rewritten as:

$$\theta_{jt(k)} = \phi_k + \mathbf{a}'_{jk} \mathbf{A} \omega(t) \tag{5}$$

where $\omega(t) = (1, t, t^2, \dots, t^R)'$ is an $(R + 1)$ -dimensional design vector and $\mathbf{A} = (\lambda_1, \dots, \lambda_Q)'$ is a $Q \times (R + 1)$ matrix of polynomial coefficients. Each row λ_q , is an $(R + 1)$ -dimensional vector containing segment-specific coefficients $\lambda_{qr}, r = 0, \dots, R$. For a large enough degree R , the polynomial function may approximate any function of the time. Clearly, different choices of the degree R lead to different polynomials. Usually, R is fixed to be as low as possible (smaller than 3 or 4); in fact, for larger values, the polynomial curve may be overly flexible and this may make the corresponding estimates unstable. See James et al. (2013) for a discussion.

3.1.2 Basis time functions

To add some flexibility and define a smooth segment-specific function of time, we may refer to a more general approach in which polynomials are a special case. The idea is to represent the segment-specific process $\beta_q(t)$ by a linear combination of L known basis functions $\{\gamma_1(t), \dots, \gamma_L(t)\}$:

$$\beta_q(t) = \lambda_{q0} + \sum_{l=1}^L \lambda_{ql} \gamma_l(t). \tag{6}$$

Here, λ_{q0} is an intercept constrained to be zero for identifiability reasons and $\lambda_{ql}, l = 1, \dots, L$, are coefficients associated to the basis functions $\gamma_1(t), \dots, \gamma_L(t)$. Note that, in the polynomial case, the basis functions are simply $\gamma_l(t) = t^l$. Besides this, several other basis functions may be used; the most common choice are the spline basis functions. These represent a collection of piecewise polynomials that connect smoothly in the abscissas (here, the time) at specified points $\rho_1 < \rho_2 < \dots < \rho_D$ (called knots). The total number of basis functions, L , depends on the number of knots, D : the higher the latter, the higher the flexibility. Spline functions are known to produce more stable estimates when compared to polynomials (Hastie et al. 2009).

Without loss of generality, we will consider the R -th degree truncated power basis function, although the model can be extended to other basis functions as well (B-splines, cubic splines, Fourier, etc.). For an overview of the potential alternatives, we refer to Green and Silverman (1993), Hastie and Tibshirani (1990) and, Wood (2017).

The R -th degree truncated power basis function with knots at ρ_1, \dots, ρ_D is defined as

$$1, t, \dots, t^R, (t - \rho_1)_+^R, \dots, (t - \rho_D)_+^R, \tag{7}$$

where $(t - \rho_d)_+^R, d = 1, \dots, D$, is given by

$$(t - \rho_d)_+^R = \begin{cases} 0 & t < \rho_d, \\ (t - \rho_d)^R & t \geq \rho_d. \end{cases}$$

By increasing the number of knots, D , we increase the number of basis and, thus, the flexibility of the model. The total number of basis functions depends on the number of knots given that $L = R + 1 + D$. If we adopt the R -th degree truncated power basis in equation (7), we may re-write the segment-specific function $\beta_q(t)$ in equation (6) as

$$\beta_q(t) = \sum_{r=0}^R \lambda_{qr} t^r + \sum_{d=1}^D \lambda_{qR+d} (t - \rho_d)_+^R. \tag{8}$$

The parametrization of the canonical parameter, $\theta_{jt(k)}$, is the same as in equation (5). The term $\omega(t)$ now denotes the L -dimensional design vector $\omega(t) = (1, t^1, t^2, \dots, t^R, (t - \rho_1)_+^R, \dots, (t - \rho_D)_+^R)'$ and \mathbf{A} is a $Q \times L$ matrix of coefficients $(\lambda_1, \dots, \lambda_Q)'$.

Choosing the number of basis functions, as well as the number and the position of knots, is usually a challenging task. A potential approach is that of using the smoothing spline regression, where we set the number of knots equal to the number of time points and the resulting over-fitting is controlled by adding a penalty term to the objective function. Such a method has usually a high computational complex-

ity due to both the penalty choice and the number of time occasions (Coffey et al. 2014).

As an alternative, we may use the penalized spline smoothing (P-spline; Eilers and Marx, 1996). This is a low-rank smoothing method requiring a lower number of basis functions. It tries to avoid over-fitting by using a discrete penalty, ν , called the smoothing parameter, on the estimated coefficients. Representing the smoothing problem by P-splines reduces its dimensionality and, thus, the computational complexity. Ruppert (2002) argued that P-splines are also relatively insensitive to the number of basis functions selected once a sufficient number is used. He suggested to choose the number of knots via the rule-of-thumb $D = \max(5, \min(\frac{T}{4}, 35))$ and to place knots at the quantiles of the data.

To simplify estimation of the basis function coefficients, we may make use of the generalized linear mixed model representation of P-splines, and rewrite the segment-specific vector as $\beta(t) = \Lambda\omega(t) + \Xi u(t)$. Accordingly, equation (5) becomes

$$\theta_{jt(k)} = \phi_k + \mathbf{a}'_{jk} [\Lambda\omega(t) + \Xi u(t)], \tag{9}$$

where $\omega(t) = (1, t^1, t^2, \dots, t^R)'$ now corresponds to the first $(R+1)$ basis functions, $\mathbf{u}(t) = ((t - \rho_1)_+^R, \dots, (t - \rho_D)_+^R)'$ consists of the remaining D basis functions, while Λ and Ξ are matrices of unknown coefficients associated to $\omega(t)$ and $\mathbf{u}(t)$, respectively. The generic row of Ξ corresponds to the D -dimensional random effects $\xi_q \sim N(\mathbf{0}, \sigma_\xi^2 \mathbf{I}_D)$ ($q = 1, \dots, Q$), such that $\sigma_\xi^2 = 1/\nu$, where, as stated above, ν controls the trade-off between fit and smoothness. Estimates of Λ , Ξ , and σ_ξ^2 are solutions to generalized linear mixed model equations, so that we may take advantage and use ML implementations available in standard statistical software.

3.2 Model identifiability

Model identifiability is an important issue in this context. As detailed by Martella and Alfò (2017), it is a twofold problem depending on both the mixture model and the canonical parameter re-parameterization. The identifiability of the mixture model can be established following Teicher (1961, 1963). These works provide a foundation for finite mixtures of parametric distributions. See also McLachlan and Peel (2000) for a general discussion, Yakowitz and Spragins (1968) and Atienza et al. (2006) for specific contributions. As far as the identifiability of parameters defining $\theta_{jt(k)}$, we may proceed as follows: since the vector \mathbf{a}_{jk} is binary and row stochastic, the elements in $\beta(t)$ can be easily identified. In fact, the vector \mathbf{a}_{jk} is unique, but for trivial labels permutation. For fixed \mathbf{a}_{jk} and $\beta(t)$, $k = 1, \dots, K$, $j = 1, \dots, p$, to identify ϕ_k , $k = 1, \dots, K$, we may introduce

the usual constraint $\sum_{k=1}^K \pi_k \phi_k = 0$, that is a typical identifiability condition in regression models with unit-specific discrete random effects. Thus, conditions given in Hennig (2000) ensure identifiability of the parameters in $\beta(t)$ and $\phi = (\phi_1, \dots, \phi_K)'$ appearing in the model for $\theta_{jt(k)}$. Further details can be found in Martella and Alfò (2017).

4 Maximum likelihood estimation

The observed log-likelihood for n independent observations may be expressed as follows

$$\begin{aligned} \ell(\Psi) &= \sum_{i=1}^n \log f(y_i | \Psi) \\ &= \sum_{i=1}^n \log \sum_{k=1}^K \pi_k f(y_i | Z_{ik} = 1), \end{aligned}$$

where Ψ represents the vector of model parameters and $f(y_i | Z_{ik} = 1)$ is defined according to equation (2) with a canonical parameter specified according to either equation (5) or (9). If the latter is considered, depending on the time function we adopt to describe dynamics over time, we have a different number of parameters to estimate. However, regardless the choice, we exploit the EM algorithm (Dempster et al. 1977) to compute the ML estimate for Ψ . Considering the component indicators Z_i 's as missing data and adopting for them a Multinomial distribution with index 1 and parameter $\pi = (\pi_1, \dots, \pi_K)'$, the complete-data log-likelihood function for Ψ is

$$\begin{aligned} \ell_c(\Psi) &= \sum_{i=1}^n \sum_{k=1}^K z_{ik} \log(\pi_k) \\ &+ \sum_{i=1}^n \sum_{k=1}^K \sum_{j=1}^p \sum_{t=1}^T z_{ikt} \log f(y_{ijt} | Z_{ik} = 1). \end{aligned}$$

Let $\mathbf{W} = \{w_{ik}\}_{(k=1, \dots, K, i=1, \dots, n)}$ denote the matrix of posterior probabilities of units' component membership, and let $\mathbf{A} = \{\mathbf{a}_{jk}\}_{(k=1, \dots, K, j=1, \dots, p)}$ be the matrix of binary vectors that define, within each component, the segment the j -th variable belongs to. The EM algorithm proceeds by initializing \mathbf{A} and \mathbf{W} to $\hat{\mathbf{A}}_0$ and $\hat{\mathbf{W}}_0$ and iterating the following steps until convergence.

1. *M-step 1: update $\beta(t)$, $\phi = (\phi_1, \dots, \phi_K)'$, and $\sigma = (\sigma_{1(1)}, \dots, \sigma_{p(K)})'$*
 Conditional on $\hat{\mathbf{A}}$ and $\hat{\mathbf{W}}$, the likelihood is a weighted version of a standard generalized linear (mixed-effects in the case of P-spline) model likelihood. As no closed form solutions are available, we update $\hat{\beta}(t)$ $t = 1, \dots, T$, $\hat{\phi}$,

and $\hat{\sigma}$ via a Newton–Raphson algorithm on the augmented data $(\mathbf{Y}, \hat{\mathbf{W}}, \hat{\mathbf{A}})$.

