Clustering Hierarchies via a Semi-Parametric Generalized Linear Mixed Model: a statistical significance-based approach

Alessandra Ragni[1,*], Chiara Masci[1], Francesca Ieva[1,2] and Anna Maria Paganoni[1]

[1]MOX Lab, Department of Mathematics, Politecnico di Milano, Milan, 20133, Italy

[2]Human Technopole, Health Data Science Center, Milan, 20157, Italy

[*]alessandra.ragni@polimi.it

Abstract

We introduce a novel statistical significance-based approach for clustering hierarchical data using semi-parametric linear mixed-effects models designed for responses with laws in the exponential family (e.g., Poisson and Bernoulli). Within the family of semi-parametric mixed-effects models, a latent clustering structure of the highest-level units can be identified by assuming the random effects to follow a discrete distribution with an unknown number of support points. We achieve this by computing α-level confidence regions of the estimated support point and identifying statistically different clusters. At each iteration of a tailored Expectation Maximization algorithm, the two closest estimated support points for which the confidence regions overlap collapse. Unlike the related state-of-the-art methods that rely on arbitrary thresholds to determine the merging of close discrete masses, the proposed approach relies on conventional statistical confidence levels, thereby avoiding the use of discretionary tuning parameters. To demonstrate the effectiveness of our approach, we apply it to data from the Programme for International Student Assessment (PISA - OECD) to cluster countries based on the rate of innumeracy levels in schools. Additionally, a simulation study and comparison with classical parametric and state-of-the-art models are provided and discussed.

Keywords: Mixed-Effects models, Nonparametric methods, Discrete random effects, EM algorithm, Generalized linear mixed models

1 Introduction

Databases featuring a hierarchical structure contain observations nested within higher-level groups in a tree-like fashion, resulting in interdependence between observations. This type of data, known as multilevel or hierarchical data, is frequently found in repeated measurements and longitudinal studies with grouping factors. Examples include student data in schools or patient data in healthcare centers, which may contain multiple levels of hierarchy, such as students within classrooms within schools. The hierarchical nature of data requires the use of specialized models like mixed effect models (Pinheiro and Bates (2000)), which account for both random and fixed effects, modeling variability at both group and individual levels.

Classic mixed-effects models assume the random effects to follow a gaussian distribution, but, over the past few years, novel semi-parametric mixed-effects models in which the random effects are assumed to follow a discrete distribution have been proposed in literature for continuous (Masci et al. (2019, 2021)) and multinomial responses (Masci et al. (2022)). The theoretical foundations of this modeling are based on the works proposed in Bock and Aitkin (1981); Lindsay (1983b,a), while the parameters estimation procedure is based on iterative Expectation-Maximization (EM) algorithms are inspired by Aitkin (1999) and Azzimonti et al. (2013). The advantage of this approach relies on the fact that, under the discrete distribution assumption, the random effects collapse within an *a priori* unknown number of support points, identifying a latent clustering structure of the *hierarchy* (groups), e.g., schools or hospitals, where the statistical units are students or patients, respectively. Through such an approach, each random effect would represent a cluster of groups, instead of a group itself, leading to several advantages. Clustering the groups is a valuable dimensionality reduction tool, especially when the cardinality of the groups is huge. For example, external institutions may want to apply a *limited* number of targeted intervention policies for the performance improvements of given providers, such as schools or hospitals. By fitting a parametric mixed effects model for observations nested within providers (i.e., the groups), the random effects will provide a ranked list of the providers, visualized in a caterpillar plot, that, although widely used in performance monitoring, is associated with some important conceptual issues (Mohammed and Deeks, 2008).

A semi-parametric mixed effects model, on the other hand, will generate a ranking of clusters of the providers, gaining in interpretability. The identification of clusters and their sizes is also a useful tool for outliers detection, where outliers are intended as very small clusters with respect to others. Moreover, this approach offers greater flexibility than the parametric version as it does not require the assumption of normal distribution of random effects. On the other side, the main drawback of this approach regards the collapsing criterion. As the number of iterations of a tailored developed Expectation Maximization algorithm grows, the support masses of the discrete distribution are made collapsed and a

new optimal discrete distribution is identified. The final number of identified clusters is not selected directly but depends on a threshold that determines the merging of discrete masses with smaller Euclidean distances along the iterations of the algorithm. However, choosing the threshold is a drawback when this method is applied to real-world data, especially without prior knowledge of the number of clusters to be identified or the difference to be observed across clusters. Tuning the threshold requires multiple runs of the algorithm, making it computationally expensive.

Within this framework, we propose an innovative method to perform the clustering of groups standing on the conventional statistical significance levels, avoiding the use of a discretional tuning parameter for cluster distance. Our proposed approach involves the computation of confidence regions centered on the two closest support points estimated using Maximum Likelihood Estimators (MLEs) and their asymptotic properties. At each iteration of the algorithm, the confidence regions are constructed after maximizing the likelihood. If the regions overlap, the two discrete masses are merged into one. The advantage of this criterion is that it identifies the latent structure solely through a statistical significance-based approach by selecting a level of confidence α, rather than an arbitrary and subjective threshold. Moreover, this approach leads to even more interpretable clusters, as their differences are statistically significant.

More specifically, we address a Semi-Parametric Generalized Linear Mixed-effects Model (SPGLMM) for responses with law in the exponential family. We recall that Generalized Linear Mixed Models (GLMMs) (Breslow and Clayton (1993)) extend upon Generalized Linear Models (GLMs) (Nelder and Wedderburn (1972), McCullagh and Nelder (1983)) by incorporating random effects into the linear predictor in addition to the fixed effects. Pointedly, we utilize the statistical significance-based approach for Poisson and Bernoulli responses, but the model is readily adaptable to other responses in the exponential family.

To show an example of the proposed model utility, we provide an application with data extracted from the Programme for International Student Assessment (PISA, OECD (2019)) to cluster countries standing on their innumeracy levels, i.e., the levels of mathematical illiteracy, as coined in Evered (1990). The OECD's PISA measures 15-year-olds' knowledge and skills in reading, mathematics, and science to handle real-life challenges. Our focus is on mathematical performance, which evaluates students' ability to apply math in various contexts. The global indicators for the United Nations Sustainable Development Goals identify a minimum Level of Proficiency - computed on the obtained scores - that all children should acquire by the end of secondary education: students below this level are considered *low-achieving students*. We aim at investigating the effect the countries involved in the OECD's PISA 2018 survey have on the rate of low-achieving students in mathematics. To do so, we fit a GLMM with non-parametric random effects which provides a random effect for each cluster of countries and auto-tunes the number of clusters according to a

chosen level of confidence.

A simulation study is proposed to assess the performance of the SPGLMM in comparison to other existing methods. To the best of our knowledge, there are no models in literature designed to perform clustering of the hierarchies in mixed models with generalized responses. To evaluate the validity, solidity and benefits of the SPGLMM based on statistical significance, we compare it with an SPGLMM that uses a discretionary threshold and with two parametric GLMMs.

The novelty of the paper is twofold: the development of semi-parametric mixed-effects models for generalized responses and, most importantly, the definition of a new methodological approach that allows identifying the clustering structure at the grouping level building the procedure on the solely statistical significance. This second point is the key point of the proposal and can be employed for any type of response variable.

The paper is organized as follows: in Section 2 we address the SPGLMM for a generalized response and we present the tailored EM algorithm for the estimation of the parameters based on a statistical-significance approach; in Section 3 we apply the SPGLMM algorithm to OECD's PISA survey data for clustering the countries standing on their school's innumeracy rates (i.e. assuming a Poisson distributed response); in Section 4 we present the simulation study in which we test the SPGLMM performances within different settings and compare them with the ones obtained by other *state-of-the-art* methods; in Section 5 we draw our conclusions and discuss some future perspectives. Further details concerning the methodology, results, proofs and a parallel discussion on a Bernoulli distributed response, can be found in the Supplementary Materials. Models implementation and results analysis are performed both through the statistical software R (R Core Team (2022)) and Python 3 (Van Rossum and Drake (2009)). The code is available upon request.

2 Methodology

In this section, we will cover the basics of a GLMM and its extension to non-parametric random effects (Section 2.1), describe the EM algorithm for parameters estimation (Section 2.2) and present the key method for reducing random effects support (Section 2.3).

2.1 GLMMs with nonparametric random effects

Our methodology focuses on the case of hierarchical data with nested observations and a single level of grouping, with N groups indexed by $i = 1, ..., N$, each containing n_i observations indexed by $j = 1, ..., n_i$, with $\sum_{i=1}^{N} n_i = J$. The vector of responses within the i^{th} group, $\mathbf{y}_i$, contains (conditionally) independent observations y_{ij} for $j = 1, ..., n_i$. The conditional distribution of $\mathbf{y}_i$ given the random effects $\mathbf{b}_i$ belongs to the exponential family with probability mass (or density) function $p(\mathbf{y}_i|\boldsymbol{\beta}, \mathbf{b}_i)$, being $\boldsymbol{\beta}$ the vector of fixed

coefficients. GLMMs are defined such that the expectation of $\mathbf{y}_i$ conditioned on $\mathbf{b}_i$ in the i^{th} group is related to the linear predictor $\mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\mathbf{b}_i$ via the monotonic and differentiable *link function* $g(\cdot)$:

$$g(\,\mathbb{E}[\mathbf{y}_i|\mathbf{b}_i]\,) = g(\boldsymbol{\mu}_i) = \boldsymbol{\eta}_i = \mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\mathbf{b}_i \quad \text{for} \quad i = 1, ..., N \tag{1}$$

where $\mathbf{X}_i$ and $\mathbf{Z}_i$ are, respectively, the $n_i \times P$ and $n_i \times Q$ matrices of fixed and random covariates* in the i^{th} group; $\boldsymbol{\beta}$ is the P-dimensional vector of fixed coefficients and $\mathbf{b}_i$ the Q-dimensional† vector of random coefficients relative to the i^{th} group. In the parametric framework, the random coefficients are assumed to be normally distributed, i.e., $\mathbf{b}_i \sim \mathcal{N}_Q(\mathbf{0}, \boldsymbol{\Sigma}_{\mathbf{b}_i})$, $i = 1, ..., N$. For the estimation of the model parameters within the frequentist approach, likelihood-based approaches are used.

Following the approach presented in Masci et al. (2019) and Masci et al. (2022), we move to a nonparametric framework, assuming the random effects $\mathbf{b}_1, ..., \mathbf{b}_N$ to follow a discrete distribution $\mathcal{P}$ composed by an *a priori* unknown number of support points. The proposed algorithm starts by assuming a number of discrete masses equal to the number of groups, N. It then iteratively reduces the number by combining groups into $M < N$ *clusters*. This process allows for identification of a latent structure in which groups within the same cluster exhibit a certain degree of similarity. The number of discrete masses, M, is determined by the algorithm in conjunction with the estimation of other model parameters. In this nonparametric framework, we define the *latent* variables $\mathbf{c}_1, ..., \mathbf{c}_M$ as the set of random coefficients where each $\mathbf{c}_m \in \mathbb{R}^Q$ $m = 1, ..., M$ corresponds to the random coefficient of the m^{th} cluster. These latent variables are related to each previously defined random effect $\mathbf{b}_i$ through the relationship $p(\mathbf{b}_i = \mathbf{c}_m) = \omega_m$ for $m = 1, ..., M$ where $\omega_1, ..., \omega_M$ is a set of weights such that $\sum_{m=1}^{M} \omega_m = 1$ and $\omega_m \geq 0$. The i^{th} group is assigned with probability ω_m to a cluster m with parameter values $\mathbf{c}_m$ allowing the identification of a latent structure among the groups. Consequently, starting from the GLMM formulation in Eq. (1), we make the dependence on m (and j) explicit and we get our SPGLMM formulation:

$$g(\,\mathbb{E}[y_{ij}|\mathbf{c}_m]\,) = \eta_{ijm} = \mathbf{x}'_{ij}\boldsymbol{\beta} + \mathbf{z}'_{ij}\mathbf{c}_m \quad \text{for} \quad i = 1, ..., N,\ j = 1, ..., n_i,\ m = 1, ..., M \tag{2}$$

where $\mathbf{x}_{ij}$ is he P-dimensional vector and $\mathbf{z}_{ij}$ the Q-dimensional vector of covariates relative to the $(i, j)^{\text{th}}$ observation.

The marginal likelihood $\mathcal{L}(\boldsymbol{\beta}, \mathbf{b}_1, ..., \mathbf{b}_N|\mathbf{y}) = \prod_{i=1}^{N} \prod_{j=1}^{n_i}\ p(y_{ij}|\boldsymbol{\beta}, \mathbf{b}_i)$, where $p(y_{ij}|\boldsymbol{\beta}, \mathbf{b}_i)$ denotes the conditional probability mass (or density) function of y_{ij} given random and fixed

*In many cases, the matrix $\mathbf{Z}_i$ is created by selecting a subset of appropriate columns of the matrix $\mathbf{X}_i$, i.e. the corresponding fixed and random effects are *coupled* (Gałecki et al. (2013)).

†In the theoretical discussion, we address the general case of $Q \in \mathbb{N} \setminus \{0\}$; however, in the simulation study in Section 4, we restrict ourselves to the case $Q \leq 2$, for which the model is composed by either a random intercept only, a random slope only (cases $Q = 1$) or both ($Q = 2$).

effects. Including the latent variables with the corresponding contribution of the weights of the mixture (Aitkin (1999)), the loglikelihood $\ln \mathcal{L}$ can be expressed as

$$\ln \mathcal{L}(\boldsymbol{\beta}, \mathbf{c}_1, ..., \mathbf{c}_M | \mathbf{y}) = \sum_{m=1}^{M} \omega_m \sum_{i=1}^{N} \sum_{j=1}^{n_i} \ln p(y_{ij} | \boldsymbol{\beta}, \mathbf{c}_m). \tag{3}$$

By maximizing the quantity in Eq. (3) we jointly estimate the values of $\boldsymbol{\beta}$, $(\mathbf{c}_1, ..., \mathbf{c}_M)$ and $(\omega_1, ..., \omega_M)$. We develop a tailored EM algorithm (Dempster et al. (1977)), that we discuss in Subsection 2.2. In Appendix A, we express the loglikelihood in Eq. (3) for the two special cases of Bernoulli and Poisson distributions.

2.2 EM algorithm for SPGLMM

Inspired by Aitkin (1999) and Azzimonti et al. (2013), we implement an EM algorithm to obtain the pointwise estimates $\hat{\boldsymbol{\beta}}$, $(\hat{\mathbf{c}}_1, ..., \hat{\mathbf{c}}_M)$ and $(\hat{\omega}_1, ..., \hat{\omega}_M)$ of the unknown parameters through the evaluation and the maximization of the (log) likelihood. Specifically, the EM algorithm is an iterative procedure that alternates between two steps: the *expectation step*, in which the conditional expectation of the log-likelihood ($\ln \mathcal{L}$) is computed with respect to the random effects, given the parameters and observations obtained from the previous iteration; and the *maximization step*, in which the conditional expectation of $\ln \mathcal{L}$ is numerically maximized. The algorithm terminates when either convergence is achieved or a maximum number of iterations is reached. The parameters updates are given by:

$$\hat{\omega}_m^{(up)} = \frac{\sum_{i=1}^{N} \hat{W}_{im}}{N} \quad \text{for} \quad m = 1, ..., M \tag{4}$$

where

$$\begin{aligned} \hat{W}_{im} &= \frac{\hat{\omega}_m \, p(\mathbf{y}_i | \hat{\boldsymbol{\beta}}, \hat{\mathbf{c}}_m)}{\sum_{k=1}^{M} \hat{\omega}_k \, p(\mathbf{y}_i | \hat{\boldsymbol{\beta}}, \hat{\mathbf{c}}_k)} = \frac{p(\mathbf{b}_i = \hat{\mathbf{c}}_m) \, p(\mathbf{y}_i | \hat{\boldsymbol{\beta}}, \hat{\mathbf{c}}_m)}{p(\mathbf{y}_i | \hat{\boldsymbol{\beta}})} = \frac{p(\mathbf{y}_i, \mathbf{b}_i = \hat{\mathbf{c}}_m | \hat{\boldsymbol{\beta}})}{p(\mathbf{y}_i | \hat{\boldsymbol{\beta}})} \\ &= p(\mathbf{b}_i = \hat{\mathbf{c}}_m | \mathbf{y}_i, \hat{\boldsymbol{\beta}}) \quad \text{for} \quad m = 1, ..., M, \; i = 1, ...N \end{aligned} \tag{5}$$

and

$$(\hat{\boldsymbol{\beta}}^{(up)}, \hat{\mathbf{c}}_1^{(up)}, ..., \hat{\mathbf{c}}_M^{(up)}) = \arg\max_{\boldsymbol{\beta}, \mathbf{c}_m} \sum_{m=1}^{M} \sum_{i=1}^{N} \hat{W}_{im} \; \ln p(\mathbf{y}_i | \boldsymbol{\beta}, \mathbf{c}_m) \; . \tag{6}$$

The proof of the increasing likelihood property and the derivation of the updates in Eqs. (4) and (6) can be found in Section S1 of Supplementary Materials. The weight $\hat{\omega}_m^{(up)}$ in Eq. (4) corresponds to the sample mean over the N groups of all the weights relative to the m^{th} cluster. $\hat{W}_{im}$ represents the probability that group i belongs to cluster m, conditionally on observations $\mathbf{y}_i$ and fixed coefficients $\hat{\boldsymbol{\beta}}$. The maximization in Eq. (6) involves two different

steps, performed iteratively: in the first step, we compute $\hat{\mathbf{c}}_m^{(up)}$ maximizing with respect to the support points of the random coefficients $\mathbf{c} = (\mathbf{c}_1, ..., \mathbf{c}_M)$, setting $\hat{\boldsymbol{\beta}}$ equal to the values computed at the previous iteration, namely

$$\hat{\mathbf{c}}_m^{(up)} = \arg\max_{\mathbf{c}} \sum_{i=1}^{N} \hat{W}_{im} \ \ln p(\mathbf{y}_i|\hat{\boldsymbol{\beta}}, \mathbf{c}) \quad \text{for} \quad m = 1, ..., M \ . \tag{7}$$

In the second step, we fix the support points of the random coefficients computed in the previous step and we compute the arg max of Eq. (6) with respect to $\boldsymbol{\beta}$, namely:

$$\hat{\boldsymbol{\beta}}^{(up)} = \arg\max_{\boldsymbol{\beta}} \sum_{m=1}^{M} \sum_{i=1}^{N} \hat{W}_{im} \ \ln p(\mathbf{y}_i|\boldsymbol{\beta}, \hat{\mathbf{c}}_m) \ . \tag{8}$$

In order to compute the point estimate $\hat{\mathbf{b}}_i$ of the coefficients $\mathbf{b}_i$ of the random effects for each group $i = 1, ..., N$, we maximize over m the conditional probability $p(\mathbf{b}_i = \hat{\mathbf{c}}_m|\mathbf{y}_i, \hat{\boldsymbol{\beta}})$. For Eq. (5), the estimation of $\hat{\mathbf{b}}_i$ is given by the maximization of $\hat{W}_{im}$ over m, namely:

$$\hat{\mathbf{b}}_i = \hat{\mathbf{c}}_{\tilde{l}_i} \quad \text{where} \quad \tilde{l}_i = \arg\max_{m} \hat{W}_{im} \quad \text{for} \quad i = 1, ..., N \ . \tag{9}$$

All details concerning parameters initialization procedure are addressed in Section S2.1 of Supplementary Materials.