2. *M-step 2: update π*

Conditional on $\hat{\mathbf{A}}$ and $\hat{\mathbf{W}}$, update $\hat{\pi}$, via the standard finite mixture solutions

$$\hat{\pi}_k = \frac{\sum_{i=1}^n \hat{w}_{ik}}{n} \quad k = 1, \dots, K.$$

3. *M-step 3: update \mathbf{A}*

Conditional on $\hat{\mathbf{W}}, \hat{\beta}(t), \hat{\phi}, \hat{\sigma}$, and $\hat{\pi}$, allocate units into components according a Maximum A Posteriori (MAP) rule and update $\hat{\mathbf{A}}$ as follows

- (a) consider the j -th variable and the k -th component, $j = 1, \dots, p, k = 1, \dots, K$, and compute the log-likelihood contribution

$$\ell_{j(kq)} = \sum_{i \in k} \sum_{t=1}^T \frac{y_{ijt} \hat{\theta}_{jt(kq)} - c(\hat{\theta}_{jt(kq)})}{b(\hat{\sigma}_{j(k)})} + d(y_{ijt}; \hat{\sigma}_{j(k)}),$$

with $\hat{\theta}_{jt(kq)} = \hat{\phi}_k + \hat{\beta}_q(t)$ and $q = 1 \dots, Q$;

- (b) for fixed j and k , compute the maximum log-likelihood value $\ell_{j(kq)}$ over $q = 1, \dots, Q$, and denote it by $\ell_{j(k)}^{max}$;
- (c) for the k -th component, allocate the j -th variable to the q -th segment (by setting $a_{jkq} = 1$) if $\ell_{j(kq)} = \ell_{j(k)}^{max}$, for $j = 1, \dots, p, k = 1, \dots, K$

4. *E-step: update \mathbf{W}*

Conditional on $\hat{\Psi}$, update $\hat{\mathbf{W}}$ by computing

$$\hat{w}_{ik} = \frac{\hat{\pi}_k f(\mathbf{y}_i | Z_{ik} = 1)}{\sum_{k=1}^K \hat{\pi}_k f(\mathbf{y}_i | Z_{ik} = 1)},$$

for $k = 1, \dots, K$, and $i = 1, \dots, n$.

At convergence, each unit may be assigned to the component with the highest posterior probability (MAP rule), while each variable is assigned to the q -th segment according to the elements in the matrix $\hat{\mathbf{A}}$. We may notice that, while the estimated row (unit) partition is fuzzy, the column (variable) partition is assumed to be *hard*, with non-overlapping segments.

5 Initialization, convergence and model selection

Even if convergence of the EM algorithm is guaranteed when the (log-)likelihood function is bounded from above, as it is in the current case, its performance may heavily depend on starting values, since the log-likelihood surface has often multiple (local) maxima. A good initialization strategy is defined as

one that attains a good solution in a low number of iterations. While the literature is rich in proposals to find reasonable starting values, no strategy seems to uniformly outperform the others. For an overview, see Giordani et al. (2020). Random initialization of component/segment memberships may lead to slower convergence than starting from other clustering solutions, such as k-means, even though none of the two guarantees that a global maximum is attained. To this end, a multi-start strategy is usually suggested. In the simulation study, we used a multiple starting procedure based on independent k-means algorithms on the rows and the columns of the data matrix, in order to obtain an initial partition of units and variables.

As far as the stopping rule criteria is concerned, following Martella and Alfò (2017), the EM algorithm is stopped when

$$\ell^{(r)}(\cdot) - \ell^{(r-1)}(\cdot) < \epsilon, \quad \epsilon > 0,$$

where r is the current iteration and $\epsilon = 10^{-5}$. Clearly, other suitable stopping rules, like $\|\hat{\Psi}^{(r)} - \hat{\Psi}^{(r-1)}\| < \epsilon$, could be reasonable.

We assumed that the number of components and segments, as well as the degree of the polynomial are known; this is usually not the case and they must be estimated through the data. As in standard mixture models, we can make use of information criteria, where we minimize twice the negative log-likelihood function penalized by model complexity. Examples are the Akaike Information Criterion (AIC; Akaike 1973)

$$AIC = -2\ell(\hat{\Psi}) + 2\nu,$$

where ν is the number of free parameters to be estimated. Here, $\nu = K + JK + Q(R + 1) + (K - 1)$ in the polynomial case and $\nu = K + JK + QL + (K - 1)$ in the basis function case. We may also consider the Bayesian Information Criterion (BIC; Schwarz 1978) defined as

$$BIC = -2\ell(\hat{\Psi}) + \nu \log(n),$$

as well as the Integrated Completed Likelihood (ICL; Biernacki et al. 2000), which can be calculated via the following approximation

$$ICL = BIC - \sum_{k=1}^K \sum_{i=1}^n \hat{w}_{ik} \log(\hat{w}_{ik}).$$

The Normalized Entropy Criterion (NEC; Celeux and Soromenho 1996) obtained as

$$NEC = \begin{cases} \frac{-\sum_{k=1}^K \sum_{i=1}^n \hat{w}_{ik} \log(\hat{w}_{ik})}{\ell(\hat{\Psi}_{K^*}) - \ell(\hat{\Psi}_1)} & \text{if } K > 1, \\ 1 & \text{otherwise,} \end{cases}$$

represents a further alternative. Here, $\hat{\Psi}_K$ denotes the estimated parameter vector obtained with K components. For details on the performance of such criteria, see Soromenho (1994), Kass and Raftery (1995), Celeux and Soromenho (1996), McLachlan and Peel (2000), and Baudry et al. (2010), among others.

6 Simulation study

We conducted a large scale simulation study, based on varying sample sizes, number of time occasions, and number of components and segments. The aim is to evaluate the performance of the proposed biclustering method in different controlled contexts. In Sects. 6.1 and 6.2, we present the results we obtained by considering either a polynomial or a spline specification, as described in Sects. 3.1.1 and 3.1.2, respectively.

6.1 Polynomial time functions

We start by considering the case of count data with conditional Negative Binomial distribution

$$f(y_{ijt}|Z_{ik} = 1) = \binom{y_{ijt} + s_{j(k)} - 1}{s_{j(k)} - 1} \times \left(\frac{\theta_{jt(k)}}{\theta_{jt(k)} + s_{j(k)}} \right)^{y_{ijt}} \left(\frac{s_{j(k)}}{\theta_{jt(k)} + s_{j(k)}} \right)^{s_{j(k)}}. \quad (10)$$

This distribution is widely used when modeling counts as it allows for overdispersion in the data and, further, includes the Poisson distribution as a particular case.

The term $\theta_{jt(k)}$ in equation (10) denotes the component-specific canonical parameter for the j -th variable at occasion t and is modeled according to equation (5). On the other hand, the term $s_{j(k)}$ controls for variable- and component-specific overdispersion and, for simplicity, it is assumed to remain constant with j (i.e., $s_{j(k)} = s_{(k)}$, $j = 1, \dots, p$).

To study the behavior of the proposed method in controlled experiments, we considered different simulation scenarios based on varying sample sizes ($n = 50, 100, 200, 500$), number of time occasions ($T = 4, 8$) and variables ($p = 12, 24$), by keeping constant the polynomial degree to $R = 1$. We considered two alternative block structures:

- (a) $K = 3$ components and $Q = 2$ segments, with component-specific locations $\phi = (1, 2, 3)'$, segment-specific polynomial coefficients $\Lambda = (\lambda_{11}, \lambda_{21})' = (0.2, 0.8)'$, component-specific overdispersions $\mathbf{s} = (s_{(1)}, s_{(2)}, s_{(3)})' = \kappa \times (1, 1.5, 2)'$ and component prior probabilities $\boldsymbol{\pi} = (0.5, 0.1, 0.4)'$;
- (b) $K = 4$ components and $Q = 3$ segments, with component-specific locations $\phi = (1, 2, 3, 4)'$, segment-

Table 2 Polynomial time function with $R = 1$: mean and median (in brackets) ARI values for $K = 3$, $Q = 2$, and $T = 4$, in different settings

High variance			Low variance		
n	Row	Col	n	Row	Col
$p = 12$					
50	0.96 (1.00)	0.86 (1.00)	50	0.99 (1.00)	0.85 (1.00)
100	0.98 (1.00)	0.88 (1.00)	100	0.99 (1.00)	0.90 (1.00)
200	1.00 (1.00)	0.96 (1.00)	200	1.00 (1.00)	0.99 (1.00)
500	1.00 (1.00)	1.00 (1.00)	500	1.00 (1.00)	1.00 (1.00)
$p = 24$					
50	1.00 (1.00)	1.00 (1.00)	50	1.00 (1.00)	1.00 (1.00)
100	1.00 (1.00)	1.00 (1.00)	100	1.00 (1.00)	1.00 (1.00)
200	1.00 (1.00)	1.00 (1.00)	200	1.00 (1.00)	1.00 (1.00)
500	1.00 (1.00)	1.00 (1.00)	500	1.00 (1.00)	1.00 (1.00)

specific polynomial coefficients $\Lambda = (\lambda_{11}, \lambda_{21}, \lambda_{31})' = (0.2, 0.8, 1)$, component-specific overdispersions $\mathbf{s} = (s_{(1)}, \dots, s_{(4)})' = \kappa \times (1, 1.5, 2, 2.5)'$, and component prior probabilities $\boldsymbol{\pi} = (0.3, 0.2, 0.1, 0.4)'$.

As far as the variable partition is concerned, for each variable, a simple random sampling without replacement is considered. Last, under both settings (a) and (b), κ controls for heterogeneity within biclusters. In this respect, noticing that the conditional variance of y_{ijt} , given that unit i comes from the k -th component, is equal to $[\theta_{jt(k)} + \theta_{jt(k)}^2]/s_{(k)}$, we obtained *high* and *low variance* by setting $\kappa = 0.5$ and $\kappa = 1$, respectively.

For each scenario, we simulated $B = 100$ samples and, for each of them, we ran the EM algorithm 50 times from independent k-means algorithms applied to the rows and the columns of the three-way data and retained the solution corresponding to the maximum log-likelihood value. The performance of the model is evaluated through the following measures: (i) average and median Adjusted Rand Index (ARI, Hubert 1985) across simulations, which measures the agreement between the true and the estimated unit-/variable-specific partitions, corrected by chance (it is 0 for random labeling and 1 when the true and the estimated partitions completely agree); (ii) the estimated Root Mean Squared Error (RMSE) of model parameters. The latter is obtained as the root of the average squared differences between the estimated and the true parameter values across simulations (the smaller the estimated RMSE, the better the performance).

Table 2 shows the ARI values obtained with $T = 4$, $K = 3$, $Q = 2$, and $R = 1$ for varying variance levels and number of variables /units. As it can be observed, in all settings, the number of variables p has an impact on units' and, above all, variables' partitioning, as the ARI value clearly increases with p . For $p = 24$, the variance level and the sample size n do not have any impact on partitioning of both units and

Table 3 Polynomial time function with $R = 1$: RMSE values for $K = 3, Q = 2,$ and $T = 4,$ in different settings

n	RMSE ϕ	RMSE Λ	RMSEs	RMSE π
High variance				
$p = 12$				
50	[0.34, 0.31, 0.15]	[0.02, 0.02]	[0.48, 0.75, 1.00]	[0.16, 0.20, 0.27]
100	[0.31, 0.30, 0.15]	[0.01, 0.01]	[0.49, 0.75, 1.00]	[0.22, 0.18, 0.33]
200	[0.14, 0.19, 0.03]	[0.01, 0.01]	[0.49, 0.75, 1.00]	[0.39, 0.22, 0.27]
500	[0.03, 0.02, 0.02]	[0.01, 0.01]	[0.50, 0.75, 1.00]	[0.02, 0.02, 0.02]
$p = 24$				
50	[0.05, 0.05, 0.04]	[0.01, 0.01]	[0.49, 0.74, 0.99]	[0.07, 0.05, 0.08]
100	[0.03, 0.04, 0.02]	[0.01, 0.01]	[0.49, 0.75, 0.99]	[0.05, 0.03, 0.06]
200	[0.02, 0.02, 0.02]	[0.01, 0.01]	[0.49, 0.75, 1.00]	[0.03, 0.03, 0.04]
500	[0.02, 0.02, 0.01]	[0.00, 0.00]	[0.50, 0.75, 1.00]	[0.02, 0.02, 0.02]
Low variance				
$p = 12$				
50	[0.33, 0.28, 0.18]	[0.02, 0.02]	[0.26, 0.19, 0.18]	[0.26, 0.36, 0.34]
100	[0.25, 0.28, 0.16]	[0.01, 0.01]	[0.15, 0.21, 0.18]	[0.29, 0.18, 0.32]
200	[0.09, 0.02, 0.02]	[0.01, 0.01]	[0.08, 0.04, 0.03]	[0.17, 0.09, 0.12]
500	[0.02, 0.02, 0.01]	[0.01, 0.01]	[0.02, 0.03, 0.03]	[0.02, 0.02, 0.02]
$p = 24$				
50	[0.03, 0.03, 0.03]	[0.01, 0.01]	[0.04, 0.07, 0.06]	[0.07, 0.05, 0.08]
100	[0.02, 0.02, 0.04]	[0.01, 0.01]	[0.03, 0.03, 0.15]	[0.13, 0.29, 0.37]
200	[0.01, 0.01, 0.01]	[0.01, 0.01]	[0.02, 0.03, 0.02]	[0.03, 0.03, 0.04]
500	[0.01, 0.01, 0.01]	[0.00, 0.00]	[0.02, 0.02, 0.02]	[0.02, 0.02, 0.02]

Table 4 Polynomial time functions: mean and median (in brackets) ARI values for $n = 100$ and $p = 12,$ in different settings

High variance						Low variance					
K	Q	T	R	Row	Col	K	Q	T	R	Row	Col
3	2	8	1	1.00 (1.00)	1.00 (1.00)	3	2	8	1	1.00 (1.00)	1.00 (1.00)
3	2	4	2	1.00 (1.00)	1.00 (1.00)	3	2	4	2	1.00 (1.00)	1.00 (1.00)
4	3	4	1	1.00 (1.00)	0.90 (0.94)	4	3	4	1	1.00 (1.00)	0.90 (1.00)

Table 5 Polynomial time functions: RMSE values for $n = 100$ and $p = 12,$ in different settings

K	Q	T	R	RMSE ϕ	RMSE Λ	RMSEs	RMSE π
High variance							
3	2	8	1	[0.09, 0.04, 0.02]	[0.00, 0.00]	[0.49, 0.75, 0.99]	[0.21, 0.11, 0.23]
3	2	4	2	[0.10, 0.10, 0.10]	[0.09, 0.09, 0.01, 0.02]	[0.49, 0.75, 0.98]	[0.11, 0.12, 0.18]
4	3	4	1	[0.12, 0.04, 0.12, 0.04]	[0.01, 0.05, 0.01]	[0.49, 0.75, 1.00, 1.23]	[0.18, 0.20, 0.14, 0.17]
Low variance							
3	2	8	1	[0.02, 0.04, 0.01]	[0.00, 0.00]	[0.02, 0.17, 0.04]	[0.31, 0.16, 0.17]
3	2	4	2	[0.08, 0.07, 0.06]	[0.06, 0.05, 0.01, 0.01]	[0.06, 0.06, 0.05]	[0.12, 0.09, 0.16]
4	3	4	1	[0.24, 0.21, 0.14, 0.04]	[0.01, 0.05, 0.01]	[0.14, 0.18, 0.14, 0.12]	[0.19, 0.21, 0.15, 0.13]

variables, while, for $p = 12$, both quantities seem to have an impact. In any case, as expected, ARI values increase with increasing sample size and decreasing variance level. In all settings, we obtain good results in terms of recovering the unknown data structure, with an ARI index always above 0.85 (a standard threshold for evaluating clustering efficacy) and reaching the maximum achievable when dealing with a large number of variables, regardless the variance level.

In Table 3, the estimated RMSEs of model parameters obtained for $T = 4, K = 3, Q = 2$, and $R = 1$ are reported. RMSEs appear low for all model parameters and decrease with increasing p and n , especially for ϕ and π . It is also worth noticing that, when the variance level is high, the sample size n does not seem to affect the quality of estimates for s and A .

Tables 4 and 5 report the ARI and the RMSE of model parameters in other simulation settings obtained by letting T, K, Q , and R vary, and by keeping constant the sample size ($n = 100$) and the number of variables ($p = 12$). In detail, as far as the polynomial degree is concerned, we set now $R = 2$ and $A = (\lambda_{11}, \lambda_{12}, \lambda_{21}, \lambda_{22})' = (0.2, 0.8, 0.2, 0.2)'$. These tables highlight the good performance of the proposal in terms of both partitions' and parameters' recovery. ARI values are, on average and in median, all equal to the maximum possible achievable, besides the simulation scenario, except for that corresponding to $K = 4, Q = 3, T = 4$ and $R = 1$. In this case, the ARI measuring the agreement between the true partitioning of variables and the estimated one slightly reduces, even though the standard 0.85 threshold is still overtaken. Also results in Table 5 do not substantially differ from those reported in Table 3, suggesting that parameters' recovery is ensured also in simulation settings characterized by a more complex block structure.

6.2 Basis time function

To get more general insight, we simulated $B = 100$ samples by considering responses y_{ijt} from a conditional Negative Binomial distribution, with canonical parameter $\theta_{jt(k)}$ modeled according to equation (5) and over dispersion parameter $s_{j(k)} = s, j = 1, \dots, p, k = 1, \dots, K$. Based on the results described in the previous section, we focused the attention on scenarios with $n = 50, 100, 200$ sample units, $T = 4, 8$ measurement occasions and $p = 12$ variables, partitioned into $K = 3$ components and $Q = 2$ segments, respectively. We fixed the component-specific locations to $\phi = (1, 1.5, 2)'$, the overdispersion parameter vector to $s = (s_{(1)}, s_{(2)}, s_{(3)})' = (1, 1, 1)'$, the component prior probabilities to $\pi = (0.3, 0.2, 0.5)'$, and the elements of $\beta(t)$ as

$$\beta_q(t) = \begin{cases} 2 \cos[2\pi(t + 1/(T + 1))] & \text{if } q = 1, \\ \pi^{-1}[t + 1/(T + 10)] & \text{if } q = 2, \end{cases}$$

Table 6 Basis time function: RMSE values for $K = 3, Q = 2$, and $p = 12$, in different settings

n	RMSE ϕ	RMSE s	RMSE π
$T = 4$			
50	[0.00, 0.12, 0.07]	0.07	[0.10, 0.12, 0.03]
100	[0.00, 0.06, 0.05]	0.05	[0.05, 0.06, 0.02]
200	[0.00, 0.04, 0.03]	0.04	[0.03, 0.04, 0.01]
$T = 8$			
50	[0.00, 0.10, 0.07]	0.07	[0.11, 0.13, 0.03]
100	[0.00, 0.06, 0.03]	0.06	[0.07, 0.08, 0.02]
200	[0.00, 0.02, 0.02]	0.03	[0.03, 0.04, 0.02]

Table 7 Basis time function: mean and median (in brackets) ARI values for $K = 3, Q = 2$, and $p = 12$, in different settings

n	$T = 4$		$T = 8$	
	Row	Col	Row	Col
50	0.81 (0.85)	0.59 (0.60)	0.97 (1.00)	0.93 (1.00)
100	0.87 (0.87)	0.85 (0.89)	0.99 (1.00)	0.98 (1.00)
200	0.90 (0.90)	0.96 (1.00)	1.00 (1.00)	1.00 (1.00)

in order to obtain two well distinguished dynamics for the longitudinal outcomes.

As in the previous section, an initialization strategy based on 50 independent k-means algorithms was adopted and the solution corresponding to the maximum log-likelihood value was retained as the optimal one. A spline basis time function approach was considered for recovering the longitudinal dynamics of the observed outcomes. To evaluate the performance of the proposal we relied on the estimated RMSE of model parameters, as well as on the average (and median) ARI across simulations. Results of the analysis are reported in Tables 6 and 7.