2.3 Support points reduction criterion

In each of the k iterations of the algorithm, we aim to identify the latent structure composed of $M < N$ clusters by reducing the support of the random effects discrete distribution by making points *very close* to each other collapse. The notion of *very close* needs to be defined. In the state-of-the-art papers dealing with nonparametric random effects, at each iteration until convergence, the discrete masses with Euclidean distance lower than a chosen threshold (denoted by t*), are made collapsed. In this work, on the other hand, we suggest identifying the latent cluster structure by only means of the *conventional* confidence levels, gaining in interpretability within the classical framework of the inferential statistics and untying from the choice of a *discretionary* threshold. More specifically, we propose (i) to compute the confidence regions (intervals) of level 1-α centered in each of the two closest - in terms of Euclidean distance - estimated support points, exploiting the properties of the MLEs (Section 2.3.1) and (ii) to collapse the two discrete masses to a unique point, if the two confidence regions (intervals) overlap (Section 2.3.2).

*In the following, we will refer to this method as *t-criterion*. Such criterion is deepened and discussed in Section 4.

2.3.1 The computation of the confidence regions (intervals) for a MLE

Let $\hat{\boldsymbol{\theta}}$ be a MLE and $\boldsymbol{\theta}_0$ the true value. The Hessian matrix H of the loglikelihood function is defined as $[H(\boldsymbol{\theta})]_{ij} = [\nabla^2 \ln \mathcal{L}(\boldsymbol{\theta})]_{ij} = \dfrac{\partial^2 \ln \mathcal{L}(\boldsymbol{\theta})}{\partial\theta_i\, \partial\theta_j}$. The Fisher Information Matrix $\mathcal{I}(\hat{\boldsymbol{\theta}})$ is defined as $\mathcal{I}(\hat{\boldsymbol{\theta}}) = -\mathbb{E}[H(\boldsymbol{\theta})|\hat{\boldsymbol{\theta}}]$ and the variance-covariance matrix (King (1998), Long and Freese (2006)) is $\text{var}(\hat{\boldsymbol{\theta}}) = \mathcal{I}^{-1}(\hat{\boldsymbol{\theta}})$. Given the asymptotic efficiency property of the MLEs (Casella and Berger (2021)), we know that MLEs are asymptotically normal, i.e., $\sqrt{J}(\hat{\boldsymbol{\theta}}_J - \boldsymbol{\theta}_0) \xrightarrow[J\to\infty]{d} N(\mathbf{0}, \mathcal{I}^{-1}(\boldsymbol{\theta}_0))$, where $\xrightarrow{d}$ denotes the convergence in distribution.

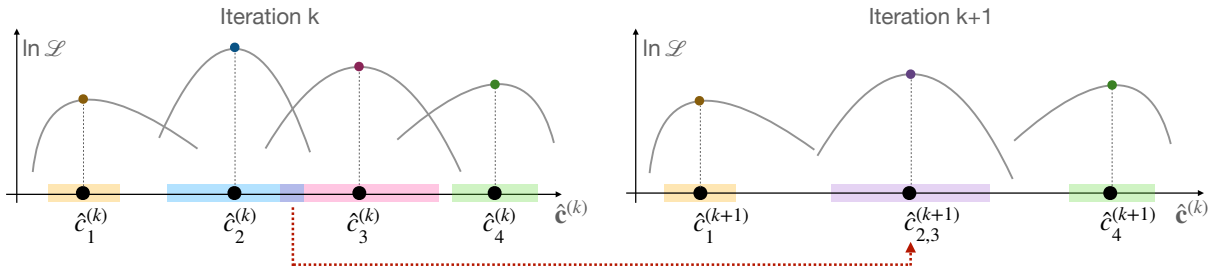
Let $\hat{\boldsymbol{\theta}}^{(k)}$ now be the MLE computed at each iteration k of our iterative algorithm. We deduce that, when $\hat{\theta}^{(k)}$ is 1-dimensional, the asymptotic confidence region is an interval CI of level $1-\alpha$ for $\hat{\theta}^{(k)}$ given by $CI_{1-\alpha}(\hat{\theta}^{(k)}) = \left[\hat{\theta}^{(k)} \pm z_{1-\frac{\alpha}{2}} \frac{1}{\sqrt{\text{var}(\hat{\theta}^{(k)})}}\right]$. Instead, when $\hat{\boldsymbol{\theta}}^{(k)}$ is Q-dimensional with $Q > 1$ and the symmetric $\text{var}(\hat{\boldsymbol{\theta}}^{(k)})$ is positive definite*, we get a confidence region with an ellipsoidal shape defined by $CR_{1-\alpha}(\hat{\boldsymbol{\theta}}^{(k)}) = \{\boldsymbol{\theta} \;:\; (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}^{(k)})' \,[\text{var}(\hat{\boldsymbol{\theta}}^{(k)})]^{-1}\, (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}^{(k)}) \leq \chi^2_{1-\alpha}(Q)\}$ (Johnson and Wichern (2002)).

2.3.2 α-criterion

At each iteration k of the SPGLMM algorithm, the MLE $\hat{\mathbf{c}}_m^{(k)}$ for $m = 1, ..., M$ is estimated as shown in Eq. (7). The elements $D_{l,m}^{(k)}$ of the matrix $\mathbf{D}^{(k)}$, composed by the Euclidean distances between the two MLEs $\hat{\mathbf{c}}_l^{(k)}$ and $\hat{\mathbf{c}}_m^{(k)}$ $\forall l, m = 1, ..., M$, are computed as follows $D_{l,m}^{(k)} = \sqrt{\sum_{h=1}^{Q} (\hat{c}_{lh}^{(k)} - \hat{c}_{mh}^{(k)})^2} \quad \forall l, m = 1, ..., M$. The two mass points $\hat{\mathbf{c}}_l^{(k)}$ and $\hat{\mathbf{c}}_m^{(k)}$ with minimum Euclidean distance are selected and the confidence regions (intervals) of level $1-\alpha$ centered in those mass points are computed as explained in Section 2.3.1. Thus, we check whether $CR_{1-\alpha}(\hat{\mathbf{c}}_l^{(k)})$ overlaps $CR_{1-\alpha}(\hat{\mathbf{c}}_m^{(k)})$ through the following *overlapping condition*, addressed separately for the unidimensional and multidimensional cases. In the unidimensional case, the two confidence intervals do overlap if the following inequality is satisfied: $\max\{\min\{CI_{1-\alpha}(\hat{c}_l^{(k)})\}, \min\{CI_{1-\alpha}(\hat{c}_m^{(k)})\}\} \;<\; \min\{\max\{CI_{1-\alpha}(\hat{c}_l^{(k)})\}, \max\{CI_{1-\alpha}(\hat{c}_m^{(k)})\}\}$. In the Q-dimensional case, when $Q = 2$, we determine if one ellipse is entirely contained within the other. This can occur when among the two closest MLEs $\hat{\mathbf{c}}_l^{(k)}$ and $\hat{\mathbf{c}}_m^{(k)}$, one exhibits higher values for the Information Matrix and thus has a larger confidence region. To assess this in our code, we use a *sufficient* condition that checks if the Euclidean distance between the centers of the two ellipses is smaller than the difference between the semi-minor axis length $(\chi^2_{1-\alpha}(2) \cdot \lambda_{min})^{1/2}$ of the larger ellipse and the semi-major axis length $(\chi^2_{1-\alpha}(2) \cdot \lambda_{max})^{1/2}$ of the smaller ellipse. If the sufficient condition is not met, indicating that one ellipse is not entirely inside the other, we use the Fast Ellipsoid Intersection

*If not, the same formula holds by replacing $[\text{var}(\hat{\boldsymbol{\theta}}^{(k)})]^{-1}$ with the generalized inverse and by substituting the degrees of freedom of the $\chi^2_{1-\alpha}$ from Q to the rank of $\text{var}(\hat{\boldsymbol{\theta}}^{(k)})$.

Figure 1: Support masses collapse procedure through α-*criterion* for $Q = 1$.



Notes: In the left-side chart, the confidence intervals centered in four different support points (black dots), estimated at a given iteration k, are displayed on the horizontal axis. The confidence intervals centered in $\hat{c}_2^{(k)}$ and $\hat{c}_3^{(k)}$ meet the *overlapping condition*, thus $\hat{c}_2^{(k)}$ and $\hat{c}_3^{(k)}$ are merged. The right-side chart shows the scenario in iteration $k+1$, where the support points have been reduced to three and the new confidence intervals have been recomputed.

Test* by performing a unidimensional minimization. A detailed description of the Test is addressed in Section S3 of Supplementary Materials. Such a method works for all the Q-dimensional cases in which $Q > 1$.

If $CR_{1-\alpha}(\hat{\mathbf{c}}_l^{(k)})$ overlaps $CR_{1-\alpha}(\hat{\mathbf{c}}_m^{(k)})$, the two points $\hat{\mathbf{c}}_l^{(k)}$ and $\hat{\mathbf{c}}_k^{(k)}$ collapse to a unique point, result of the *weighted*† mean among $\hat{\mathbf{c}}_l^{(k)}$ and $\hat{\mathbf{c}}_m^{(k)}$:

$$\hat{\mathbf{c}}_{l,m}^{(k)} = \frac{\hat{\omega}_l^{(k)}\hat{\mathbf{c}}_l^{(k)} + \hat{\omega}_m^{(k)}\hat{\mathbf{c}}_m^{(k)}}{\hat{\omega}_l^{(k)} + \hat{\omega}_m^{(k)}} \tag{10}$$

and the weight is updated with the sum of the weights of the two points:

$$\hat{\omega}_{l,m}^{(k)} = \hat{\omega}_l^{(k)} + \hat{\omega}_m^{(k)} \ . \tag{11}$$

In this way, at each iteration k we compute $\hat{\mathbf{c}}_{l,m}^{(k)}$ and $\hat{\omega}_{l,m}^{(k)}$, which will be used in iteration $k+1$ as new mass and weight. Instead, if the two confidence regions centered on the two closest mass points do not overlap, the *overlapping condition* is then checked for all other pairs of mass points (ordered by increasing Euclidean distance) until either two confidence regions overlap or all pairs have been checked. A graphical representation of the masses collapse procedure for α-*criterion* is reported in Figures 1 and 2 for $Q = 1$ and $Q = 2$, respectively.

We report the pseudo-code for the SPGLMM with α-*criterion* in Algorithm 1. For easier comparison with the state-of-the-art collapsing criterion, also the *t-criterion* is reported and the main differences between the two criteria are highlighted.

*https://github.com/NickAlger/nalger_helper_functions/blob/master/tutorial_notebooks/ellipsoid_intersection_test_tutorial.ipynb

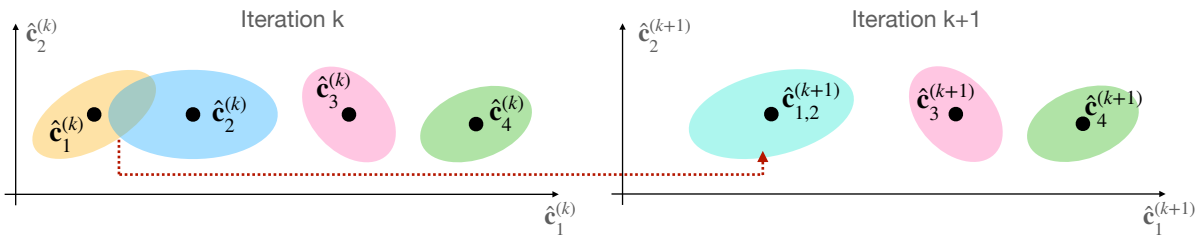
†In the earlier proposed literature, Eq. (10) was a classical (non-weighted) mean for the *t*-criterion. In our methodology, we propose a weighted mean for the *alpha*-criterion. This enables us to progressively approach the desired mass point as we iterate towards convergence.

Pseudo-code 1: SPGLMM

function SPGLMM(Initial estimates for $(\hat{\mathbf{c}}_1^{(0)}, ..., \hat{\mathbf{c}}_{M=N}^{(0)})$, $(\hat{\omega}_1^{(0)}, ..., \hat{\omega}_{M=N}^{(0)})$ and $\hat{\boldsymbol{\beta}}^{(0)}$; Tolerance parameters t (only for *t-criterion*), α (only for *α-criterion*), K, K1, K2 (only for *α-criterion*), itmax, tR, tF)