By looking at these results, it is evident that the model is able to properly recover the underlying data generation process, in terms of parameters' and partitions' recovery, both with respect to components and segments. The quality of results clearly improves with larger sample sizes and when a higher number of occasions is available. This is particularly true when looking at the estimated column partition as the larger the number of time occasions, the more distinguishable the segment-specific trajectories we obtain according to the data generating process we considered. This feature is clearly evident when looking at Figs. 2 and 3 where, for each of the scenarios described above, we report the average longitudinal trajectories across units obtained in a single draw of the simulation study. Units are distinguished in terms of the component they belong to. In the same plots, we also report the estimated basis time functions obtained by averaging across simulations the estimates (solid curves) and the true functions we used for simulating the data (dashed curves). Such a figure clearly highlights very good performance in recover-

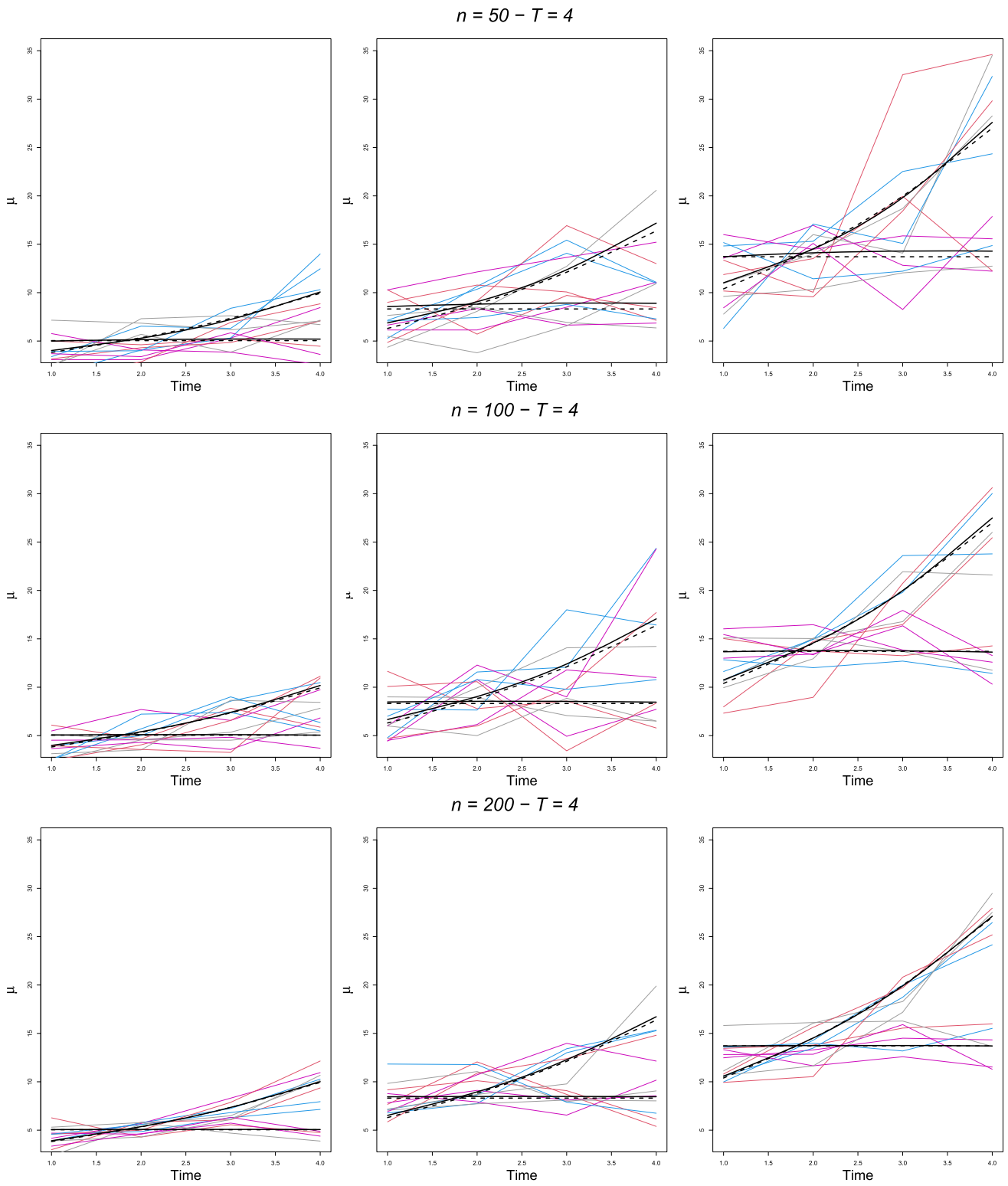


Fig. 2 Basis time function: average longitudinal trajectories across components and segments for $T = 4$, $p = 12$, $K = 3$ and $Q = 2$, in different settings. The solid and the dashed line correspond to the estimated and the true basis time function, respectively

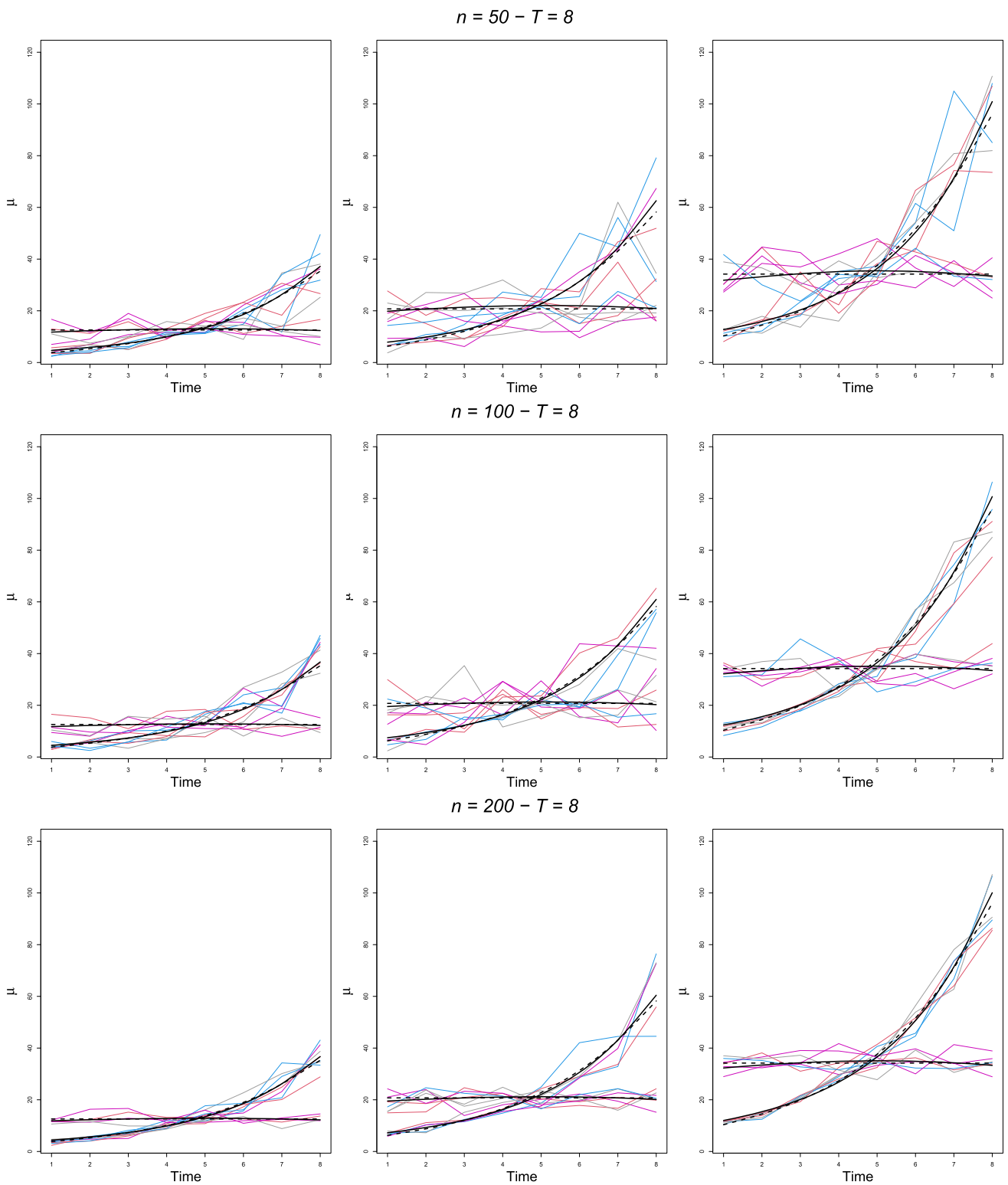


Fig. 3 Basis time function: average longitudinal trajectories across components and segments, for $T = 8$, $p = 12$, $K = 3$ and $Q = 2$, in different settings. The solid and the dashed line correspond to the estimated and the true basis time function, respectively

Table 8 Crime data: number of crime episodes per 100 thousands inhabitants in Italy, by crime type and year

Crimes	2012	2013	2014	2015	2016	2017	2018	2019	Mean	Sd
Receiving stolen goods	45.04	45.49	44.94	41.69	39.28	36.35	33.14	29.59	39.44	5.97
Beating	28.25	28.21	27.68	27.67	25.12	25.75	25.48	26.41	26.82	1.28
Cybercrimes	13.24	17.01	19.64	17.88	19.68	19.26	24.26	29.62	20.07	4.94
Damage followed by fire	20.22	17.74	15.52	16.22	15.46	17.81	13.99	13.65	16.33	2.18
Extortion	11.65	12.40	14.85	17.82	17.36	15.44	18.18	16.48	15.52	2.44
Fires	23.75	13.36	12.41	16.21	14.56	18.77	8.68	11.99	14.97	4.64
Counterfeiting of brands and products	16.05	13.44	14.18	15.32	14.06	11.53	10.80	9.14	13.06	2.37
Sexual violence	8.44	8.10	7.70	7.25	7.36	8.43	8.93	8.94	8.14	0.66
Laundering	3.00	3.38	2.87	3.24	3.14	3.37	3.39	3.33	3.21	0.20
Manslaughter	3.10	2.88	2.96	3.17	2.96	3.11	3.17	3.22	3.07	0.12
Attempted murders	2.39	2.21	2.26	2.18	1.96	2.00	1.92	1.87	2.10	0.19
Infringement of intellectual property	2.75	1.94	1.94	2.19	1.91	1.44	1.05	0.94	1.77	0.60
Exploitation, aiding and abetting of prostitution	2.32	2.08	1.97	1.77	1.70	1.37	1.14	0.96	1.66	0.47
Criminal conspiracy	1.53	1.25	1.61	1.32	1.11	1.03	0.83	0.68	1.17	0.32
Child pornography	0.68	0.88	0.98	1.11	0.89	1.01	1.12	1.45	1.02	0.23
Sexual acts with minors	1.00	0.95	0.89	0.92	0.83	0.88	0.90	0.95	0.92	0.05
Voluntary murders	0.95	0.91	0.86	0.85	0.73	0.67	0.60	0.58	0.77	0.14
Usury	0.72	0.83	0.72	0.67	0.73	0.56	0.34	0.34	0.61	0.18
Mean	10.28	9.61	9.66	9.86	9.38	9.38	8.77	8.90		
Sd	12.32	11.93	11.88	11.48	10.89	10.62	10.23	10.35		

The mean and the standard deviation across years and crime types are reported at the margins

ing the true underlying mechanism in all simulation settings we considered. As expected, the larger the sample size and, above all, the number of time occasions, the better the results.