- $k \leftarrow 1$; $conv1 \leftarrow 0$; $conv2 \leftarrow 0$
- **while** ($conv1$ is 0 or $conv2$ is 0) and $k <$ K **do**
  - $\mathbf{D}^{(k)} \leftarrow$ compute distance matrix $\mathbf{D}$ as in Section 2.3.2 ▷ **only for *t-criterion***
  - **while** $\sum_{l,m}(D_{l,m}^{(k)} < t) \neq M^2$ and $\sum_{l,m}(D_{l,m}^{(k)} < t) > M$ **do**
    - Select $\hat{\mathbf{c}}_l^{(k)}$ and $\hat{\mathbf{c}}_m^{(k)}$ for which $D_{l,m}^{(k)}$ is minimum
    - Collapse $\hat{\mathbf{c}}_l^{(k)}$ and $\hat{\mathbf{c}}_m^{(k)}$ to unique mass point and update as in Eq. (10)
    - Update weights as in Eq. (11)
    - $\mathbf{D}^{(k)} \leftarrow$ Compute distance matrix $\mathbf{D}$ as in Section 2.3.2
    - $M \leftarrow M - 1$
  - $\hat{\mathbf{W}}^{(k)} \leftarrow$ compute $\hat{\mathbf{W}}$ as in Eq. (5)
  - M, $(\hat{\mathbf{c}}_1^{(k)}, .., \hat{\mathbf{c}}_M^{(k)})$, $(\hat{\omega}_1, ^{(k)} .., \hat{\omega}_M^{(k)})$, $conv2$, $conv1 \leftarrow$ Check weights ▷ See Suppl. Mat. S2.2
  - $\hat{\mathbf{W}}^{(k)} \leftarrow$ update $\hat{\mathbf{W}}$ as in Eq. (5)
  - $it \leftarrow 1$; $\hat{\mathbf{c}}^{(it-1)} \leftarrow \hat{\mathbf{c}}^{(k)}$; $\hat{\boldsymbol{\beta}}^{(it-1)} \leftarrow \hat{\boldsymbol{\beta}}^{(k)}$
  - $\hat{\mathbf{c}}^{(it)} \leftarrow$ Update the M support points according to Eq. (7), keeping $\hat{\boldsymbol{\beta}}$ fixed
  - $\hat{\boldsymbol{\beta}}^{(it)} \leftarrow$ Update $\hat{\boldsymbol{\beta}}$ according to Eq. (8), keeping $\hat{\mathbf{c}}$ fixed
  - **while** $\sum(|\hat{\boldsymbol{\beta}}^{(it)} - \hat{\boldsymbol{\beta}}^{(it-1)}| >$ tF) is 0 and $\sum(|\hat{\mathbf{c}}^{(it)} - \hat{\mathbf{c}}^{(it-1)}| >$ tR) is 0 and $it <$ itmax **do**
    - $it \leftarrow it + 1$
    - $\hat{\mathbf{c}}^{(it)} \leftarrow$ Update the M support points according to Eq. (7), keeping $\hat{\boldsymbol{\beta}}$ fixed
    - $\hat{\boldsymbol{\beta}}^{(it)} \leftarrow$ Update $\hat{\boldsymbol{\beta}}$ according to Eq. (8), keeping $\hat{\mathbf{c}}$ fixed
  - $\hat{\mathbf{c}}^{(k)} \leftarrow \hat{\mathbf{c}}^{(it)}$; $\hat{\boldsymbol{\beta}}^{(k)} \leftarrow \hat{\boldsymbol{\beta}}^{(it)}$
  - **if** $k >$ K2 **then** ▷ **only for *α-criterion***
    - not_merged $\leftarrow 1$
    - $\mathbf{D}^{(k)} \leftarrow$ compute distance matrix $\mathbf{D}$ as in Section 2.3.2
    - **while** not_merged is 1 and sum(NaN values in D)<dim(D) **do**
      - Select $\hat{\mathbf{c}}_l^{(k)}$ and $\hat{\mathbf{c}}_m^{(k)}$ for which $D_{l,m}^{(k)}$ is minimum
      - **if** *overlapping condition* is satisfied **then**
        - Collapse $\hat{\mathbf{c}}_l^{(k)}$ and $\hat{\mathbf{c}}_m^{(k)}$ to unique mass point and update as in Eq. (10)
        - Update weights as in Eq. (11)
        - $M \leftarrow M - 1$
        - not_merged $\leftarrow 0$
      - **else**
        - Set $D_{l,m}^{(k)} =$ NaN
  - **if** $\sum(|\hat{\boldsymbol{\beta}}^{(k)} - \hat{\boldsymbol{\beta}}^{(k-1)}| >$ tF) is 0 and $\sum(|\hat{\mathbf{c}}^{(k)} - \hat{\mathbf{c}}^{(k-1)}| >$ tR) is 0 **then**
    - **if** sum(NaN values in D)==dim(D) **then** ▷ **only for *α-criterion***
      - $conv1 \leftarrow 1$
  - $k \leftarrow k + 1$
- **return** Final estimates of $(\hat{\mathbf{c}}_1^{(k)}, ..., \hat{\mathbf{c}}_M^{(k)})$, $(\hat{\omega}_1^{(k)}, ..., \hat{\omega}_M^{(k)})$ and $\hat{\boldsymbol{\beta}}^{(k)}$; $\hat{\mathbf{W}}^{(k)}$

Figure 2: Support masses collapse procedure through α-*criterion* for $Q = 2$.



Notes: In the left-side chart, the confidence regions centered in four different support points (black dots), estimated at a given iteration k, are displayed in the plan. The ellipses centered in $\hat{\mathbf{c}}_1^{(k)}$ and $\hat{\mathbf{c}}_2^{(k)}$ do intersect, hence $\hat{\mathbf{c}}_1^{(k)}$ and $\hat{\mathbf{c}}_2^{(k)}$ are collapsed. The right-side chart shows the situation in iteration $k+1$, where the support points have been reduced to three and the new confidence regions have been recomputed.

Final checks concerning the support reduction and details regarding convergence criteria are addressed in Sections S2.2 and S2.3 of Supplementary materials.

3 Case study: application to innumeracy rates

In this section, we apply the SPGLMM to data extracted from PISA survey of 2018, available online at https://www.oecd.org/pisa/data/2018database/. Students' scores in mathematics tests are divided into Levels of Proficiency, as described in Chapter 6 of OECD (2019), with Level 2 being the minimum required by global indicators for the United Nations Sustainable Development Goals to be acquired by the end of secondary education. Level 2 proficiency only provides a basic understanding of math for simple real-life situations and does not prepare students for decision-making requiring mathematical literacy. For this reason, students below such a level of proficiency are considered as *low-achieving students*. We develop a model to predict the percentage number of low-achieving students in each school and country, taking into account school characteristics like size and socio-economic status. Additionally, we aim to identify groups of countries that have a similar impact on the rate of low-achievers by using a SPGLMM with Poisson response.

The survey provides, among others, data both at the *student* and at the *school level*. Table 1 reports the selected variables extracted from OECD PISA dataset, together with their description. Low-achieving students are identified by `PV1MATH` scores below 482.38, i.e., the ones with Proficiency levels strictly less than Level 3 (OECD (2019)).

In the following sections, we will address the data preprocessing (Section 3.1), the model formulation (Section 3.2) and the results obtained by fitting the model with Poisson response (Section 3.3). Parallel handling for the Bernoulli response is addressed in Section S4 of Supplementary materials.

Table 1: List, description and summary statistics of variables extracted from OECD PISA dataset.

Variable	Description	Type	Summary statistics
`ESCS`	Index of economic, social and cultural status [*student level*]	Continuous	mean = -0.29, sd = 1.11, median = -0.18, [min; max] = [-8.17; 4.21]. 14379 NaNs (2.35%)
`PV1MATH`	Score** in mathematics [*student level*]	Continuous	mean = 461.88, sd = 104.49, median = 461.39, [min; max] = [24.74; 888.06]
`SCHSIZE`	School size (sum) [*school level*]	Continuous	mean = 839.65, sd = 869.61, median = 624, [min; max] = [1; 13400]. 3582 NaNs (16.35%)
`CNTSCHID`	International school id [*school level*, *student level*]	Categorical	21903-levels factor
`CNT`	Country code 3-character [*school level*, *student level*]	Categorical	82-levels factor

**More precisely, we considered Plausible Value 1 (OECD (2019)); Plausible Values are a selection of likely proficiencies for students' attained scores, i.e., multiple imputations of the unobservable latent achievement for each student.

3.1 Data preprocessing

After having discarded missing values, continuous variables at the *student level* are aggregated at the *school level*: specifically, for each school we consider (i) `avg_ESCS_std`, the average students' `ESCS`, subsequent to a standardization (mean 0 and standard deviation 1) within the country of the school, for keeping into account differences between countries and (ii) `Y_MATH`, the rounded percentage of students with a proficiency level strictly less than Level 3 (low-achieving students). The analysis is restricted to schools with a minimum of 10 students to ensure more accurate results. A dataset at the school level containing information on 12620 schools (the variable `CNTSCHID` becomes a 12620-levels factor) nested within 50 countries (`CNT` becomes a 50-levels factor) is created. Moreover, the two predictors `avg_ESCS_std` and `SCHSIZE` are further standardized.

3.2 Model formulation

We consider a two-level SPGLMM, as in Eq. (2), and we employ the α-*criterion*. For each country i, with $i = 1, ...N$, and each school j, with $j = 1, ..., n_i$, given that $N = 50$ is the total number of countries and $J = 12620$ the total number of schools, the model is

$$g(\,\mathbb{E}[y_{ij}|c_m]\,) = \eta_{ijm} = \mathbf{x}'_{ij}\boldsymbol{\beta} + c_m \quad \text{for} \quad i = 1, ..., N,\ j = 1, ..., n_i,\ m = 1, ..., M_\alpha \qquad (12)$$

where M_α is the total number of clusters the model identifies and depends on the level of confidence α chosen for the α-*criterion*; $\mathbf{x}'_{ij}$ is the two-dimensional vector of fixed effects

covariates at the school level that contains $\texttt{SCHSIZE}_{ij}$ and $\texttt{avg\_ESCS\_std}_{ij}$; $\boldsymbol{\beta} = [\beta_1, \beta_2]'$ is the two-dimensional vector of fixed effects coefficients; c_m is the random intercept relative to the m^{th} cluster. The response $\mathbf{y}_i$ is given by $\texttt{Y\_MATH}_i$ and is assumed to be Poisson distributed. To validate this assumption, a Chi-Square goodness of fit test was conducted and the distribution was visually inspected using Q-Q plots (Wilk and Gnanadesikan (1968)) for Poisson distribution. The link function is assumed to be the canonical $g(\cdot) = \ln(\cdot)$ (see Appendix A.2).

We run the SPGLMM algorithm with $\texttt{K} = 60$, $\texttt{K1} = 20$, $\texttt{K2} = 5$, $\texttt{itmax} = 20$, $\texttt{tR} = \texttt{tF} = 10^{-5}$ and, in turn, $\alpha = 0.01, 0.05$ and 0.10. The algorithm starts with $M = N = 50$ support points and the support weights are uniformly initialized on the M support points, as explained in Section S2.1 of Supplementary Materials. To better assess the validity and the robustness of the obtained results, we report them together with the estimates of the parametric GLMM, fitted through *glmer* R function from package *lme4* (Bates et al. (2015), R Core Team (2022)), which assumes the following formulation:

$$g(\,\mathbb{E}[y_{ij}|b_i]\,) = \eta_{ij} = \mathbf{x}'_{ij}\boldsymbol{\beta} + b_i \quad \text{for} \quad i = 1, ..., N,\ j = 1, ..., n_i \tag{13}$$

All the terms are the same as in Eq. (12), except for b_i, which is the random intercept relative to the i^{th} country. Also in this case, the fixed intercept is not included in the model.

3.3 Results

SPGLMM outputs for Poisson response with $\alpha = 0.01, 0.05, 0.10$ are addressed in Table 2. We get $\hat{M}_{0.01} = 13$, $\hat{M}_{0.05} = 16$ and $\hat{M}_{0.10} = 18$. As expected, at higher values of α correspond higher values of M. Indeed, the higher is α, the smaller the confidence intervals and less likely to overlap (see Section 4 for a further discussion). The nomenclature of $\hat{\mathbf{c}}$ in the leftmost column of Table 2 is in harmony with the estimates obtained for $\alpha = 0.10$ (i.e. the highest α for which the algorithm is run), where the random intercepts $\hat{c}_1, ..., \hat{c}_{18}$ are reported on 18 different rows. For α equal to 0.01 and 0.05, the algorithm identifies fewer clusters and the estimated random intercept is reported in between two rows, to indicate that two distinct clusters were merged into one. The gray or white backgrounds indicate whether discrepancies between the outputs with different α occur*. In addition, on the rightmost column of Table 2, we report the results obtained with the parametric GLMM of Eq. (13) with Poisson response. In the first 18 rows, we report the means of the random intercepts $b_1, ..., b_{50}$ computed by the GLMM within each cluster $m = 1, ..., 18$. We can

*For instance, with $\alpha = 0.01$ and $\alpha = 0.05$, in correspondence of $\hat{c}_2$ and $\hat{c}_3$ (reported on white background), only one value of random intercept is identified. We will denote this random intercept with $\hat{c}_{2+3}$, meaning for simplicity the random intercept associated with cluster $2+3$. This coefficient turns out to be a weighted mean between $\hat{c}_2$ and $\hat{c}_3$, as expected from Eq. (10). Remarkable is the case in correspondence of $\hat{c}_{12}$, $\hat{c}_{13}$ and $\hat{c}_{14}$ (all reported on white background), where the countries in the cluster 13, when $\alpha = 0.01$, are partially assigned to the cluster 12 and partially to the cluster 14.

Table 2: SPGLMM estimates and comparison with GLMM output.

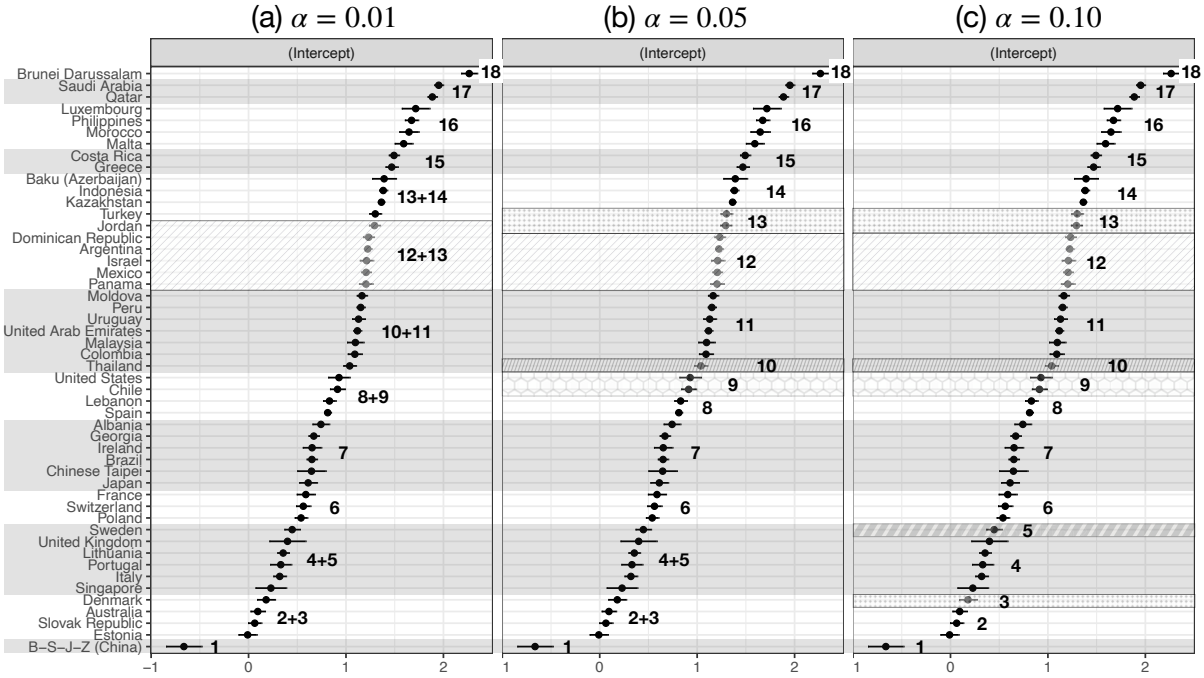
Coeff. estimates		**SPGLMM**			**GLMM**
		$\alpha = 0.01$	$\alpha = 0.05$	$\alpha = 0.10$	
$\hat{\mathbf{c}}$ [$\hat{\omega}$]	$\hat{c}_1$ [0.02]	-0.668 (0.092)	-0.668 (0.092)	-0.668 (0.092)	-0.664
	$\hat{c}_2$ [0.06]	0.082 (0.020)	0.082 (0.020)	0.060 (0.022)	0.050
	$\hat{c}_3$ [0.02]			0.183 (0.044)	0.180
	$\hat{c}_4$ [0.10]	0.350 (0.018)	0.350 (0.018)	0.337 (0.020)	0.327
	$\hat{c}_5$ [0.02]			0.445 (0.041)	0.448
	$\hat{c}_6$ [0.06]	0.549 (0.021)	0.549 (0.021)	0.559 (0.021)	0.563
	$\hat{c}_7$ [0.12]	0.656 (0.015)	0.655 (0.015)	0.656 (0.016)	0.662
	$\hat{c}_8$ [0.04]	0.831 (0.014)	0.813 (0.015)	0.813 (0.015)	0.821
	$\hat{c}_9$ [0.04]		0.917 (0.033)	0.917 (0.033)	0.921
	$\hat{c}_{10}$ [0.02]	1.122 (0.010)	1.048 (0.030)	1.048 (0.03)	1.038
	$\hat{c}_{11}$ [0.12]		1.132 (0.011)	1.133 (0.011)	1.125
	$\hat{c}_{12}$ [0.10]	1.229 (0.011)	1.217 (0.012)	1.217 (0.012)	1.216
	$\hat{c}_{13}$ [0.04]	1.361 (0.012)	1.296 (0.022)	1.296 (0.022)	1.297
	$\hat{c}_{14}$ [0.06]		1.369 (0.012)	1.369 (0.012)	1.379
	$\hat{c}_{15}$ [0.04]	1.482 (0.020)	1.483 (0.020)	1.483 (0.02)	1.481
	$\hat{c}_{16}$ [0.08]	1.654 (0.023)	1.654 (0.023))	1.655 (0.023)	1.657
	$\hat{c}_{17}$ [0.04]	1.920 (0.017)	1.920 (0.017)	1.920 (0.017)	1.919
	$\hat{c}_{18}$ [0.02]	2.268 (0.041)	1.268 (0.041)	2.268 (0.041)	2.265
$\hat{\boldsymbol{\beta}}$	$\hat{\beta}_1$	-1.361 (0.007) ***	-1.361 (0.007) ***	-1.361 (0.007) ***	-1.359 (0.012) ***
	$\hat{\beta}_2$	-0.094 (0.004) ***	-0.094 (0.004) ***	-0.094 (0.004) ***	-0.095 (0.004) ***

Notes: The estimated random intercepts $\hat{\mathbf{c}}$ are presented in increasing order, together with their respective weights $\hat{\omega}$ in brackets, as well as the fixed effects $\hat{\boldsymbol{\beta}}$ for both SPGLMM (with $\alpha = 0.01, 0.05, 0.10$) and GLMM (for each row of $\hat{c}_m$, the average of the $\hat{b}_i$s in each cluster m is reported). In parenthesis, the standard error is computed by square rooting the inverse of the Fisher Information Matrix. For $\hat{\boldsymbol{\beta}}$, the p-value is estimated by means of likelihood-ratio test (* p-value < 0.1; ** p-value < 0.01; *** p-value < 0.001).

appreciate that the means of the random intercepts $\hat{b}_i$ in each cluster are slightly lower in absolute value than the SPGLMM estimates. Anyhow, we observe huge coherence between the two models. The second part of the table is dedicated to the fixed effects $\hat{\boldsymbol{\beta}}$. Both the two fixed slopes are negative, meaning that the percentage of low-achieving students in mathematics is inversely proportional to the school size and the index of economic, social and cultural status. Specifically, the higher the value of `SCHSIZE` and `avg_ESCS_std`, the lower the percentage of low-achieving students, though `avg_ESCS_std` has a lower impact than `SCHSIZE` (the former slope is $\beta_2 = -0.09$ compared to the latter one of $\beta_1 = -1.36$). In general, we can conclude that also for the fixed slopes, SPGLMM and GLMM provide coherent results in the estimates, the standard errors and the p-values (estimated through the likelihood-ratio test).