7 Application: Italian crime data

To provide further insight into the empirical behavior of the proposed model, we considered a real-life dataset, describing the distribution over time and space of the number of crime events in Italy. Data are freely available for download from the website of the Italian National Institute for Statistics (ISTAT) at the following link: http://dati.istat.it/Index.aspx?DataSetCode=dccv_delittips.

7.1 Details

The analyzed data provide information on the number of crime episodes reported yearly by the Italian enforcement authorities (Polizia, Arma dei Carabinieri, Guardia di Finanza) to justice from 2012 to 2019 ($T = 8$). Data are avail-

able at province level (LAU1), so that information entails a total of $n = 106$ statistical units. Only operational activities of the enforcement authorities are considered, regardless the subsequent judicial process of the reported facts. A set of 18 different crime types were considered for the analysis; that is, we have information on $J = 18$ variables. A complete list of such variables is reported in Table 8, together with the corresponding number of episodes registered in Italy per 100 thousands inhabitants, over the analyzed time window. Also, the mean number of crime episodes registered yearly in the country and the mean number of episodes of a given type registered in the whole observation window are reported at the margins of the table, together with the corresponding standard deviation.

Looking at the last columns in the table, we may observe that the average number of crime episodes reported to justice every year significantly varies with the crime type. Receiving stolen goods and beating are the most frequent ones, with an average number of reported episodes equal to 39.44 and 26.82 per 100 thousands inhabitants in the country, respectively. High values are also observed for cybercrimes, damage



Fig. 4 Crime data: distribution of the total number of crime episodes per 100 thousands inhabitants, by province

followed by fire, and extortion (20.07, 16.33, and 15.52, respectively). At the bottom of the list, we find voluntary murders and usury, with an average number of episodes equal to 0.77 and 0.61 per 100 thousands inhabitants, respectively.

For a deeper understanding, we report in Fig. 4 the yearly average number of episodes for any type of crime per 100 thousands inhabitants across the Italian provinces. Such a representation highlights differences across areas, with averages that generally increase as we move from the North to the South of the country and the isles. Exceptions mainly entail some provinces located in Liguria (North-West region), where the average yearly number of episodes seems to be the highest of the country.

To further explore this latter result, we report in Fig. 5 the distribution across provinces of the average number of episodes reported to justice every year per 100 thousands inhabitants, by looking at the crime types appearing at the top (most frequent) of Table 8. This further representation lets emerge even more pronounced differences across areas. Provinces in Liguria seems to be more incline to crimes of the type *receiving stolen goods*, *beating*, and *cybercrime*. Beating episodes are rather frequent also in the Center-North of the country, with some picks in some of the provinces of Piemonte, Lombardia and Emilia Romagna, in the South (provinces in the South of Campania, Basilicata and Calabria), and in Sardinia. *Damages followed by fire* and *fire* are more likely in the South and in the isles of the peninsula, while *extortion* episodes are more spread all around the

country. Some peaks are evident in Campania, Puglia, and Calabria, as well as in the province of Bologna, Milan, and the nearby. As expected, besides the crime type, provinces located in the North-East present lower criminality levels with the respect to the rest of the country.

Going back to Table 8, as stated above, we report in the last rows, the average number of crime episodes for each year and the corresponding standard deviation. Here, we observe that averages slightly reduce over years and the same does also the variability of data around such quantities. To put emphasis on this aspect, we represent in Fig. 6 the distribution of the total number of crime episodes reported to justice per 100 thousands inhabitants across provinces, at the beginning and at the end of the observation period. Some areas may in fact have experienced a more evident change.

By looking at the figure, we observe that improvements entail most of the Italian provinces, but more evidently those located in the Center-South and the South of the country.

Last, focusing on single cells of Table 8 reporting the total number of crime episodes by year and crime type per 100 thousands inhabitant, we may observe quite different trends for the variables under investigation. Some do mainly remain constant over time (e.g., sexual violence), some others tend to increase or decrease over time (e.g., cybercrimes and receiving stolen goods, respectively), others have higher variability (e.g., fires), with subsequent up and down. This aspect will be further investigated in the subsequent section.

7.2 Analysis

As it is evident from above, data do present a complex structure and deriving a clear interpretation is not trivial. For this purpose, we used the proposed model-based biclustering approach to obtain a dimensionality reduction of both rows (provinces) and columns (crime types) of the three-way data structure. In detail, our aim is that of identifying clusters of units (the Italian provinces) characterized by similar dynamics in the number of crime episodes per 100 thousand inhabitants, for subsets of variables (the crime types).

We considered a Negative Binomial distribution (conditional on provinces' component membership) for the count of crime episodes of type $j = 1, \dots, J$, reported to justice in province $i = 1, \dots, n$, at occasion $t = 2012, \dots, 2019$. That is, $Y_{ijt} \mid Z_{ik} = 1 \sim \text{NegBin}(\theta_{ijt(k)}, s_{(k)})$, where $s_{(k)}$ is the overdispersion parameter and $\theta_{ijt(k)}$ is the canonical parameter described by the following model

$$\theta_{ijt(k)} = \phi_k + \mathbf{a}'_{jk}[\mathbf{A}\boldsymbol{\omega}(t) + \boldsymbol{\Xi}\mathbf{u}(t)] + \log(n_{it}).$$

We decided to focus on a spline basis representation of the linear predictor in order to give high flexibility to the model, by setting the corresponding degree to $R = 2$. This latter choice was motivated by the low number of yearly mea-

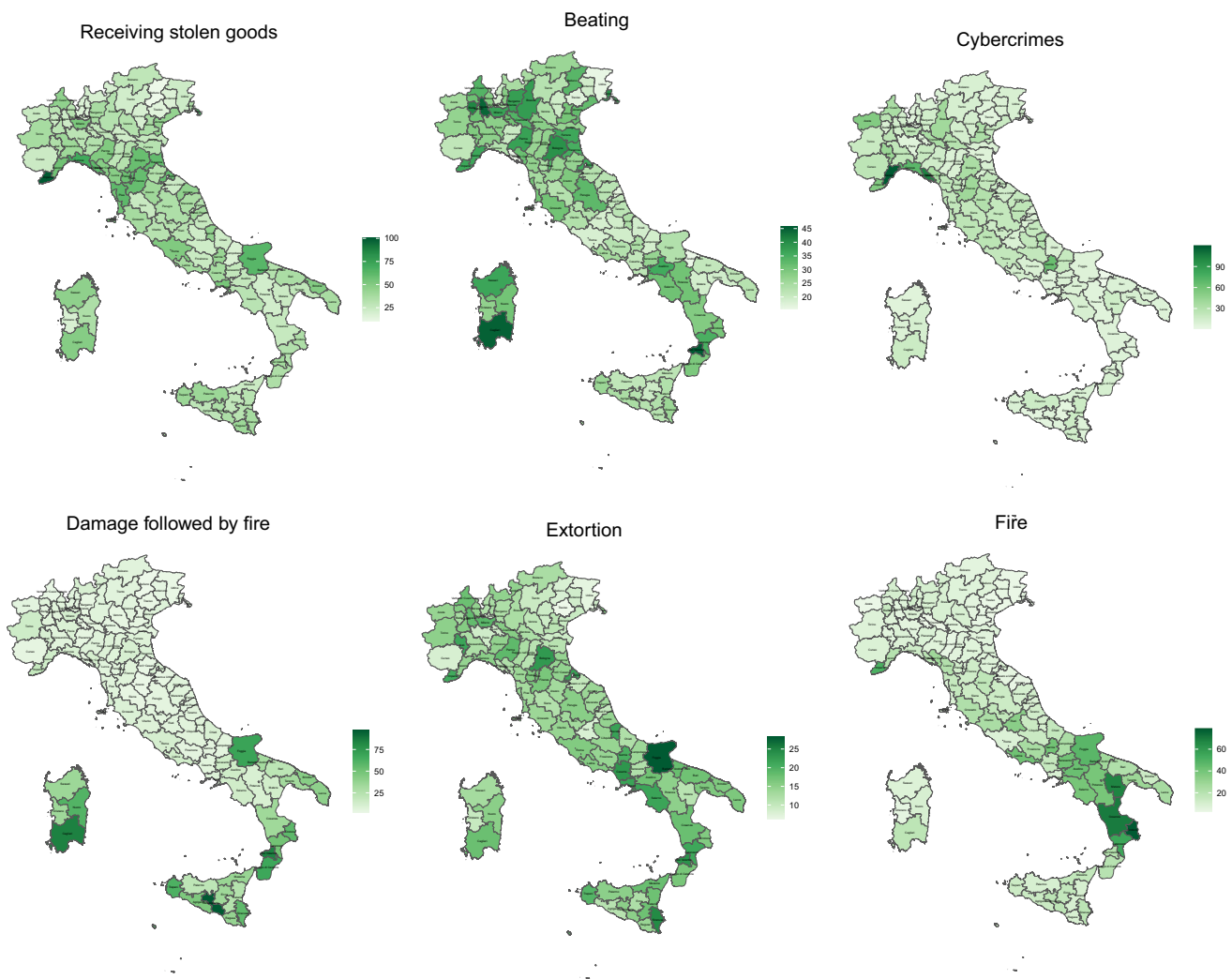


Fig. 5 Crime data: distribution of the total number of crime episodes per 100 thousands inhabitants, by province and crime type

surement occasions ($T = 8$) available for each province. Further, in the linear predictor above, n_{it} denotes the population size for province i at occasion t , playing the role of an offset that captures the differential weight of provinces. Considering such an offset term may help us finding groups of provinces having similar trend of crime rates during the analyzed period. For simplicity, we considered a common dispersion parameter across variables and components; that is $s_{j(k)} = s, j = 1, \dots, p, k = 1, \dots, K$.