In the three panels in Figure 3, we display the caterpillar plots for the random intercepts (together with their confidence intervals) of the 50 countries obtained through parametric GLMM with Poisson response. On each panel, we highlight the identified clusters of coun-

Figure 3: Caterpillar plots reporting the comparison between the 50 random intercepts estimated by GLMM and the clusters obtained by SPGLMM, for $\alpha = 0.01, 0.05, 0.10$.

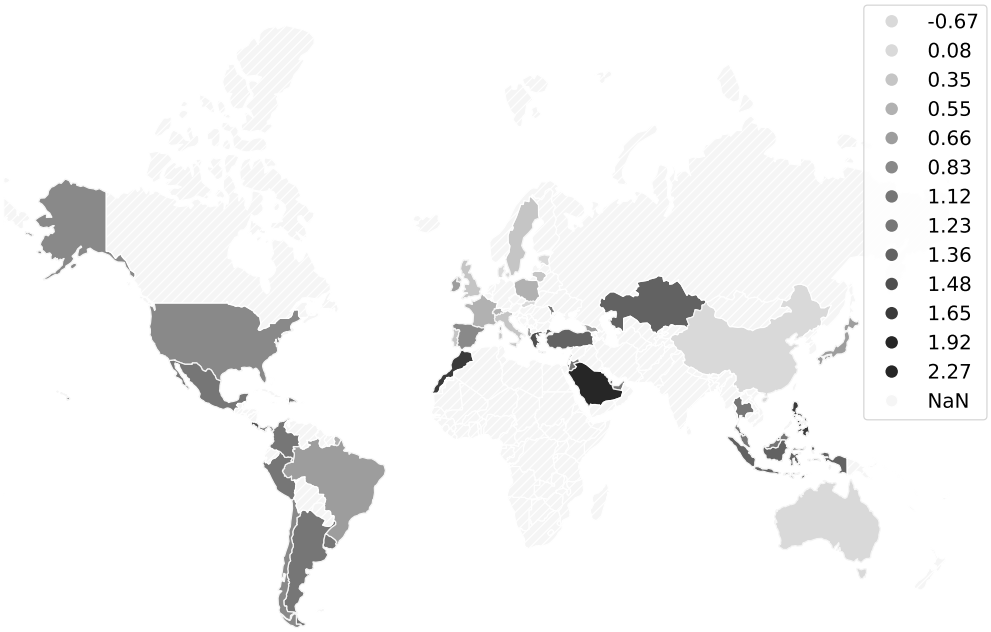


Notes: To ease the comparison with Table 2, colours used to highlight clusters of countries in panel (a) are equal to the ones used in the table. Panels (b) and (c) adopt different textures in order to better highlight the differences in the detection of the more numerous clusters. Next to the random intercepts, we report the number of the cluster, following the nomenclature of Table 2 (i.e., the plain enumeration $1, ..., M_{0.10}$).

tries, both for $\alpha = 0.01$ in panel (a), $\alpha = 0.05$ in panel (b) and $\alpha = 0.10$ in panel (c). We remind that each country i is assigned to the cluster m by maximizing the posterior conditional weight $\hat{\omega}_{im}$, as shown in Eq. (9). Results can be interpreted as follows: the lower the estimated random intercept for a cluster (i.e., the bottom countries in the caterpillar plots), the lower the percentage of low-achieving students in mathematics in the schools of the countries of that cluster, and vice-versa. For better visualization of the clusters of countries identified by the SPGLMM, we highlight with the same shade of gray on the map in Figure 4 the countries identified by the same random intercept (i.e. the countries in the same cluster), for $\alpha = 0.01$. We notice that B-S-J-Z (China) and Australia, net of the other features, decrease the percentage of low-achieving students in mathematics. After them, the European countries slightly increase it, while the Americas and other middle-east countries have a wider impact.

For the Goodness of Fit (GoF) evaluation, we consider the following metrics for integer responses: the MSE of responses ($\frac{1}{J}\sum_{i,j}(y_{ij}-\hat{y}_{ij})^2$), the MSE of log responses ($\frac{1}{J}\sum_{i,j}(log(y_{ij}+1) - log(\hat{y}_{ij}+1))^2$) and the Chi-Squared Error $\frac{1}{J}\sum_{i,j}\frac{(y_{ij}-\hat{y}_{ij})^2}{\hat{y}_{ij}+1}$ (McCullagh and Nelder (1989)). The +1 at the denominator and inside the logarithm is added because y_{ij} and $\hat{y}_{ij}$ could possibly assume value 0. The same data used for training the models are also utilized

Figure 4: Choropleth map of the clusters of countries identified by the random intercepts in SPGLMM with $\alpha = 0.01$.



Notes: Countries represented with the same color belong to the same cluster. The lighter, the lower the random intercept. Light grey-striped countries are the ones for which the survey was not performed, or which were presenting missing values.

Table 3: GoF metrics estimates for the case study, fitted via SPGLMM and GLMM.

	SPGLMM			**GLMM**
	$\alpha = 0.01$	$\alpha = 0.05$	$\alpha = 0.10$	
MSE of responses	27.461	27.405	27.398	27.406
MSE of log responses	0.284	0.283	0.282	0.282
Chi-Squared Error	2.376	2.363	2.361	2.369

for computing the predictions and the aforementioned indexes, to ensure a fair comparison between the predictive abilities of SPGLMM and GLMM. In Table 3 we report on three different rows the three metrics obtained comparing $\mathbf{y} = \texttt{Y\_MATH}$ and the predicted $\hat{\mathbf{y}}$ (retrieved by rounding $\hat{\boldsymbol{\mu}}$ to the closest integer) for each of the three SPGLMM and GLMM. The two methods reveal similar predictive performances. The SPGLMM with Poisson response does not worsen the predictions with respect to the parametric GLMM, whereas, it further provides in output a clustering of the hierarchies (i.e., the countries in our case study), revealing the inner structure the model assumes. This result is further discussed in Section 4, analyzing results obtained through a simulation study.

4 Discussion and comparison with existing methods

In this section, we discuss the accuracy and reliability of the SPGLMM with α-*criterion* proposed in Section 2, proving its well-performance against other state-of-the-art methods. More specifically, we propose a simulation study for generating sets of data with an *a priori* built latent grouping structure on which to test our SPGLMM with α-*criterion* under different settings and easily compare its results with the ones obtained by other models, i.e. SPGLMMs with t-*criterion* and parametric GLMMs. In fact, as briefly introduced in Section 2.3, all the papers in literature dealing with discrete random effects make the collapsing step of the algorithm relying on a priori chosen threshold t (from here, the denomination t-*criterion*). At each iteration until convergence, the discrete masses with Euclidean distance lower than t are made collapsed. Through such methodology, the number of obtained clusters M is not chosen *a priori*, but it intrinsically depends on the choice of t. The higher is t, the lower the number of clusters and the less homogeneous the groups within each cluster: t should be chosen depending on the required homogeneity level within clusters. The selection of the threshold t represents the main drawback when these methods are applied to real-world data, especially when no prior information on the heterogeneity among clusters is available: results could be very sensitive to t. For dealing with such a choice, we conduct in Section S6 of Supplementary Materials a sensitivity analysis and we propose a criterion for driving the choice of the threshold t based on the individuation of an elbow in the plot of the average entropy of the conditional weights matrix; nevertheless, such a criterion has the drawback to be computationally expensive, especially when dealing with massive amounts of data (e.g., the numerosity of the case study addressed in Section 3).

In our simulation study, which we perform both for Poisson response (addressed in Section 4.2) and Bernoulli response (addressed in Section S5 of Supplementary Materials), we run the SPGLMM with α-*criterion* for different confidence levels α (i.e., $\alpha = 0.01, 0.05$ and 0.10) and the t-*criterion* for different values of t and we compare their performances. The SPGLMM with t-*criterion* is, to the best of our knowledge, the only method in literature able to fit a GLMM with discrete random effects (i.e., able to cluster "the hierarchies"). As a matter of fact, similarly to what we have done for the case study, we challenge our method with two different versions of a parametric GLMM, taking into account that parametric GLMM proposes a different interpretation of the random effects, estimating a single coefficient for each group rather then clustering them.

In the next two sections, we address the set-up of the simulation study (Section 4.1) and more specifically the Poisson response (Section 4.2).

4.1 The simulation study set-up

For our simulation study, we consider $N = 10$ groups of data* and, since SPGLMM can handle different number of observations within groups, we sample the number of observations n_i from a uniform distribution $n_i \sim \mathcal{U}(70, 100)$ for $i = 1, ..., N$. We simulate the data by inducing the presence of 3 clusters. We set $\texttt{K} = 60$, $\texttt{K1} = 20$, $\texttt{itmax} = 20$, $\texttt{tR} = \texttt{tF} = 10^{-5}$ and $\texttt{K2} = 5$.

4.2 Poisson response case

For the case of a Poisson response distribution, we simulate a model presenting only a random intercept† with either one or two fixed slopes. The second fixed slope will be indicated in parenthesis in the following equations. The linear predictor $\boldsymbol{\eta}_i = \boldsymbol{\beta}_1 \mathbf{x}_{1i} + (\boldsymbol{\beta}_2 \mathbf{x}_{2i}) + c_{1i} \mathbb{1}_{n_i}$ is defined by the following DGP:

$$\boldsymbol{\eta}_i = \begin{cases} 0.3\mathbf{x}_{1i} + (0.9\mathbf{x}_{2i}) + 2.5\ \mathbb{1}_{n_i} & \text{if } i = 1, 2, \\ 0.3\mathbf{x}_{1i} + (0.9\mathbf{x}_{2i}) + 1\ \mathbb{1}_{n_i} & \text{if } i = 3, 4, 5, 6, 7, \\ 0.3\mathbf{x}_{1i} + (0.9\mathbf{x}_{2i}) - 1\ \mathbb{1}_{n_i} & \text{if } i = 8, 9, 10 \end{cases} \tag{14}$$

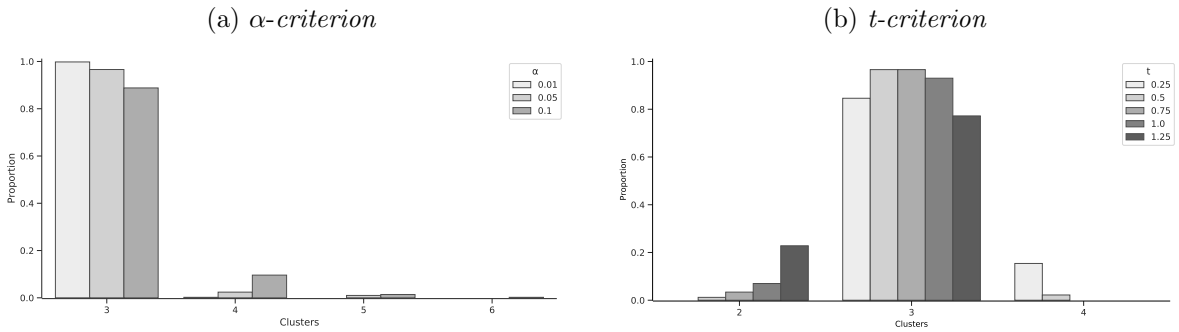
Variables $\mathbf{x}_{1i}$ and $\mathbf{x}_{2i}$ are normally distributed with mean equal to 0 and standard deviation equal to 1. The choice of the coefficients values is arbitrary. In this case, they are chosen in order to simulate different situations in which we obtain a different skewness with respect to the zero, but also avoiding generating too high numbers, which could cause numerical issues in the computation of the poisson density. After the computation of $\boldsymbol{\eta}_i$ according to DGP in Eq. (14), we retrieve $\mu_{ij} = \exp(\eta_{ij})$ and we compute $y_{ij} \sim \text{Poi}(\mu_{ij})$ for $i = 1, ..., N$ and $j = 1, ..., n_i$. Namely, y_{ij} is extracted from a Poisson distribution with mean equal to the retrieved μ_{ij}. Afterwards, we apply SPGLMM with both t- and α-*criterion*s, performing 500 runs for the setting with one fixed slope shown in Eq. (14), for different values of t and α. The values of t are chosen simmetrically around the value that maximizes the times in which the true number of clusters is identified, in order to best analyze and visualize the model behaviour.

The obtained results with only one fixed slope are collected in Figures 5, 6 and 7. By looking at Figure 5, we appreciate that the algorithm correctly identifies the true number of clusters (i.e., three) in the majority of the runs. If *t-criterion* is adopted, for all the DGPs,

*This choice is driven by a trade-off between the need of having enough groups for making SPGLMM detect non-trivial clusters and the constraint of not introducing too many groups in the generative models, which would lead to more complexity and less interpretability to the results of our simulation.

†This choice is due to the fact that in a GLMM with Poisson response, the exponential as inverse canonical link function (i.e., $\boldsymbol{\mu_i} = \exp(\boldsymbol{\eta_i})$), makes the identification of the model coefficients computationally difficult. Simulation study for the Binary response includes also the case of only random slope and both random intercept and slope.

Figure 5: Barplot for the frequency a certain number of clusters is identified over 500 runs, across different values of *alpha* and t, for DGP for Poisson response with one fixed slope.



Notes: To ease the comparison, the frequencies on y-axis are reported on the same scale. The number of identified clusters is reported on the x-axis. In panel (a), results across *alpha* $= 0.01, 0.05, 0.1$ are represented with different shadows of grey. Similarly, in panel (b) results across $t = 0.25, 0.5, 0.75, 1, 1.25$ are addressed.

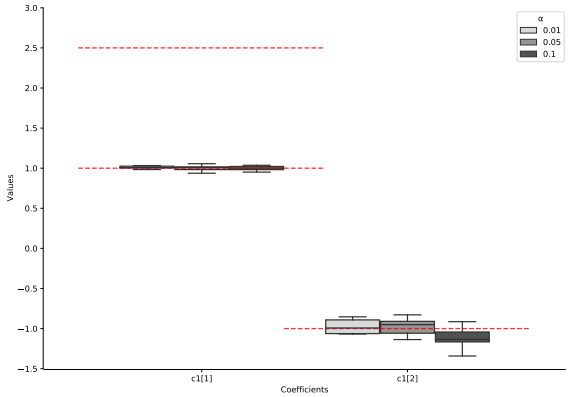
the higher the threshold, the higher the proportion of the cases in which the algorithm identifies fewer clusters than the true number; vice versa, the lower the threshold, the higher the proportion of the cases in which the algorithm identifies more clusters than the true number. In fact, decreasing the value of t, the support points of the random effects coefficients distribution with distances lower than t collapse to a unique point, and the SPGLMM algorithm is more sensitive to the variability among the groups, identifying a higher number of clusters, and vice versa. It follows that, for each DGP, there exists an *optimal* threshold, for which the proportion of the correctly identified true number of clusters is maximized. When the *α-criterion* is adopted, the proportion of such number is always very high, higher than the optimal case of the *t-criterion*; moreover, the higher is α, the higher the proportion of times in which the algorithm identifies more clusters than the true number. This happens because, as anticipated in Section 3, the higher is α, the smaller the confidence region (interval) of level $1-\alpha$ is; this induces the confidence regions (intervals) relative to different support points to overlap less and the collapse effect to have a lower impact. In Figure 6, we clearly see that when the SPGLMM identifies 3 clusters, the estimated coefficients are very close to the simulated ones and their variability is low. When the algorithm identifies a higher number of clusters with respect to the true one, it generally splits a cluster in two; vice versa, when SPGLMM identifies a lower number of clusters, it merges two clusters into one. The estimates for the fixed effects coefficients represented in Figure 7 result to be only marginally affected by the identified number of clusters, being their estimates quite robust with respect to the random effects.

For completeness, the results obtained for DGP with both one and two fixed slopes are reported respectively in Tables S7.1 and S7.2 in Section S7 of Supplementary Materials.

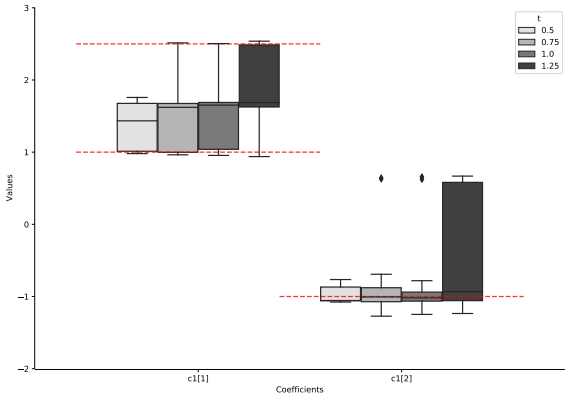
To assess the goodness of fit of SPGLMM, we compare our results to the ones obtained by a parametric GLMM fitted through the function *glmer* in R package *lme4* (Bates et al.

Figure 6: Boxplots for the random intercept $\mathbf{c}_1$ distribution for the DGP for Poisson response, with one fixed slope.

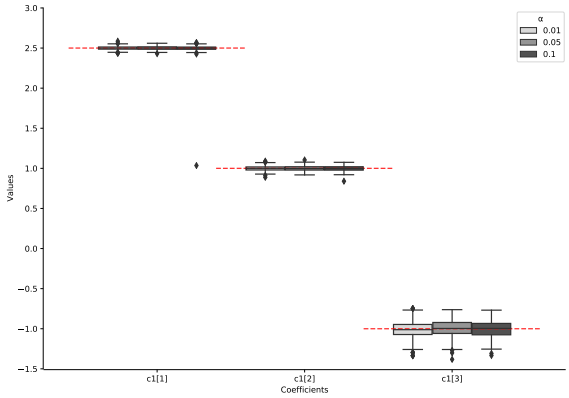
(a) $\mathbf{c}_1$ (2 clusters) - *α-criterion*



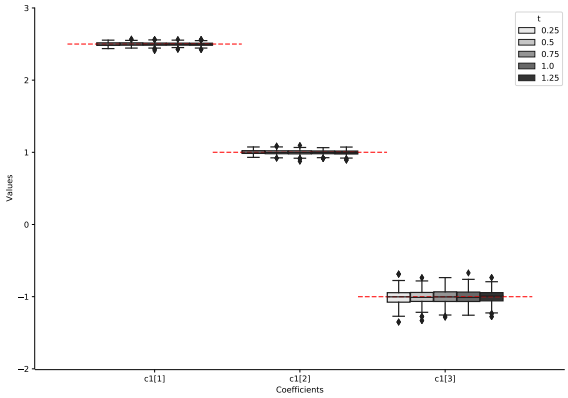
(b) $\mathbf{c}_1$ (2 clusters) - *t-criterion*



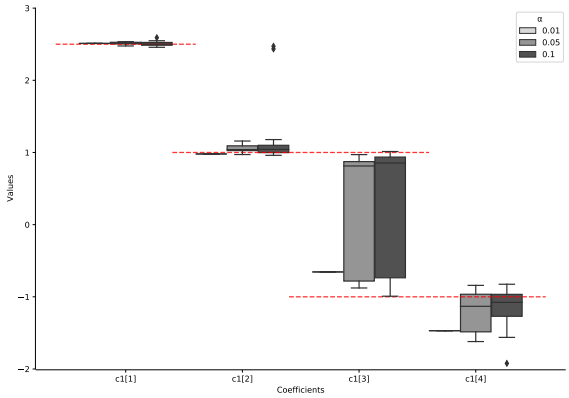
(c) $\mathbf{c}_1$ (3 clusters) - *α-criterion*



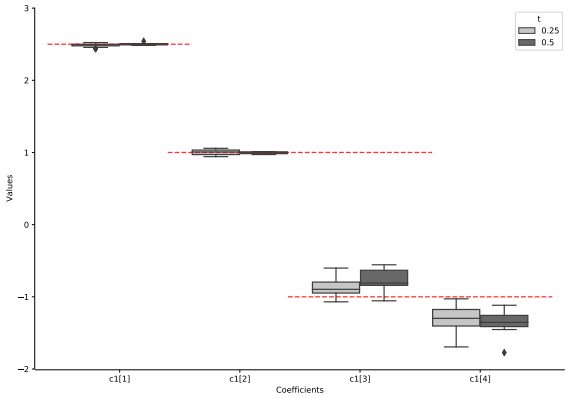
(d) $\mathbf{c}_1$ (3 clusters) - *t-criterion*



(e) $\mathbf{c}_1$ (4 clusters) - *α-criterion*



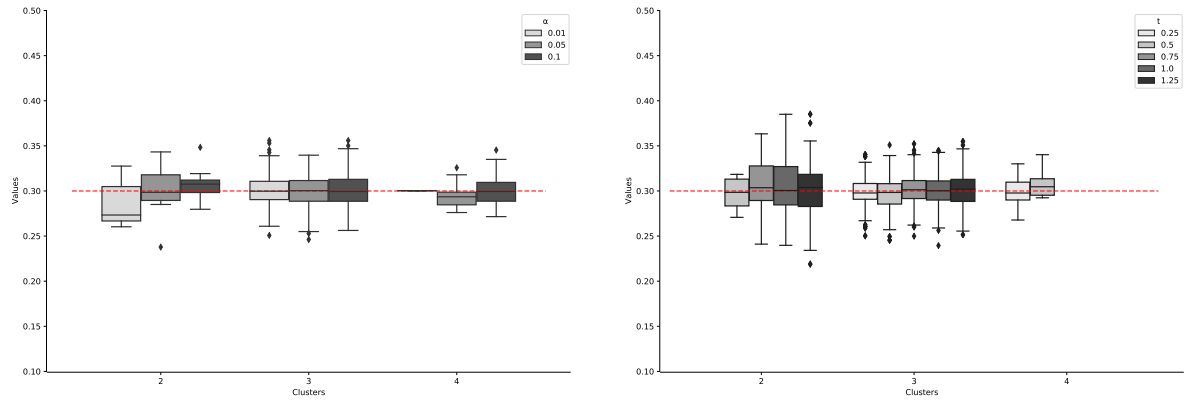
(f) $\mathbf{c}_1$ (4 clusters) - *t-criterion*



Notes: For each of the 500 runs with a chosen threshold, we represent boxplots for the values of the components of the random intercept $\mathbf{c}_1$ (y-axis), separately, according to the number of identified clusters (panels (a) and (b) for 2 clusters, panels (c) and (d) for 3 clusters, panels (e) and (f) for 4 clusters). In the left panels, we report the results of the SPGLMM run via *α-criterion*, while in the right panels the results obtained via *t-criterion*. The horizontal dotted lines indicate the simulated coefficients.

Figure 7: Boxplots for the fixed slope β_1 distribution for the DGP for Poisson response, with one fixed slope.

(a) β_1 - *α-criterion* (b) β_1 - *t-criterion*



Notes: For each of the 500 runs with a chosen threshold, we represent boxplots of the value for the fixed slope β_1 (y-axis) according to the number of identified clusters (indicated on the x-axis). In the left panel, we report the results of the SPGLMM via *α-criterion*, while in the right panel the results are obtained via *t-criterion*. The horizontal dotted lines indicate the simulated coefficients.

(2015), R Core Team (2022)). We simulate 100 different DGPs as in Eq. (14) and we fit, in turn, two different GLMMs, the first one considering a random intercept for each group $i = 1, ..., 10$ and the second one with a random intercept for each cluster $m = 1, 2, 3$. Moreover, we fit the SPGLMM with *α-criterion* with $\alpha = 0.05$, knowing that with such a value the algorithm identifies 3 clusters in the 94.8% of the times, as highlighted in Table S7.1 in Section S7 of Supplementary Materials. For each of the three models (i.e., GLMM with 10 random intercepts, GLMM with 3 random intercepts and SPGLMM), we represent through boxplots the distribution of the obtained coefficients across the 100 DGPs, emphasizing the simulated values through dotted lines, respectively in Figure 8, panels (a), (b) and (c). We report in Table 4 the summary statistics of the GoF metrics, which are computed comparing y_{ij} of the DGP and the predicted $\hat{y}_{ij}$, retrieved by rounding $\hat{\mu}_{ij}$ to the closest integer. As in Section 3, we consider the MSE of responses ($\frac{1}{J}\sum_{i,j}(y_{ij} - \hat{y}_{ij})^2$), the MSE of log responses ($\frac{1}{J}\sum_{i,j}(log(y_{ij}+1) - log(\hat{y}_{ij}+1))^2$) and Chi-Squared Error ($\frac{1}{J}\sum_{i,j}\frac{(y_{ij}-\hat{y}_{ij})^2}{\hat{y}_{ij}+1}$).

Results in Table 4 show that the SPGLMM has similar performance to GLMM ones in which three clusters are provided to the parametric model. This confirms that our semiparametric model is able to recognize the three clusters as efficiently as when we give this information in input to the model. The case in which we run a GLMM with 10 groups performs slightly better, as expected, since the models have more flexibility to adapt to the differences between the groups. Concerning the computation of the coefficients, in the boxplots in Figure 8 we can appreciate that the true values are correctly identified in all the cases, with a slightly higher presence of outliers when a GLMM with 10 groups is fitted (panel (a)).

Table 4: Summary statistics of the GoF metrics estimates for DGP for Poisson response, with GLMM (10 groups and 3 clusters) and SPGLMM (3 clusters, α-*criterion*, $\alpha = 0.05$).

				Quantile		
Model	Metric	Mean	Std. dev.	25%	50%	75%
GLMM, 10 groups	*MSE of responses*	4.0995	0.3502	3.8153	4.0987	4.3132
	MSE of log responses	0.2024	0.01269	0.1944	0.2023	0.2099
	Chi-Squared Error	0.6873	0.0388	0.6620	0.6764	0.7184
GLMM, 3 clusters	*MSE of responses*	4.1377	0.3519	3.8847	4.1328	4.3753
	MSE of log responses	0.2044	0.0129	0.1962	0.2042	0.2131
	Chi-Squared Error	0.6962	0.0385	0.6704	0.6871	0.7239
SPGLMM, 3 clusters	*MSE of responses*	4.1382	0.3519	3.8775	4.1334	4.3764
	MSE of log responses	0.2044	0.0129	0.1963	0.2043	0.2127
	Chi-Squared Error	0.6965	0.0384	0.6709	0.6870	0.7248

5 Conclusions

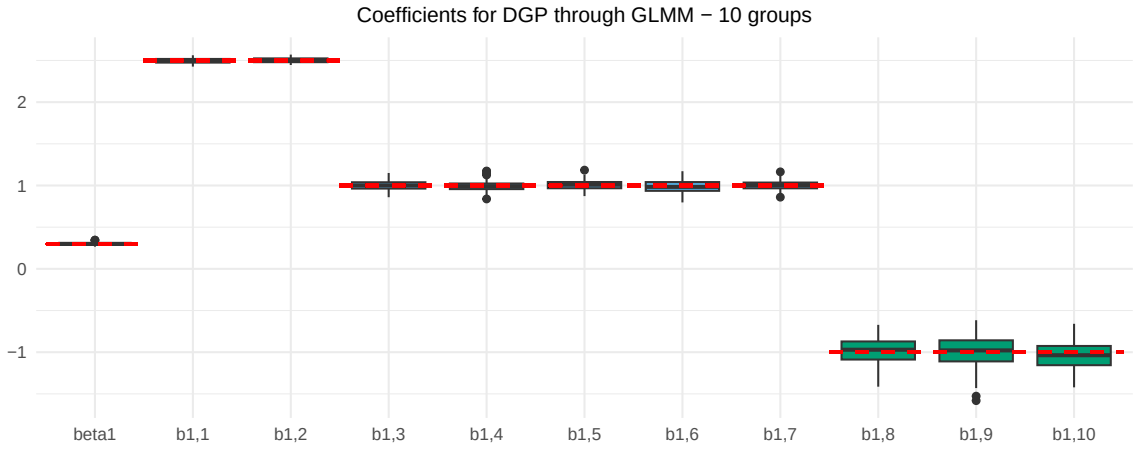
In this work, a novel statistical significance-based approach for implementing semi-parametric generalized linear mixed models with discrete random effects is proposed. When dealing with hierarchical data, the so-called α-*criterion* induces a clustering of the groups in which observations are nested, having no priors on the number of clusters to be identified. This approach results to be suitable and extremely useful when dealing with a high number of groups and reducing the dimensionality by identifying a latent structure at the group level is of interest. By setting a confidence level α, only clusters that are statistically different are identified within the iterations of a tailored EM algorithm.

This work enters into the literature about mixed-effects models with discrete random effects in which the state-of-the-art collapsing criterion of clusters was based on a discretional threshold - the *t-criterion*. This approach is suitable if the user knows a priori the distance to be observed between clusters or if a huge computational effort is put into running the algorithm under different thresholds in order to identify either the sought number of clusters or an elbow on the average entropy. Even in these cases, the *t-criterion* approach does not provide any insight about the statistical significance of the difference between the identified clusters. Besides these advantages, the simulation study in Section 4 shows that the α-*criterion* performs better than the *t-criterion* in most of the cases, also when the choice of the threshold is driven.

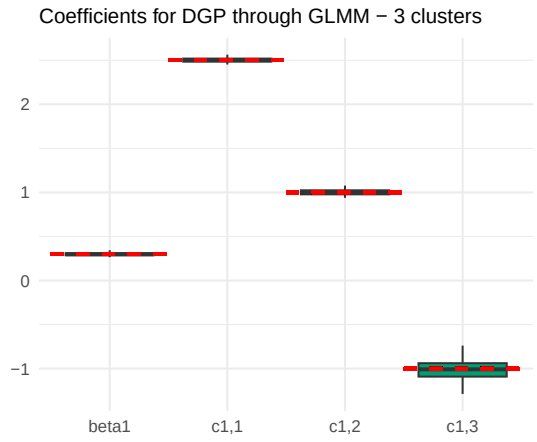
A further contribution of this work regards the generalization of mixed-effects models with discrete random effects to responses with law in the exponential family, in particular, binary and Poisson. This allows to enlarge the families of response variables that this type of models can handle, enriching the potential areas of applications. When tested on real data, the proposed methodology achieves a good prediction performance holding on the

Figure 8: Boxplots representing the distribution of the fixed slope β_1 and random intercepts for DGP for Poisson response, for 100 iterations of the DGP.

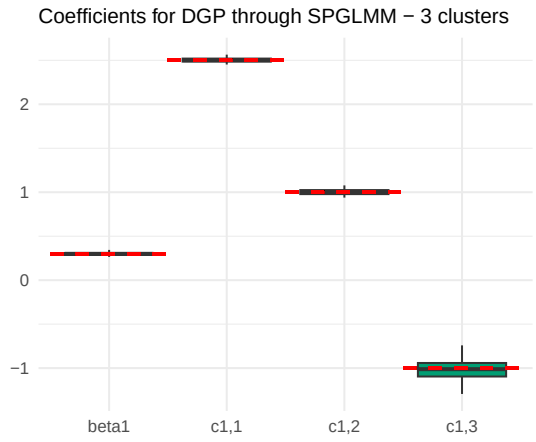
(a) GLMM (10 groups)



(b) GLMM (3 clusters)



(c) SPGLMM (3 clusters)



Notes: Random intercepts are denoted by b_{1i} for $i = 1, ..., 10$ in panel (a) and c_{1m} for $m = 1, 2, 3$ in panels (b) and (c). The horizontal dotted lines indicate the simulated coefficients.

comparison with the parametric version of the model, still performing the extra task of clustering the groups.

Some limitations of the work and, consequently, possible future directions regard the parameters initialization procedure and the definition of a clear objective function. Being the maximization step of the EM algorithm based on numerical approximations, the range of the parameters in which numerical methods look for the maximum plays a determinant role. Fitting a GLM within each group requires balanced responses within each group. Further research should be dedicated to the identification of a more straightforward and flexible initialization procedure. Regarding the objective function, we prove the increasing property of the likelihood when M is fixed, but we do not define an objective function

against which to assess whether the identified latent structure is the optimum one. In this perspective, the definition of an objective function that takes into account the likelihood and the model complexity would serve the purpose.

The analysis of illiteracy rates constitutes a simple and interpretable case study, but the method is applicable to different contexts. The identification of clusters of groups might be useful in the healthcare context to discover latent structures of patients with different levels of vulnerability standing on their repeated measurements or in the efficiency analysis context to evaluate different types of providers standing on the performance of their consumers or, again, in any situation in which dealing with clusters of groups is preferable that dealing with groups themselves.

A Loglikelihood for generalized responses

We here express the loglikelihood in Eq. (3) for the two special cases of Bernoulli and Poisson distributions, exploiting their canonical link function.