We ran the EM algorithm described in Sect. 4 for a varying number of components ($K = 2, \dots, 5$) and segments ($Q = 2, \dots, 8$). An initialization strategy based on 50 random start was considered, as in the simulation study. For each combination $[K, Q]$, the model corresponding to the maximum value of the log-likelihood was retained as the optimal one. We report in Fig. 7 the value of the BIC index associated to each of such optimal solutions, for varying K and Q .

Such a figure suggests the model corresponding to $K = 3$ and $Q = 8$ (BIC = 109, 313.3) to be the optimal solution. That is, the optimal model is the one lying on the boundary of the set of values we considered for Q . Looking at the results, however, we noticed that this represents a spurious solution due to the presence of almost empty segments; for this reason, we decided to focus on other choices of K and Q . By looking at Fig. 7, it is evident that the model corresponding to $K = 4$ may represent an appropriate alternative. As regards the choice of Q , we identify the optimal model as the one based on $Q = 6$ (BIC = 109, 935.5). This solution guaranties interpretability of results and parsimony.

We report in Table 9 the component-specific parameter estimates, that is, the model intercepts ϕ_k and the component prior probabilities π_k , for $k = 1, \dots, K$, as well as the size of each component obtained by allocating provinces to clusters according to a MAP rule.

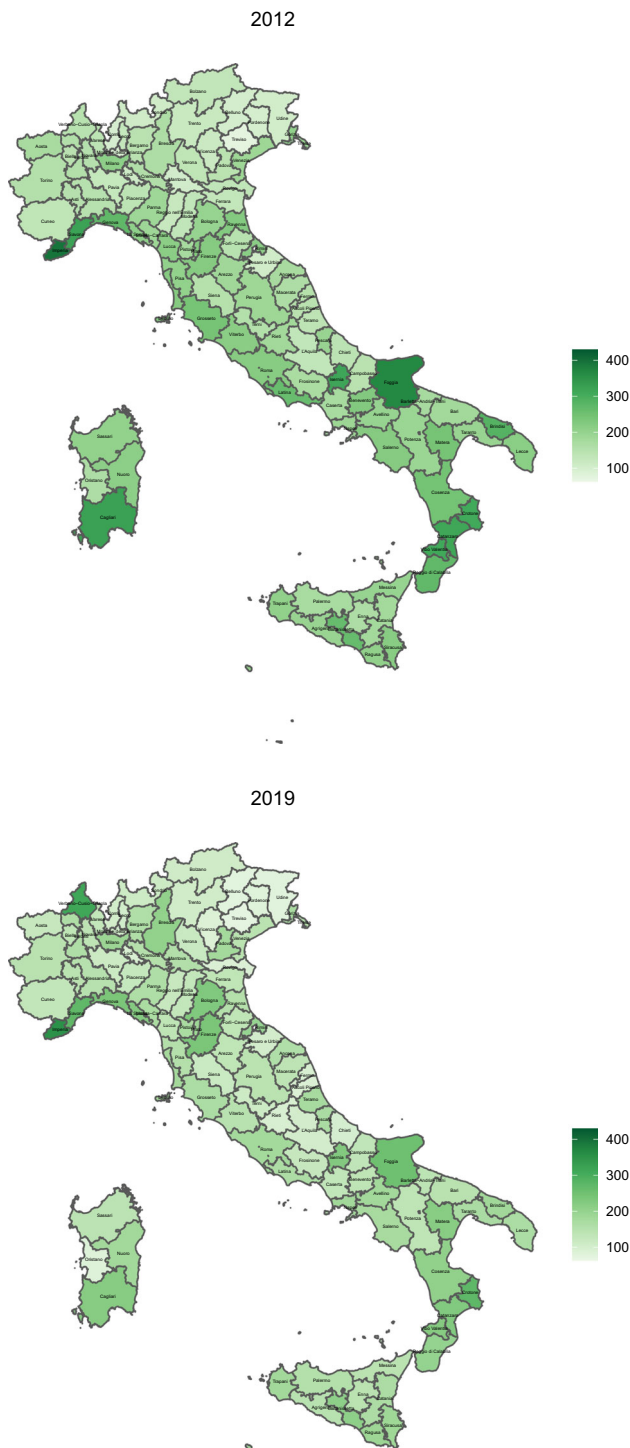


Fig. 6 Crime data: distribution of the total number of crime episodes per 100 thousands inhabitants, by province and year

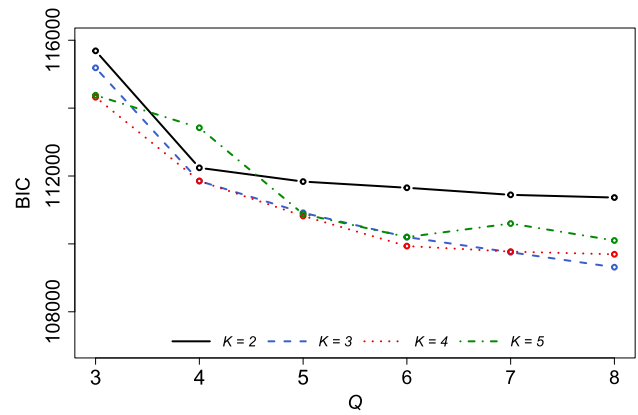


Fig. 7 Crime data: model selection. BIC index for varying K and Q

Table 9 Crime data: row-cluster parameter estimates and size of the clusters according to a MAP rule

k	1	2	3	4
ϕ_k	-9.695	-9.306	-9.206	-8.965
π_k	0.507	0.132	0.164	0.198
size	55	13	17	21

The four clusters are well separated and are characterized by a negative intercept, whose magnitude decreases when moving from cluster 1 to 4. That is, the model identifies provinces characterized by increasing baseline “criminality levels”. Provinces belonging to the former cluster do present a baseline average number of crime episodes reported to justice by the authoritative forces equal to $e^{-9.695} = 6.159e^{-05}$; that is, 6.159 episodes per 100 thousands inhabitants and such a value increases to 9.088, 10.044, and 12.781 when considering provinces allocated to the second, the third, and the fourth component, respectively.

We report in Fig. 8 the allocation of provinces to components, where areas are colored according to the cluster they are assigned to. By looking at the graphical representation and pairing it with the results discussed in Sect. 7.1, we may observe that the model allows to identify groups characterized by a homogeneous baseline propensity to the event of interest. The 4-th component groups those provinces characterized by higher criminality levels, mainly located in the South of Calabria, Sicily, and Apulia (all provinces in the South of the Italian peninsula), together with most of the provinces in Sardinia. Surprisingly the province of Turin is allocated to this cluster too. The 3-rd component clusters provinces mainly located in the Center of the country and includes, among other, all provinces in Liguria, some from Tuscany and Emilia-Romagna, as well as the province of Venice. The 2-nd component includes provinces located in the Center-South: all provinces in Basilicata, those in Apulia not belonging to the 4-th component, the province of Cosenza

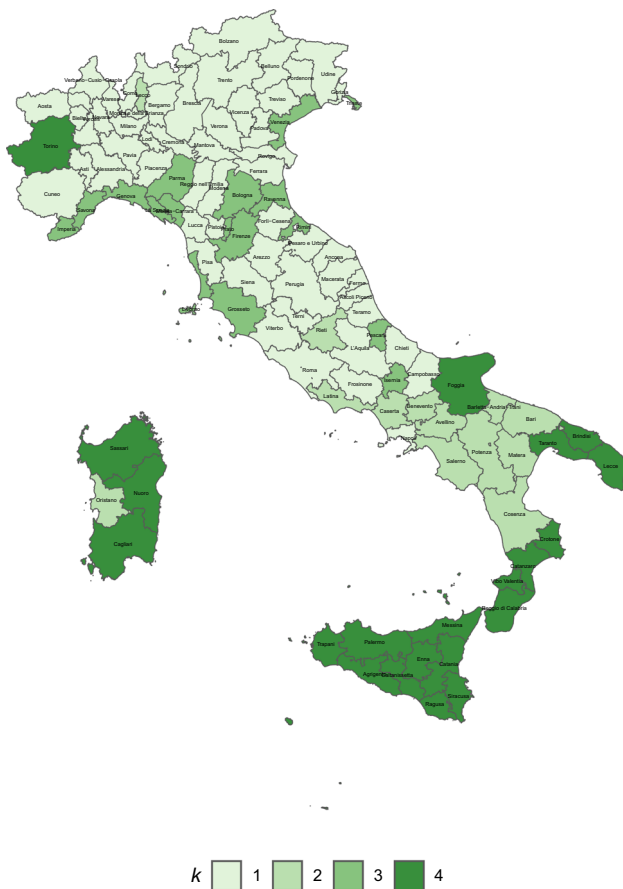


Fig. 8 Crime data: allocation of Italian provinces to clusters, according to a MAP rule

(Calabria), and all provinces in Campania, but for Naples. Finally, all the remaining Italian provinces are allocated to the 1-st component. This includes, among others, the largest and the most dense provinces of the country: Rome, Milan, and Naples.