A.1 Bernoulli case

Bernoulli distribution models binary response variables. Given the random effects $\mathbf{b}_i$, the binary responses $\mathbf{y}_1, ..., \mathbf{y}_N$ are conditionally independent such that $\mathbf{y}_i|\mathbf{b}_i \sim \text{Be}(\boldsymbol{\mu}_i)$ for $i = 1, ..., N$, where $\text{logit}(\boldsymbol{\mu}_i) = \ln\left(\frac{\boldsymbol{\mu}_i}{1-\boldsymbol{\mu}_i}\right) = \boldsymbol{\eta}_i = \mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\boldsymbol{b}_i$. Then $p(y_{ij}|\boldsymbol{\beta}, \mathbf{b}_i) = \frac{\exp(y_{ij}\ \eta_{ij})}{1+\exp(\eta_{ij})}$ and the marginal loglikelihood is $\ln \mathcal{L}(\boldsymbol{\beta}, \mathbf{c}_1, ..., \mathbf{c}_M|\mathbf{y}) = \sum_{m=1}^{M} \omega_m \sum_{i=1}^{N} \sum_{j=1}^{n_i} y_{ij}\ \eta_{ijm} - \ln(\ 1 + \exp(\eta_{ijm})\ )$.

A.2 Poisson case

Poisson distribution models counts as outcomes. Given the random effects $\mathbf{b}_i$, the counts $\mathbf{y}_1, ..., \mathbf{y}_N$ are conditionally independent such that $\mathbf{y}_i|\mathbf{b}_i \sim \text{Poi}(\boldsymbol{\mu}_i)$ for $i = 1, ..., N$, where $\ln(\boldsymbol{\mu}_i) = \boldsymbol{\eta}_i = \mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\boldsymbol{b}_i$. Then $p(y_{ij}|\boldsymbol{\beta}, \mathbf{b}_i) = \frac{\exp(y_{ij}\ \eta_{ij})\ \exp(\ -\exp(\eta_{ij})\ )}{y_{ij}!}$ and the marginal loglikelihood is $\ln \mathcal{L}(\boldsymbol{\beta}, \mathbf{c}_1, ..., \mathbf{c}_M|\mathbf{y}) = \sum_{m=1}^{M} \omega_m \sum_{i=1}^{N} \sum_{j=1}^{n_i} y_{ij}\ \eta_{ijm} - \exp(\eta_{ijm}) - \ln(y_{ij}!)$.

Supplementary Materials

S1 Proof of the increasing likelihood property

Following the setting presented in Azzimonti et al. (2013) and Masci et al. (2022), we prove the increasing likelihood property of the EM algorithm for SPGLMM, given a fixed number of clusters M. We want to prove that $\mathcal{L}(\boldsymbol{\beta}^{(up)}, \mathbf{y}) \geq \mathcal{L}(\boldsymbol{\beta}, \mathbf{y})$ where $\boldsymbol{\beta}^{(up)}$ is the updated fixed effect. From the marginal likelihood definition, we get:

$$\ln\left[\frac{\mathcal{L}(\boldsymbol{\beta}^{(up)}|\mathbf{y})}{\mathcal{L}(\boldsymbol{\beta}|\mathbf{y})}\right] = \sum_{i=1}^{N} \ln\left[\frac{p(\mathbf{y}_i|\boldsymbol{\beta}^{(up)})}{p(\mathbf{y}_i|\boldsymbol{\beta})}\right] \tag{S1.1}$$

so that

$$\ln\left[\frac{p(\mathbf{y}_i|\boldsymbol{\beta}^{(up)})}{p(\mathbf{y}_i|\boldsymbol{\beta})}\right] = Q_i(\boldsymbol{\theta}^{(up)}, \boldsymbol{\theta}) - Q_i(\boldsymbol{\theta}, \boldsymbol{\theta})$$

Thanks to the convexity of the logarithm function and some algebraic passages, we obtain:

$$\begin{aligned}
\ln\left[\frac{p(\mathbf{y}_i|\boldsymbol{\beta}^{(up)})}{p(\mathbf{y}_i|\boldsymbol{\beta})}\right] &= \ln \sum_{m=1}^{M} \frac{\omega_m^{(up)}\, p(\mathbf{y}_i|\boldsymbol{\beta}^{(up)}, \mathbf{c}_m^{(up)})}{p(\mathbf{y}_i|\boldsymbol{\beta})} \\
&= \ln \sum_{m=1}^{M} \left(\frac{\omega_m\, p(\mathbf{y}_i|\boldsymbol{\beta}, \mathbf{c}_m)}{p(\mathbf{y}_i|\boldsymbol{\beta})}\right)\left(\frac{\omega_m^{(up)}\, p(\mathbf{y}_i|\boldsymbol{\beta}^{(up)}, \mathbf{c}_m^{(up)})}{\omega_m\, p(\mathbf{y}_i|\boldsymbol{\beta}, \mathbf{c}_m)}\right) \\
&\geq \sum_{m=1}^{M} \left(\frac{\omega_m\, p(\mathbf{y}_i|\boldsymbol{\beta}, \mathbf{c}_m)}{p(\mathbf{y}_i|\boldsymbol{\beta})}\right) \ln\left(\frac{\omega_m^{(up)}\, p(\mathbf{y}_i|\boldsymbol{\beta}^{(up)}, \mathbf{c}_m^{(up)})}{\omega_m\, p(\mathbf{y}_i|\boldsymbol{\beta}, \mathbf{c}_m)}\right) \\
&= \mathcal{Q}_i(\boldsymbol{\theta}^{(up)}, \boldsymbol{\theta}) - \mathcal{Q}_i(\boldsymbol{\theta}, \boldsymbol{\theta})
\end{aligned} \tag{S1.2}$$

where

$$\mathcal{Q}_i(\boldsymbol{\theta}^{(up)}, \boldsymbol{\theta}) = \sum_{m=1}^{M} \left(\frac{\omega_m\, p(\mathbf{y}_i|\boldsymbol{\beta}, \mathbf{c}_m)}{p(\mathbf{y}_i|\boldsymbol{\beta})}\right) \ln\left(\omega_m^{(up)}\, p(\mathbf{y}_i|\boldsymbol{\beta}^{(up)}, \mathbf{c}_m^{(up)})\right)$$

and

$$\mathcal{Q}_i(\boldsymbol{\theta}, \boldsymbol{\theta}) = \sum_{m=1}^{M} \left(\frac{\omega_m\, p(\mathbf{y}_i|\boldsymbol{\beta}, \mathbf{c}_m)}{p(\mathbf{y}_i|\boldsymbol{\beta})}\right) \ln\left(\omega_m\, p(\mathbf{y}_i|\boldsymbol{\beta}, \mathbf{c}_m)\right) .$$

Defining $\mathcal{Q}(\boldsymbol{\theta}^{(up)}, \boldsymbol{\theta}) = \sum_{i=1}^{N} \mathcal{Q}_i(\boldsymbol{\theta}^{(up)}, \boldsymbol{\theta})$ and $\mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}) = \sum_{i=1}^{N} \mathcal{Q}_i(\boldsymbol{\theta}, \boldsymbol{\theta})$, we get a lower bound for the quantity of interest thanks to Eqs. (S1.1) and (S1.2):

$$\ln\left[\frac{p(\mathbf{y}_i|\boldsymbol{\beta}^{(up)})}{p(\mathbf{y}_i|\boldsymbol{\beta})}\right] \geq \mathcal{Q}(\boldsymbol{\theta}^{(up)}, \boldsymbol{\theta}) - \mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}) .$$

We must now show that $\mathcal{Q}(\boldsymbol{\theta}^{(up)}, \boldsymbol{\theta}) \geq \mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta})$. This can be proved by defining $\boldsymbol{\theta}^{(up)}$ as

$$\boldsymbol{\theta}^{(up)} = \arg\max_{\tilde{\boldsymbol{\theta}}} \mathcal{Q}(\tilde{\boldsymbol{\theta}}, \boldsymbol{\theta}) \quad \forall\, \boldsymbol{\theta} \;\text{ fixed.}$$

Defining W_{im} as in Eq. (5), we get

$$\begin{aligned}
\mathcal{Q}(\tilde{\boldsymbol{\theta}}, \boldsymbol{\theta}) &= \sum_{i=1}^{N}\sum_{m=1}^{M}\left(\frac{\omega_m\, p(\mathbf{y}_i|\boldsymbol{\beta}, \mathbf{c}_m)}{\sum_{k=1}^{M}\omega_k\, p(\mathbf{y}_i|\boldsymbol{\beta}, \mathbf{c}_k)}\right)\ln\big(\tilde{\omega}_m\, p(\mathbf{y}_i|\tilde{\boldsymbol{\beta}}, \tilde{\mathbf{c}}_m)\big) \\
&= \sum_{i=1}^{N}\sum_{m=1}^{M} W_{im}\,\ln\big(\tilde{\omega}_m\, p(\mathbf{y}_i|\tilde{\boldsymbol{\beta}}, \tilde{\mathbf{c}}_m)\big) \\
&= \sum_{i=1}^{N}\sum_{m=1}^{M} W_{im}\,\ln\tilde{\omega}_m + \sum_{i=1}^{N}\sum_{m=1}^{M} W_{im}\, p(\mathbf{y}_i|\tilde{\boldsymbol{\beta}}, \tilde{\mathbf{c}}_m) \\
&= \mathcal{J}_1(\tilde{\omega}_1, ..., \tilde{\omega}_M) + \mathcal{J}_2(\tilde{\boldsymbol{\beta}}, \tilde{\mathbf{c}}_1, ..., \tilde{\mathbf{c}}_M)
\end{aligned}$$

The functionals $\mathcal{J}_1$ and $\mathcal{J}_2$ can be maximized separately. Eq. (4) is obtained by maximinzing $\mathcal{J}_1$ in closed form. More specifically, the functional $\mathcal{J}_1$ could be rewritten as follows:

$$\begin{aligned}
\mathcal{J}_1(\tilde{\omega}_1, ..., \tilde{\omega}_M) &= \sum_{m=1}^{M-1}\sum_{i=1}^{N} W_{im}\,\ln\tilde{\omega}_m + \sum_{i=1}^{N} W_{iM}\,\ln\tilde{\omega}_M \\
&= \sum_{m=1}^{M-1}\sum_{i=1}^{N} W_{im}\,\ln\tilde{\omega}_m + \sum_{i=1}^{N} W_{iM}\,\ln\left(1 - \sum_{m=1}^{M-1}\tilde{\omega}_m\right)
\end{aligned}$$

and, imposing the gradient equal to zero, we obtain:

$$\frac{\partial\mathcal{J}_1}{\partial\tilde{\omega}_m} = \frac{\sum_{i=1}^{N} W_{im}}{\tilde{\omega}_m} - \frac{\sum_{i=1}^{N} W_{iM}}{1 - \sum_{m=1}^{M-1}\tilde{\omega}_m} = \frac{\sum_{i=1}^{N} W_{im}}{\tilde{\omega}_m} - \frac{\sum_{i=1}^{N} W_{iM}}{\tilde{\omega}_M} = 0 \quad \forall\, m = 1, ..., M-1$$

that is equivalent to $\frac{\sum_{i=1}^{N} W_{im}}{\tilde{\omega}_m} = \frac{\sum_{i=1}^{N} W_{ik}}{\tilde{\omega}_k} \quad \forall\, m, k = 1, ..., M$. Summing on $k = 1, ..., M$, since $\sum_{m=1}^{M} W_{im} = 1$, we obtain Eq. (4).

On the other hand, the update in Eq. (6) for $\boldsymbol{\beta}$ and $(\mathbf{c}_1, ..., \mathbf{c}_M)$ is obtained maximizing the functional $\mathcal{J}_2$ through numerical approximations.

S2 Methodology: further details

S2.1 Parameters initialization

SPGLMM starts by considering N discrete masses and iteratively reduces the number by grouping them into $M < N$ clusters. However, like most clustering algorithms, SPGLMM

is highly sensitive to the initial placement of the starting points. To avoid any biases in the clustering procedure, it is important to initialize the N discrete masses in a robust way. To achieve this, we fit a GLM within each group i, for $i = 1, ..., N$ and obtain N distinct models. We then extract the Q parameters distributions, composed of the estimates of the intercept and slopes of the N models. The $M = N$ starting values of the h^{th} random coefficient, for $h = 1, ..., Q$, are obtained by computing the first and third quartiles $q_{0.25,h}$ and $q_{0.75,h}$ from the N-dimensional distribution composed by the coefficients in each of the N fitted models corresponding to the h^{th} random coefficient. Inspired by the boxplot whiskers definition, we construct the interval $r_h = [r_{min,\,h} \;\; ; \;\; r_{max,\,h}] = [q_{0.25,\,h} - 1.5 \cdot IQR_h \;\; ; \;\; q_{0.75,\,h} + 1.5 \cdot IQR_h]$, where IQR_h stands for the interquartile range $q_{0.75,\,h} - q_{0.25,\,h}$. From the interval r_h, we then randomly select $M = N$ support points as follows: $r_{min,\,h} + (r_{max,\,h} - r_{min,\,h}) \cdot \mathcal{U}(0, 1)$, where $\mathcal{U}(0, 1)$ stands for the uniform distribution between 0 and 1. At the initial step, the weights $\hat{\omega}_1^{(0)}, ..., \hat{\omega}_M^{(0)}$ are uniformly distributed on the $M = N$ support points: each of them is initialized at $1/N$.

Following the method proposed in Masci et al. (2019), if the number of groups N is extremely large (e.g., $N > 100$) the algorithm could be slowed down. For this reason, in the computation of $\hat{\mathbf{c}}_h^{(0)}$, $h = 1, ..., Q$, we can alternatively fix a boundary $\tilde{N}$ and randomly select $M = \tilde{N} < N$ (instead of N) support points within the r_h interval and uniformly distribute the weights on these $\tilde{N}$ support points. The fixed coefficients are also initialized by exploiting the N distinct fitted models. Specifically, the starting value of the p^{th} fixed coefficient $\hat{\beta}_p^{(0)}$, for $p = 1, ..., P$, is assigned to the median of the N-dimensional distribution composed by the p^{th} fixed coefficients.

S2.2 Beyond the support reduction: a check on the weights

Beyond the support point reduction, a check on the weights is performed. A sketch of the pseudo-code related to this check is addressed in Algorithm 2. Specifically, at each iteration k, if the m^{th} cluster has weight $\hat{\omega}_m^{(k)}$ equal to zero (i.e., the m$^{\text{th}}$ column of $\hat{\mathbf{W}}^{(k)}$ contains all zero elements), the cluster m^{th} is removed and the total number of clusters is updated accordingly. Moreover, the remaining weights are then renormalized in such a way that they sum up to 1, as follows:

$$\hat{\omega}_m^{new} = \frac{\hat{\omega}_m^{old}}{S_{\hat{\omega}}} \; \forall m = 1, ..., M_{new} \text{ where } S_{\hat{\omega}} = \sum_{m=1}^{M_{new}} \hat{\omega}_m^{old} \; . \tag{S2.1}$$

In addition, when convergence is reached (see Section S2.3 for the convergence conditions), we remove the empty clusters, i.e., the support points to which no groups are associated (see $\tilde{l}_i$ in Eq. (9) for the association between groups and clusters). The remaining weights are then renormalized as in Eq. (S2.1). If no mass points are deleted, the

algorithm can terminate; otherwise, at least another iteration is required, in order to make the algorithm update the mass points.

Pseudo-code 2: Support reduction: final check on weights

function `Check weights(`$conv1, M, N,$ `K1,`$k,\ \hat{\mathbf{W}}^{(k)}, (\hat{\mathbf{c}}_1^{(k)}, ..., \hat{\mathbf{c}}_M^{(k)}), (\hat{\omega}_1^{(k)}, ..., \hat{\omega}_M^{(k)})$ `)`

- **if** `M is not 1` **then** ▷ we delete clusters with $\hat{\omega}_m^{(k)} = 0$
  - Delete null columns of $\hat{\mathbf{W}}^{(k)}$ (if any)
  - $M \leftarrow M-$ number of deleted columns (if any)
  - Reparametrize weights as in Eq. (9)
- **if** $conv1$ `is 1 or` $k \geq$ `K1` **then**
  - **for** `i=1,...,N` **do**
    - $\tilde{l}_i \leftarrow \tilde{l}_i$ as in Eq. (9)
  - $flag \leftarrow 1$
  - **for** `m=1,...,M` **do** ▷ we delete empty clusters
    - **if** `None of` $[\tilde{l}_1, ..., \tilde{l}_N]$ `is equal to m` **then**
      - Delete the mass point $\hat{\mathbf{c}}_m^{(k)}$ satisfying such condition
      - $M \leftarrow M - 1$
      - Reparametrize weights as in Eq. (S2.1)
      - $flag \leftarrow 0$
      - $conv1 \leftarrow 0$
  - **if** $flag$ `is 1` **then**
    - $conv2 \leftarrow 1$
- **return** M, $(\hat{\mathbf{c}}_1^{(k)}, ..., \hat{\mathbf{c}}_M^{(k)})$, $(\hat{\omega}_1^{(k)}, ..., \hat{\omega}_M^{(k)})$, $conv2$, $conv1$

S2.3 Convergence criteria

At each iteration k of the SPGLMM, the updated number of mass points M is estimated: the EM algorithm computes the updates of both fixed and random effects within each it^{th} *sub*-iteration of k, until either a maximum number of *a priori* fixed iterations `itmax` is reached (worst case of not convergence) or all the differences of fixed and random parameters estimates at two consecutive iterations it and $it-1$ are smaller than fixed tolerance values `tF` and `tR`, respectively (i.e., if $|\hat{c}_{mh}^{(it)} - \hat{c}_{mh}^{(it-1)}| <$ `tR` $\forall m = 1, ..., M,\ \forall h = 1, ..., Q$ and $|\hat{\beta}_p^{(it)} - \hat{\beta}_p^{(it-1)}| <$ `tF` $\forall p = 1, ..., P$).

The support points reduction based on α-*criterion* starts after the first `K2` iterations since before the estimates get stabilized. When all the differences between the estimates of the parameters at two consecutive iterations k and $k-1$ are smaller than fixed tolerance values (i.e. if $|\hat{c}_{mq}^{(k)} - \hat{c}_{mq}^{(k-1)}| <$ `tR` $\forall m = 1, ..., M,\ \forall h = 1, ..., Q$ and $|\hat{\beta}_p^{(k)} - \hat{\beta}_p^{(k-1)}| <$ `tF` $\forall p = 1, ..., P$) and no more confidence regions overlap, the value of the dummy variable `conv1` switches from 0 to 1. When convergence is reached (i.e., `conv1` is 1 or after a given number of iterations `K1`), the empty clusters, if present, are removed (see Section S2.2 and Algorithm 2). The algorithm stops when both `conv1` is 1 and no more empty clusters

are present, or, in the case of no convergence, when $k \geq \texttt{K}$, where K is an *a priori* fixed threshold.

S3 Fast Ellipsoid Intersection Test

Consider the scalar function $K : (0,1) \to \mathbb{R}$

$$K(s) := 1 - (\hat{\mathbf{c}}_l^{(k)} - \hat{\mathbf{c}}_m^{(k)})^{\mathrm{T}} \left(\frac{1}{1-s} \mathrm{var}(\hat{\mathbf{c}}_m^{(k)})^{-1} + \frac{1}{s} \mathrm{var}(\hat{\mathbf{c}}_l^{(k)})^{-1} \right) (\hat{\mathbf{c}}_l^{(k)} - \hat{\mathbf{c}}_m^{(k)})$$

where $^{\mathrm{T}}$ stands for the transpose.

The Fast Ellipsoid Intersection Test states that $CR_{1-\alpha}(\hat{\mathbf{c}}_l^{(k)})$ intersects $CR_{1-\alpha}(\hat{\mathbf{c}}_m^{(k)})$ if and only if $K(s) \geq 0 \; \forall s \in (0,1)$, as proven in Proposition 2 of Gilitschenski and Hanebeck (2012). Therefore, through fast one-dimensional optimization methods, we can minimize $K(s)$ on $(0,1)$ and check the sign at minimizing point s^*. One of the following three cases applies: (i) if $K(s^*) > 0$, the ellipsoids intersect; (ii) if $K(s^*) < 0$ they do not intersect while (iii) if $K(s^*) = 0$ the ellipsoids touch their boundaries.

The computation of $K(s)$ can be simplified exploiting the generalized eigenvalues λ_h and generalized eigenvectors ϕ_h of the generalized eigenvalue problem (Parlett (1998)), for $h = 1, ..., Q$, defined by $\mathrm{var}(\hat{\mathbf{c}}_m^{(k)})\ \phi_h = \lambda_h\ \mathrm{var}(\hat{\mathbf{c}}_l^{(k)})\ \phi_h$ such that $\phi_h^{\mathrm{T}}\ \mathrm{var}(\hat{\mathbf{c}}_l^{(k)})\ \phi_h = 1$. In fact, it is well established that $\mathbf{\Phi}^{\mathrm{T}}\ \mathrm{var}(\hat{\mathbf{c}}_m^{(k)})\ \mathbf{\Phi} = \mathbf{\Lambda}$ and $\mathbf{\Phi}^{\mathrm{T}}\ \mathrm{var}(\hat{\mathbf{c}}_l^{(k)})\ \mathbf{\Phi} = \mathbf{I}$, where $\mathbf{\Phi}$ is the $Q \times Q$ matrix whose h^{th} column is the vector ϕ_h, $\mathbf{\Lambda}$ is the $Q \times Q$ diagonal matrix whose h^{th} diagonal entry is λ_h and $\mathbf{I}$ is the $Q \times Q$ identity matrix. Since $\mathrm{var}(\hat{\mathbf{c}}_m^{(k)})$ and $\mathrm{var}(\hat{\mathbf{c}}_l^{(k)})$ in the basis of generalized eigenvectors are diagonal, through algebraic manipulation, we can rewrite $K(s)$ as follows:

$$K(s) := 1 - \frac{1}{\chi^2_Q(1-\alpha)} \sum_{h=1}^{Q} v_h^2 \frac{s(1-s)}{1 + s(\lambda_h - 1)} \tag{S3.1}$$

where $v_h := \mathbf{\Phi}^{\mathrm{T}}(\hat{\mathbf{c}}_{mh}^{(k)} - \hat{\mathbf{c}}_{lh}^{(k)})$.

S4 Case study results for the Bernoulli response

For the case of a Bernoulli response, SPGLMM aims to predict the presence of low-achieving students identifying clusters of countries. Starting from the data pre-processed as in Section 3.1, we create the variable Bernoulli-distributed Y_BIN_MATH, which assumes value 1 if Y_MATH is strictly greater than 2%, 0 otherwise. 2% is chosen because it is the minimum threshold ensuring that in each country (the groups) there are both 0s and 1s. We get 6125 schools with class 1 ($> 2\%$ of low-achieving students) and 6495 schools with class

0 ($\leq$ 2% of low-achieving students). By assuming the model formulations described in Section 3.2, we now consider the response y as `Y_BIN_MATH` and the canonical link function as $g(\cdot) = \text{logit}(\cdot)$.

Similarly to the Poisson case, results for Bernoulli response are addressed in Table S4.1. We get $\hat{M}_{0.05} = \hat{M}_{0.10} = 10$ and $\hat{M}_{0.01} = 8$. By comparing the SPGLMM estimates with a parametric GLMM as explained in Section 3.2, we appreciate that the means of the random intercepts b_i in each cluster are slightly lower in absolute value than the estimates for SPGLMM. Anyhow, we observe huge coherence between the two models. In addition, the random intercept obtained for $\alpha = 0.01$ in correspondence of the clusters 1 and 2 - that we denote with $\hat{c}_{1+2}$ for simplicity - turns out to be a weighted mean between $\hat{c}_1$ and $\hat{c}_2$; the same happens for $\hat{c}_{7+8}$ (i.e., the random intercept obtained for $\alpha = 0.01$ in correspondence of the clusters 7 and 8). Moreover, given the negative sign of β_1, we can observe that the school size is inversely proportional to the probability of the presence of more than 2% of low-achieving students: the higher the value of `SCHSIZE`, the lower the estimated probability. Similarly, the higher the value of `avg_ESCS_std`, the lower the estimated probability (see the negative sign of β_2), even if `avg_ESCS_std` has a much lower impact than `SCHSIZE` (the former slope is $\beta_2 = -0.62$ compared to the latter one of $\beta_1 = -3.26$). In general, we can conclude that also for the fixed slopes, SPGLMM and GLMM provide coherent results in the estimates, the standard errors and the p-values (estimated through likelihood-ratio test).

In Figure S4.1, both in panels (a) and (b), we display the caterpillar plot for the random intercepts (together with their confidence intervals) of the 50 countries, obtained through parametric GLMM. On each of the two panels, we highlight the identified clusters of countries, both for $\alpha = 0.01$ in panel (a) and $\alpha = 0.05, 0.10$ in panel (b). Results can be interpreted as follows: the lower the estimated random intercept for a cluster, the less likely (with respect to the average) the presence of at least 2% of low-achieving students in mathematics in the schools of the countries of that cluster.

For better visualisation of the clusters of countries identified by the SPGLMM, we highlight with the same shade of grey on the map in Figure S4.2 the countries identified by the same random intercept (i.e. the countries in the same cluster), for $\alpha = 0.01$. We notice that B-S-J-Z (China) and Australia, net of the other features, decrease the probability of innumeracy presence (more specifically, decrease the probability of having at least 2% of low-achieving students in mathematics). After them, the European countries decrease the probability with less impact, while in the Americas the probability of innumeracy presence increases.

For the Goodness of Fit (GoF) evaluation, we use the same data we used to train the models since our main purpose is to compare on equal terms the predictive power of SPGLMM and GLMM. In Figure S4.3 the Receiver Operating Characteristic (ROC) curve

Table S4.1: SPGLMM estimates for Bernoulli response and comparison with GLMM output.

Coeff. estimates		**SPGLMM**			**GLMM**
		$\alpha = 0.01$	$\alpha = 0.05$	$\alpha = 0.10$	
$\hat{\mathbf{c}}$ [$\hat{\omega}$]	$\hat{c}_1$ [0.02]	-2.747 (0.078)	-3.533 (0.320)	-3.533 (0.320)	-3.415
	$\hat{c}_2$ [0.10]		-2.679 (0.080)	-2.680 (0.080)	-2.635
	$\hat{c}_3$ [0.12]	-2.091 (0.069)	-2.088 (0.070)	-2.088 (0.070)	-2.029
	$\hat{c}_4$ [0.08]	-1.591 (0.078)	-1.590 (0.078)	-1.591 (0.078)	-1.562
	$\hat{c}_5$ [0.14]	-0.984 (0.054)	-0.984 (0.054)	-0.984 (0.054	-0.954
	$\hat{c}_6$ [0.04]	-0.536 (0.103)	-0.563 (0.106)	-0.564 (0.106)	-0.470
	$\hat{c}_7$ [0.06]	0.318 (0.059)	0.022 (0.105)	0.022 (0.105)	-0.020
	$\hat{c}_8$ [0.18]		0.437 (0.070)	0.437 (0.069)	0.448
	$\hat{c}_9$ [0.14]	1.239 (0.075)	1.245 (0.076)	1.245 (0.076)	1.162
	$\hat{c}_{10}$ [0.12]	2.237 (0.137)	2.237 (0.137)	2.237 (0.137)	2.057
$\hat{\boldsymbol{\beta}}$	$\hat{\beta}_1$	-3.262 (0.052) ***	-3.261 (0.052) ***	-3.261 (0.052) ***	-3.223 (0.073) ***
	$\hat{\beta}_2$	-0.618 (0.028) ***	-0.620 (0.028) ***	-0.620 (0.028) ***	-0.623 (0.029) ***

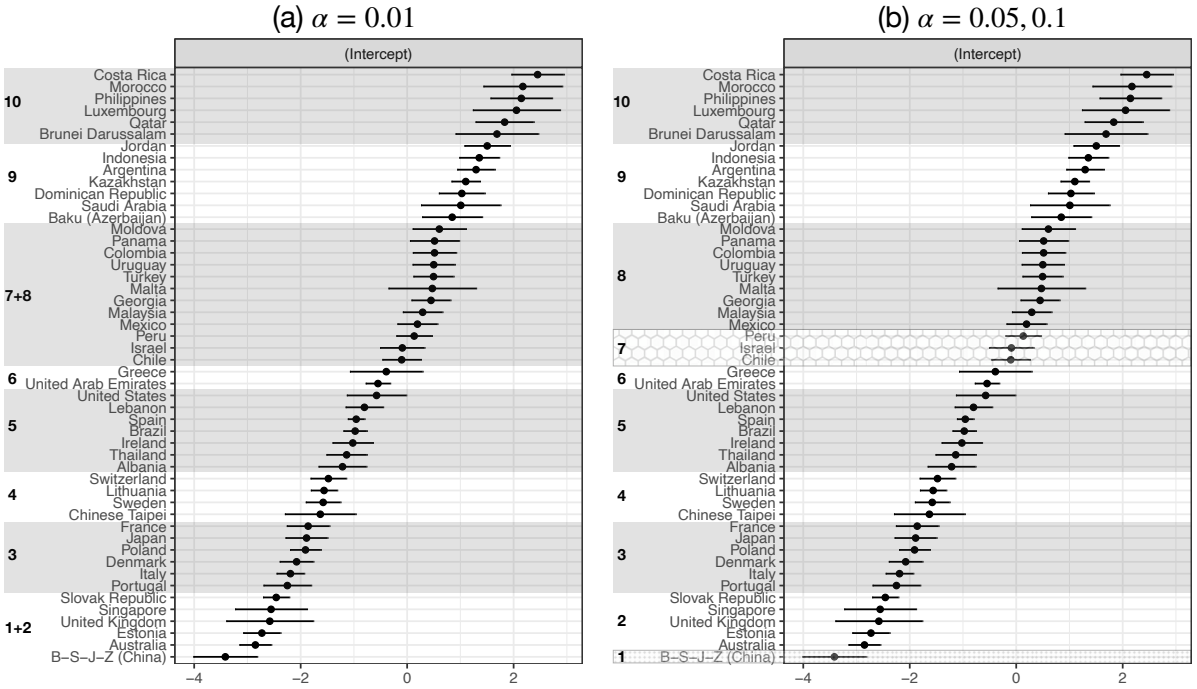
Notes: The estimated random intercepts $\hat{\mathbf{c}}$ are presented in increasing order, together with their respective weights $\hat{\omega}$ in brackets, as well as the fixed effects $\hat{\boldsymbol{\beta}}$ for both SPGLMM (with $\alpha = 0.01, 0.05, 0.10$) and GLMM (for each row of $\hat{c}_m$, the average of the $\hat{b}_i$s in each cluster m is reported). In parenthesis, the standard error is computed by square rooting the inverse of the Fisher Information Matrix. For $\hat{\boldsymbol{\beta}}$, the p-value is estimated by means of likelihood-ratio test (* p-value < 0.1; ** p-value < 0.01; *** p-value < 0.001).

is displayed both for the SPGLMM (with $\alpha = 0.05$) and GLMM. Moreover, in Table S4.2 we report for each case of the SPGLMM and for the GLMM, the Area Under Curve (AUC), the optimal identified threshold and the Sensitivity, Specificity and Accuracy computed by assuming the chosen threshold. The two methods reveal similar predictive performances. Both the ROC curves and the results in the table drive us to the conclusion that the SPGLMM with Bernoulli response does not underperform with respect to the parametric classical GLMM. Furthermore, it also provides as output a clustering of the hierarchies (i.e., the countries in our case study), revealing the inner structure the model assumes.

Table S4.2: GoF metrics estimates for the case study with Bernoulli response, fitted via SPGLMM and GLMM.

	SPGLMM			**GLMM**
	$\alpha = 0.01$	$\alpha = 0.05$	$\alpha = 0.10$	
AUC	0.9158	0.9162	0.9162	0.9165
Chosen threshold	0.4435	0.4397	0.4397	0.4523
Sensitivity	0.8143	0.8128	0.8127	0.8171
Specificity	0.8851	0.8869	0.8869	0.8807
Accuracy	0.8476	0.8475	0.8475	0.8473

Figure S4.1: Caterpillar plots representing the comparison between the 50 random intercepts estimated by GLMM and the clusters obtained through SPGLMM for $\alpha = 0.01$ and 0.05, 0.10 with Bernoulli response.



Notes: To ease the comparison with Table S4.1, the same table colours are used in panel (a) to highlight the clusters of countries. In panel (b), clusters 1 and 7 are highlighted with different texture.