As far as variable partition is concerned, we show in Fig. 9 the estimated time function $\beta_q(t)$, derived by substituting parameter estimates for Λ and Ξ into equation (8), $q = 1, \dots, Q$, and $t = 1, \dots, T$. As it is evident, each segment identifies variables evolving over time in a similar manner. Almost all of them show a negative trend: segment $q = 1$ collects variables reducing faster at the beginning of the observation period and more gradually at end; segment $q = 2$ and, more evidently, $q = 4$ are characterized by the opposite trend, with a reduction in the number of reported crime episodes that becomes steeper in the last years of observation; segments $q = 5$ and, above all, $q = 6$ do identify crime types for which the reported number of episodes reduces almost linearly over time. On the contrary, segment $q = 3$ is characterized by a reduction at first (till year 2015) and then an increase in the number of crime episodes reported to justice by administrative authorities.

For a deeper understanding, we report in Table 10 the composition of segments across components. By looking at these results, we observe that most of the variables (corresponding to the different crime types) are classified in a different segment across the row-clusters. This means that the reported number of crime episodes evolves differently in the analyzed time window across the Italian provinces. The only exceptions are represented by *attempted murders* (always belonging to segment $q = 5$), *receiving stolen goods* (always in segment $q = 6$), and *usury* and *voluntary murders* (always classified in segment $q = 2$).

In detail, focusing on provinces belonging to the 1-th component (the one presenting lower baseline criminality levels), *counterfeiting of brands and products*, *damage followed by fire*, and *sexual violence* are classified in segment $q = 1$. That is, the number of crime episodes of this type reduces at first and then tend to stabilize. *Infringement of intellectual property*, *usury*, and *voluntary murders* are classified in segment $q = 2$, while *extortions* and *fire* to segment $q = 4$. That is, the reported number of crime episodes of this type reduces mildly first and more evidently at the end of the observation period. *Attempted murders*, *child pornography*, *criminal conspiracy*, and crimes related to *prostitution* and *sexual acts with minors* belong to segment $q = 5$; that is, their levels gently reduce over time linearly. Such a linear (negative) trend is more pronounced for *beating*, *cybercrimes*, and *receiving stolen goods*. Segment $q = 3$ includes *laundering* and *manslaughter*. As stated above, these variables reduce at first and start increase again since 2016 on.

On the contrary, if we focus on provinces belonging to the 4-th component (with higher baseline criminality levels), we do observe that those variables that reduce rapidly at the beginning till reaching a plateau in the last part of the observation period (segment $q = 1$) correspond to *counterfeiting of brands and products*, *cybercrimes*, and *extortion*. *Child pornography*, crimes entailing *prostitution* and *sexual acts with minors*, *usury*, and *voluntary murders* belong to segment $q = 2$, while *beating* and *fires* to segment $q = 4$. All these variables do reduce more evidently as time passes by. *Sexual violence* is the only variable classified in segment $q = 3$ (that showing a parabolic trend), while the reported number of episodes reduces linearly over time for all the remaining crime types, classified in segment $q = 5$ and $q = 6$.

To conclude, we report in Fig. 10 the longitudinal trajectories of the number of reported crime episodes per 100 thousands inhabitants by type, while accounting for both the provinces, and the crime types's partitions. The dotted blue line corresponds to the estimated average number of episodes per 100 thousands inhabitants obtained from equation (9). These lines, reflects the temporal evolutions described above, even though the effects are mitigated by both the scale in the y-axis of the plots and the intercept in the model. By looking at the figure, similarities between provinces and crimes



Fig. 9 Crime data: the estimated time function $\beta_q(t)$

Table 10 Crime data: variable partition across components

Crimes	1	2	3	4
Attempted murders	5	5	5	5
Beating	6	6	6	4
Child pornography	5	2	5	2
Counterfeiting of brands and products	1	1	4	1
Criminal conspiracy	5	2	2	5
Cybercrimes	6	1	6	1
Damage followed by fire	1	4	1	6
Exploitation, aiding and abetting of prostitution	5	5	5	2
Extortion	4	4	4	1
Fires	4	6	4	4
Infringement of intellectual property	2	5	2	5
Laundering	3	5	3	5
Manslaughter	3	3	3	5
Receiving stolen goods	6	6	6	6
Sexual acts with minors	5	2	2	2
Sexual violence	1	3	1	3
Usury	2	2	2	2
Voluntary murders	2	2	2	2

belonging to the same bi-cluster are evident, thus highlighting the effectiveness of the proposed method.

8 Conclusion

In this paper, we propose a model-based biclustering approach for multivariate longitudinal data. These represent a specific type of three-way data where rows identify units, columns identify variables, and layers identify time occasions. The number of time occasions in longitudinal studies is usually low and this makes the application of methods defined for biclustering functional data rather inappropriate, as outlined in the text. We use a finite mixture of generalized linear models to partition units into a given number of components. Within each of them, we use a simple parameterization of the canonical parameter to obtain a partition of variables into a given number of segments, which are characterized by a similar evolution over time, described by a flexible and parsimonious representation. An EM-type algorithm is used to derive model parameter estimates. This has been developed in R language by the authors and is made available upon request. A large scale simulation study highlights the performance of the proposed approach, both in terms of model parameters' recovery and biclustering ability. The features of our proposal are also illustrated via the application to a real dataset entailing crimes reported to justice by the Italian enforcement authorities in 2012–2019. Results of this analysis let us identifying geographical areas in the country sharing common longitudinal trajectories for specific subsets of crime-types, where the subsets vary when we look at different clusters of provinces.

An interesting evolution of the proposed approach entails the treatment of missing data. It is indeed rather common in the longitudinal framework to deal with missing observations. Even though the proposal can be straightforwardly extended to the case of unbalanced longitudinal data in the

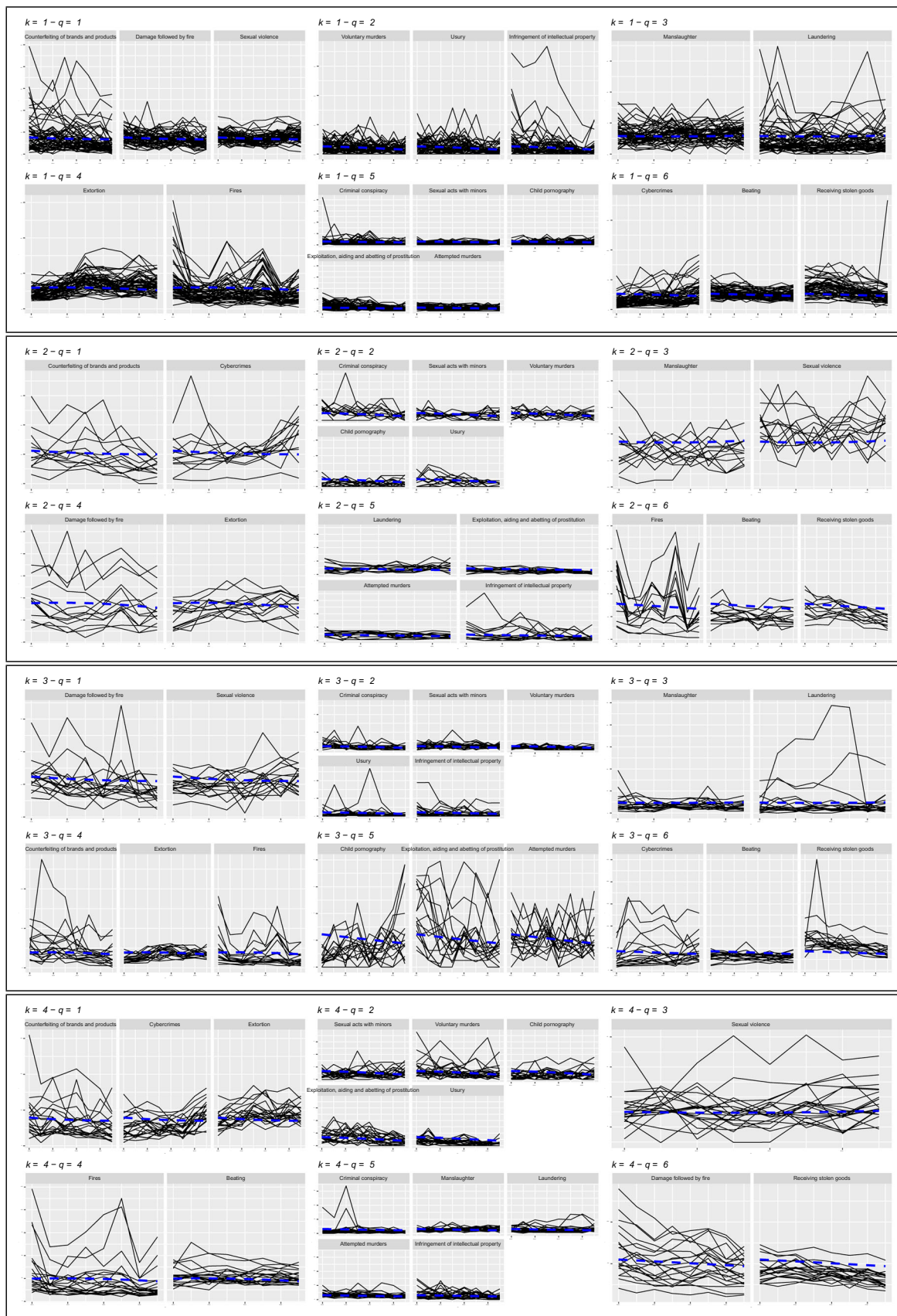


Fig. 10 Crime data:longitudinal trajectories of variables across clusters and segments

case of non-informative missingness by basing inference and partitioning on the observed data only, (at the cost of a slightly more complex implementation of the EM algorithm), the informative case is more demanding. This would require modeling the missing data process together with the longitudinal one, at the cost of a more complex model specification which, however, would allow us deriving unbiased parameter estimates, as well as more reliable biclusters. A further issue that may arise due to missing data is that of non common time occasions; when a few time occasions are available, this may make retrieving the longitudinal trajectory more complex. In this case, multiple imputation techniques may be a useful tool of analysis.

Funding Open access funding provided by Università degli Studi di Firenze within the CRUI-CARE Agreement.

Declarations

Conflict of interest All authors declare that they have no conflicts of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Akaike, H.: Information Theory and an Extension of the Maximum Likelihood Principle. Springer, New York, 199–213 (1973)
- Arnold, R., Hayakawa, Y., Yip, P.: Capture-recapture estimation using finite mixtures of arbitrary dimension. *Biometrics* **66**, 644–655 (2010)
- Atienza, N., Garcia-Heras, J., Munoz-Pichardo, J.: A new condition for identifiability of finite mixture distributions. *Metrika* **63**, 215–221 (2006)
- Basford, K.E., McLachlan, G.J.: The mixture method of clustering applied to three-way data. *J. Classif.* **2**, 109–125 (1985)
- Baudry, J.-P., Raftery, A.E., Celeux, G., Lo, K., Gottardo, R.: Combining mixture components for clustering. *J. Comput. Graph. Stat.* **19**, 332–353 (2010)
- Biernacki, C., Celeux, G., Govaert, G.: Assessing a mixture model for clustering with the integrated completed likelihood. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**, 719–725 (2000)
- Bock, H.: Simultaneous clustering of objects and variables. In: Tomasone, R. (ed.) *Anal. des donnees es et Inform.*, 187–204. INRIA, Le Chesnay, France (1979)
- Bouveyron, C., Bozzi, L., Jacques, J., Jollois, F.-X.: The functional latent block model for the co-clustering of electricity consumption curves. *J. R. Stat. Soc.: Ser. C (Appl. Stat.)* **67**(4), 897–915 (2018)
- Bouveyron, C., Jacques, J., Schmutz, A.: funLBM: Model-Based Co-Clustering of Functional Data. R package version 2, 3 (2022)
- Brault, V., Lomet, A.: Methods for co-clustering: a review. *J. de la Société Française de Stat.* **156**, 27–51 (2015)
- Bruckers, L., Molenberghs, G., Drinkenburg, P., Geys, H.: A clustering algorithm for multivariate longitudinal data. *J. Biopharm. Stat.* **26**(4), 725–741 (2016)
- Celeux, G., Soromenho, G.: An entropy criterion for assessing the number of clusters in a mixture model. *J. Classif.* **13**, 195–212 (1996)
- Cheng, Y., Church, G.M.: Biclustering of expression data. In *Ismb* **8**, 93–103 (2000)
- Coffey, N., Hinde, J., Holian, E.: Clustering longitudinal profiles using p-splines and mixed effects models applied to time-course gene expression data. *Comput. Stat. Data Anal.* **71**(C), 14–29 (2014)
- Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the em algorithm. *J. R. Stat. Soc.: Ser. B (Methodology)* **39**, 1–22 (1977)
- Everitt, B.S., Landau, S., Leese, M., Stahl, D.: *Cluster Analysis*. Wiley, London (2011)
- Fernández, D., Arnold, R., Pledger, S., Liu, I., Costilla, R.: Finite mixture biclustering of discrete type multivariate data. *Adv. Data Anal. Classif.* **13**, 117–143 (2019)
- Galvani, M., Torti, A., Menafoglio, A., Vantini, S.: FunCC: a new bi-clustering algorithm for functional data with misalignment. *Comput. Stat. Data Anal.* **160**, 107219 (2021)
- Ghahramani, Z., Hinton, G.E., et al.: The em algorithm for mixtures of factor analyzers. Technical report, Citeseer (1996)
- Giordani, P., Ferraro, M.B., Martella, F.: *An Introduction to Clustering with R*. Springer, Berlin (2020)
- Good, I.: *Categorization of Classification*. Mathematics and Computer Science in Biology and Medicine. Her Majesty's Stationary Office, London (1965)
- Gordon, A., Vichi, M.: Partitions of partitions. *J. Classif.* **15**, 265–285 (1998)
- Govaert, G., Nadif, M.: Clustering with block mixture models. *Pattern Recognit.* **36**, 463–473 (2003)
- Govaert, G., Nadif, M.: Block clustering with Bernoulli mixture models: comparison of different approaches. *Comput. Stat. Data Anal.* **52**, 3233–3245 (2008)
- Govaert, G., Nadif, M.: Latent block model for contingency table. *Commun. Stat. -Theory Methods* **39**, 416–425 (2010)
- Govaert, G., Nadif, M.: *Co-Clustering: Models, Algorithms and Applications*. Wiley, London (2013)
- Green, P.J., Silverman, B.W.: *Nonparametric Regression and Generalized Linear Models: A Roughness Penalty Approach*. CRC Press (1993)
- Hartigan, J.A.: Direct clustering of a data matrix. *J. Am. Stat. Assoc.* **67**, 123–129 (1972)
- Hartigan, J.A.: *Clustering Algorithms*. Wiley, London (1975)
- Hastie, T., Tibshirani, R.: *Generalized Additive Models*. Wiley Online Library (1990)
- Hastie, T., Tibshirani, R., Friedman, J.H., Friedman, J.H.: *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, vol. 2. Springer, Berlin (2009)
- Hennig, C.: Identifiability of models for clusterwise linear regression. *J. Classif.* **17** (2000)
- Hennig, C., Meila, M., Murtagh, F., Rocci, R.: *Handbook of Cluster Analysis*. CRC Press (2015)
- Hubert, L., Arabie, P.: Comparing partitions. *J. Classif.* **2**, 193–218 (1985)
- Hunt, L.A., Basford, K.E.: Fitting a mixture model to three-mode three-way data with categorical and continuous variables. *J. Classif.* **16**, 283–296 (1999)
- Kass, R.E., Raftery, A.E.: Bayes factors. *J. Am. Stat. Assoc.* **90**, 773–795 (1995)

- Kaufman, L., Rousseeuw, P.J.: Finding Groups in Data: An Introduction to Cluster Analysis. Wiley, London (2009)
- Lazzeroni, L., Owen, A.: Plaid models for gene expression data. *Stat. Sinica*, 61–86 (2002)
- Lee, S., Huang, J.Z.: A biclustering algorithm for binary matrices based on penalized Bernoulli likelihood. *Stat. Comput.* **24**, 429–441 (2014)
- Li, J., Zha, H.: Two-way poisson mixture models for simultaneous document classification and word clustering. *Comput. Stat. Data Anal.* **50**, 163–180 (2006)
- Madeira, S.C., Oliveira, A.L.: Biclustering algorithms for biological data analysis: a survey. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **1**, 24–45 (2004)
- Mankad, S., Michailidis, G.: Biclustering three-dimensional data arrays with plaid models. *J. Comput. Graph. Stat.* **23**, 943–965 (2014)
- Martella, F., Alfò, M.: A finite mixture approach to joint clustering of individuals and multivariate discrete outcomes. *J. Stat. Comput. Simul.* **87**, 2186–2206 (2017)
- McLachlan, G., Peel, D.: Finite Mixture Models. Wiley, London (2000)
- Mechelen, I. V., Schepers, J.: A unifying model for biclustering. In: *Compstat 2006-Proceedings in Computational Statistics*, 81–88. Springer (2006)
- Pledger, S., Arnold, R.: Multivariate methods using mixtures: correspondence analysis, scaling and pattern-detection. *Comput. Stat. Data Anal.* **71**, 241–261 (2014)
- Priam, R., Nadif, M., Govaert, G.: The block generative topographic mapping. In: *IAPR Workshop on Artificial Neural Networks in Pattern Recognition*, 13–23. Springer (2008)
- Priam, R., Nadif, M., Govaert, G.: Topographic Bernoulli block mixture mapping for binary tables. *Pattern Anal. Appl.* **17**, 839–847 (2014)
- Rubin, D.B.: Inference and missing data. *Biometrika* **63**, 581–592 (1976)
- Ruppert, D.: Selecting the number of knots for penalized splines. *J. Comput. Graph. Stat.* **11**, 735–757 (2002)
- Schwarz, G.: Estimating the dimension of a model. *Ann. Stat.*, 461–464 (1978)
- Slimen, Y.B., Allio, S., Jacques, J.: Model-based co-clustering for functional data. *Neurocomputing* **291**, 97–108 (2018)
- Soromenho, G.: Comparing approaches for testing the number of components in a finite mixture model. *Comput. Stat.* **9**, 65–78 (1994)
- Tanay, A., Sharan, R., Shamir, R.: Biclustering algorithms: a survey. *Handb. Comput. Mol. Biol.* **9**, 122–124 (2005)
- Teicher, H.: Identifiability of mixtures. *Ann. Math. Stat.* **32**, 244–248 (1961)
- Teicher, H.: Identifiability of finite mixtures. *Ann. Math. Stat.*, 1265–1269 (1963)
- Torti, A., Galvani, M., Menafoglio, A., Vantini, S.: FunCC: Functional Cheng and Church Bi-Clustering. R package version 1.0 (2020)
- Turner, H.L., Bailey, T.C., Krzanowski, W.J., Hemingway, C.A.: Biclustering models for structured microarray data. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2**, 316–329 (2005)
- Vermunt, J.K.: A hierarchical mixture model for clustering three-way data sets. *Comput. Stat. Data Anal.* **51**, 5368–5376 (2007)
- Vicari, D., Alfó, M.: Model based clustering of customer choice data. *Comput. Stat. Data Anal.* **71**, 3–13 (2014)
- Vichi, M.: One-mode classification of a three-way data matrix. *J. Classif.* **16**, 27–44 (1999)
- Vichi, M., Rocci, R., Kiers, H.A.: Simultaneous component and clustering models for three-way data: within and between approaches. *J. Classif.* **24**, 71–98 (2007)
- Viroli, C.: Finite mixtures of matrix normal distributions for classifying three-way data. *Stat. Comput.* **21**, 511–522 (2011)
- Viroli, C.: Model based clustering for three-way data structures. *Bayesian Anal.* **6**, 573–602 (2011)
- Wierzchoń, S.T., Kłopotek, M.A.: *Modern Algorithms of Cluster Analysis*, vol. 34. Springer, Berlin (2018)
- Wood, S.N.: *Generalized Additive Models: An Introduction with R*. CRC Press (2017)
- Wyse, J., Friel, N.: Block clustering with collapsed latent block models. *Stat. Comput.* **22**, 415–428 (2012)
- Yakowitz, S.J., Spragins, J.D.: On the identifiability of finite mixtures. *Ann. Math. Stat.* **39**, 209–214 (1968)
- Zhao, X., Marron, J., Wells, M.T.: The functional data analysis view of longitudinal data. *Stat. Sin.*, 789–808 (2004)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.