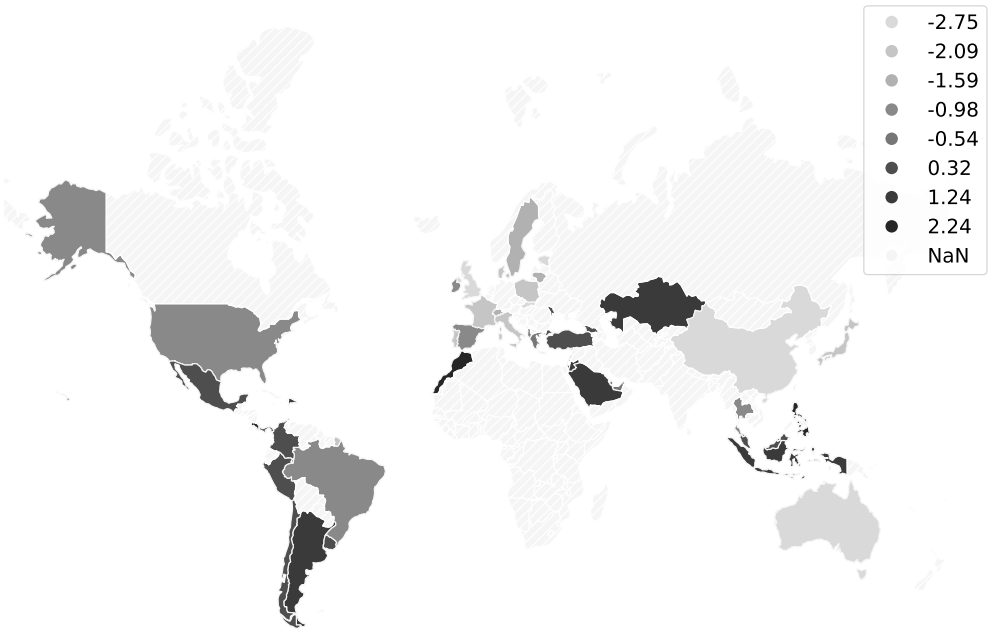
S5 Simulation study for the Bernoulli response

Starting from the simulation study set-up described in Section 4.1, we simulate three different model types, which in turn present: (i) a random intercept; (ii) a random slope; (ii) both a random intercept and a random slope. For each model, we set up the cases with both one and two fixed slopes (the second fixed slope will be indicated in parenthesis in the following equations), obtaining six different models. The linear predictor $\boldsymbol{\eta}_i$ is defined by the following Data Generating Processes (DGPs):

(i) Random intercept case ($\boldsymbol{\eta}_i = \boldsymbol{\beta}_1 \mathbf{x}_{1i} \ + (\boldsymbol{\beta}_2 \mathbf{x}_{2i}) + c_{1i} \mathbb{1}_{n_i}$)

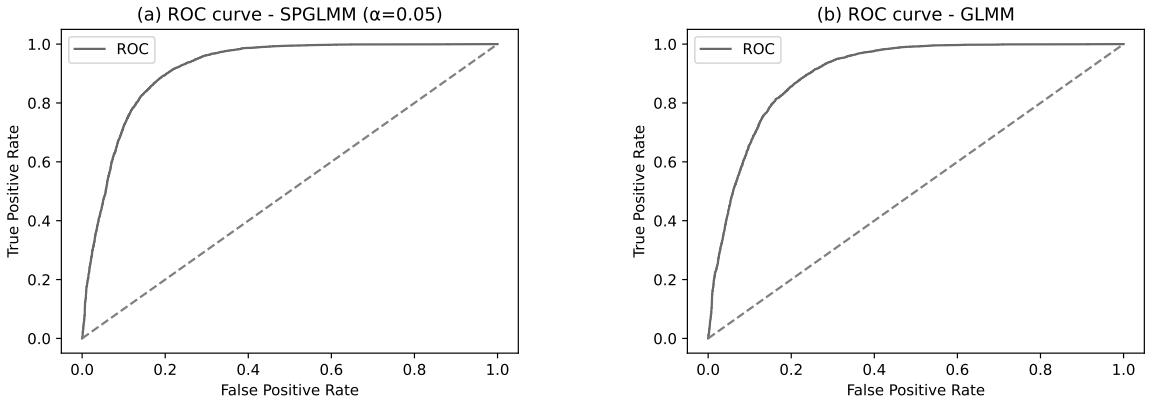
$$\boldsymbol{\eta}_i = \begin{cases} -6\mathbf{x}_{1i} \ + (3\mathbf{x}_{2i}) + 5 \ \mathbb{1}_{n_i} & \text{if } i = 1, 2, \\ -6\mathbf{x}_{1i} \ + (3\mathbf{x}_{2i}) + 2 \ \mathbb{1}_{n_i} & \text{if } i = 3, 4, 5, 6, 7, \\ -6\mathbf{x}_{1i} \ + (3\mathbf{x}_{2i}) - 10 \ \mathbb{1}_{n_i} & \text{if } i = 8, 9, 10 \end{cases} \tag{S5.1}$$

Figure S4.2: Choropleth map of the clusters of countries identified by the random intercepts in SPGLMM with $\alpha = 0.01$ for Bernoulli response.



Notes: Countries represented with the same colour belong to the same cluster. The lighter, the lower the random intercept. Light grey-striped countries are the ones for which the survey was not performed, or which were presenting missing values.

Figure S4.3: ROC curves for the **Bernoulli response**, for the SPGLMM ($\alpha = 0.05$) and GLMM.



(ii) Random slope case ($\boldsymbol{\eta}_i = \boldsymbol{\beta}_1 \mathbb{1}_{n_i} + \boldsymbol{\beta}_2 \mathbf{x}_{1i} \ + (\boldsymbol{\beta}_3 \mathbf{x}_{2i}) + c_{1i}\mathbf{z}_{1i}$)

$$\boldsymbol{\eta}_i = \begin{cases} +10\ \mathbb{1}_{n_i} - 6\mathbf{x}_{1i} \ + (3\mathbf{x}_{2i}) + 10\mathbf{z}_{1i} & \text{if } i = 1, 2, \\ +10\ \mathbb{1}_{n_i} - 6\mathbf{x}_{1i} \ + (3\mathbf{x}_{2i}) + 5\mathbf{z}_{1i} & \text{if } i = 3, 4, 5, 6, 7, \\ +10\ \mathbb{1}_{n_i} - 6\mathbf{x}_{1i} \ + (3\mathbf{x}_{2i}) + 0\mathbf{z}_{1i} & \text{if } i = 8, 9, 10 \end{cases} \tag{S5.2}$$

(iii) Random intercept and slope case ($\boldsymbol{\eta}_i = \boldsymbol{\beta}_1 \mathbf{x}_{1i} \ + (\boldsymbol{\beta}_2 \mathbf{x}_{2i}) + c_{1i}\mathbb{1}_{n_i} + c_{2i}\mathbf{z}_{1i}$)

$$\boldsymbol{\eta}_i = \begin{cases} -6\mathbf{x}_{1i} \ + (3\mathbf{x}_{2i}) + 5\ \mathbb{1}_{n_i} + 10\mathbf{z}_{1i} & \text{if } i = 1, 2, \\ -6\mathbf{x}_{1i} \ + (3\mathbf{x}_{2i}) + 2\ \mathbb{1}_{n_i} + 5\mathbf{z}_{1i} & \text{if } i = 3, 4, 5, 6, 7, \\ -6\mathbf{x}_{1i} \ + (3\mathbf{x}_{2i}) - 10\ \mathbb{1}_{n_i} + 0\mathbf{z}_{1i} & \text{if } i = 8, 9, 10 \end{cases} \tag{S5.3}$$

Variables $\mathbf{x}_{1i}$, $\mathbf{x}_{2i}$ and $\mathbf{z}_{1i}$ are normally distributed with mean equal to 0 and standard deviation equal to 1. The choice of the coefficients is driven by the need to simulate situations in which we obtain both balanced and unbalanced proportions of zeros and ones. In fact, after the computation of $\boldsymbol{\eta}_i$ according to each DGP, we retrieve μ_{ij} by the inverse of the link function $g^{-1}(\eta_{ij})$ and we compute y_{ij} for $i = 1, ..., N$ and $j = 1, ..., n_i$ as follows:

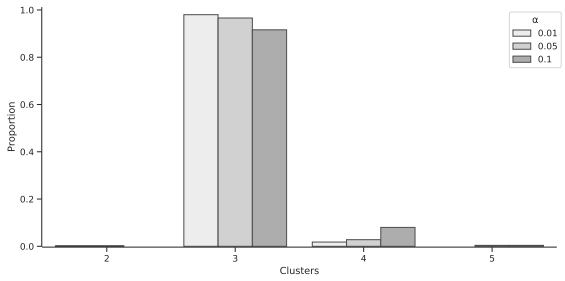
$$y_{ij} = \begin{cases} 0 & \text{if } u > \mu_{ij} \\ 1 & \text{otherwise} \end{cases} \tag{S5.4}$$

where u is randomly extracted from $\mathcal{U}[0, 1]$. Afterwards, we apply SPGLMM with both t- and α-*criterion*s, performing 500 runs for each of the six settings shown in Eqs. (S5.1-S5.3), for different values of t and α. Results obtained for DGPs (i), (ii) and (iii) with **one fixed slope** are reported respectively in Tables S7.3, S7.5 and S7.6 in Section S7. To further prove the generality of our results, we also report the DGP (i) with **two fixed slopes** in Table S7.4 in Section S7. In the tables, estimates of the proportion of identified clusters, entropy (refer to Section S6 for the definition), weights $\hat{\omega}$, random and fixed coefficients are reported in terms of mean and standard deviation (sd) across the 500 iterations. Moreover, we report results concerning DGP (i) with one fixed slope in a more compact way in Figures S5.1, S5.2 and S5.3. Similar conclusions to the Poisson distributed response can be inferred. In particular, in Figure S5.1 we notice that the α-*criterion* performs much better than the t-*criterion* for DGPs (i) and (ii). For DGP (iii) (last row), the model is more complex since we move to 2-dimensional random effects and the model struggles more in identifying the true number of clusters. In this case, we are able to identify a t performing better than the α-*criterion*. Nevertheless, tuning the most well performing t is computationally expensive, and the α-*criterion* gives a well performing alternative.

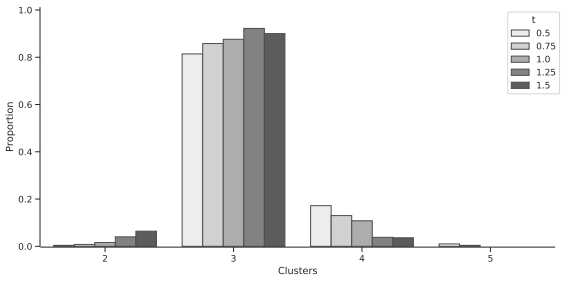
On the other hand, we compare the results of SPGLMM to the ones obtained through

Figure S5.1: Barplot for the frequency a certain number of clusters is identified over 500 runs, across different values of *alpha* and t, for DGPs (i), (ii) and (iii) for Bernoulli response with one fixed slope.

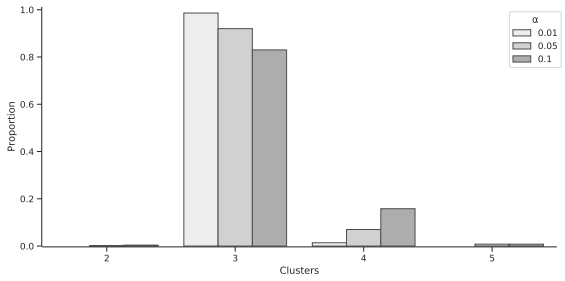
(a) DGP (i) - α-*criterion*



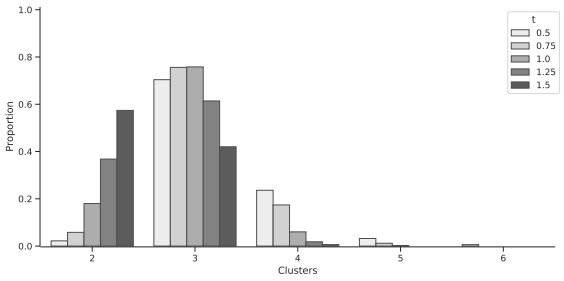
(b) DGP (i) - *t-criterion*



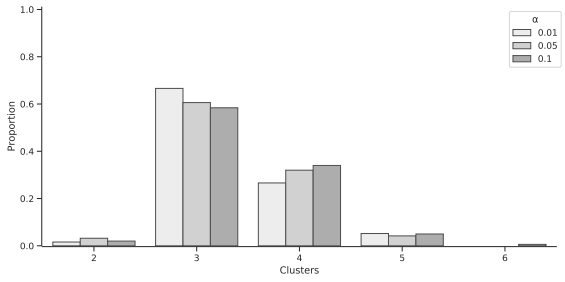
(c) DGP (i) - α-*criterion*



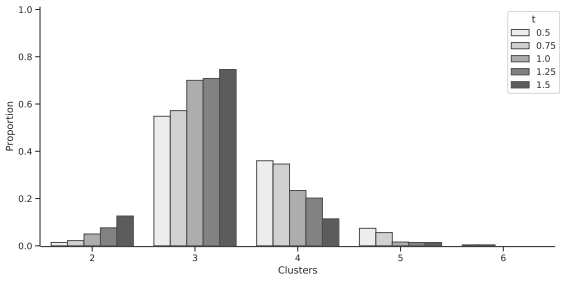
(d) DGP(ii) - *t-criterion*



(e) DGP (iii) - α-*criterion*



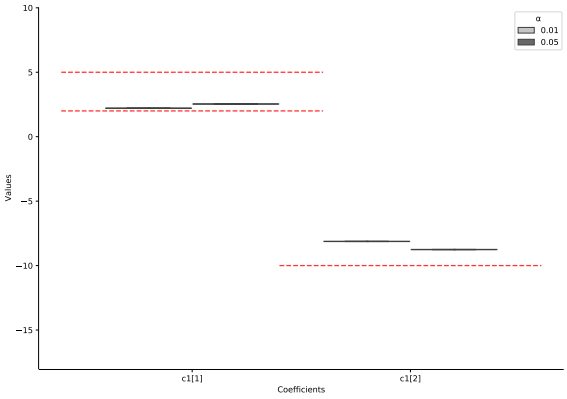
(f) DGP (iii) - *t-criterion*



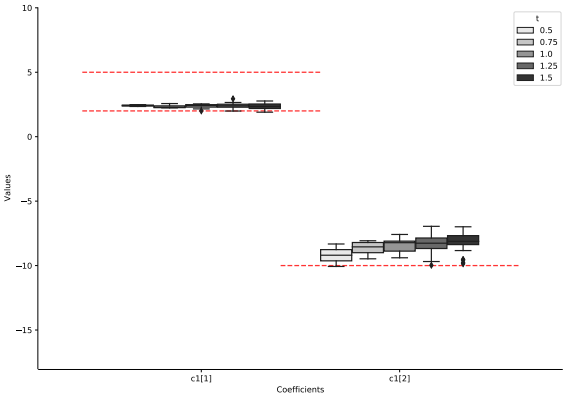
Notes: to ease the comparison, the proportions on y-axis are reported on the same scale. The number of identified clusters is reported on the x-axis. In panels (a), (c), and (e), results across $\alpha = 0.01, 0.05, 0.1$ are represented with different shadows of grey (see legend in the right-up corner). Similarly, in panel (b) results across $t = 0.25, 0.5, 0.75, 1, 1.25$ are addressed.

Figure S5.2: Boxplots for the random intercept $\mathbf{c}_1$ distribution for the DGP (i) for Bernoulli response, with one fixed slope.

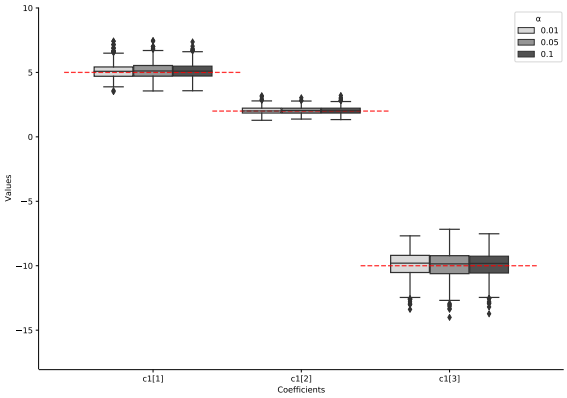
(a) $\mathbf{c}_1$ (2 groups) - *α-criterion*



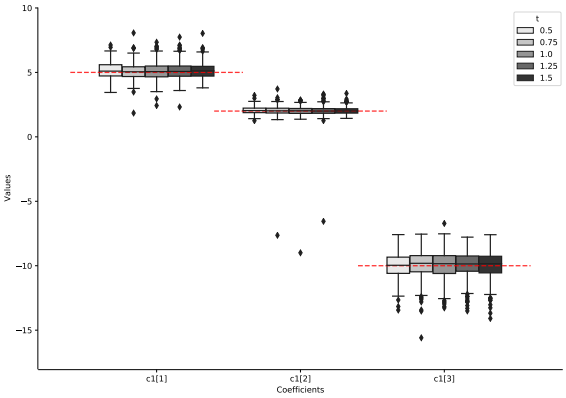
(b) $\mathbf{c}_1$ (2 groups) - *t-criterion*



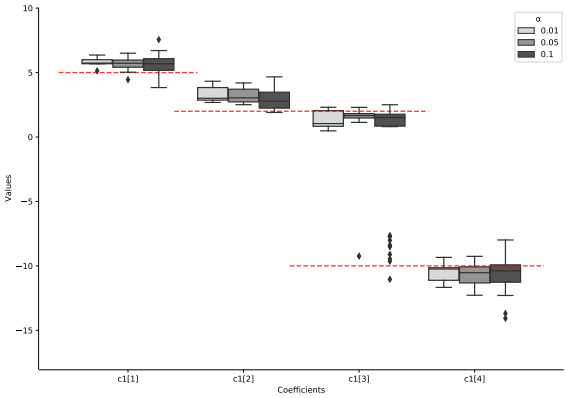
(c) $\mathbf{c}_1$ (3 groups) - *α-criterion*



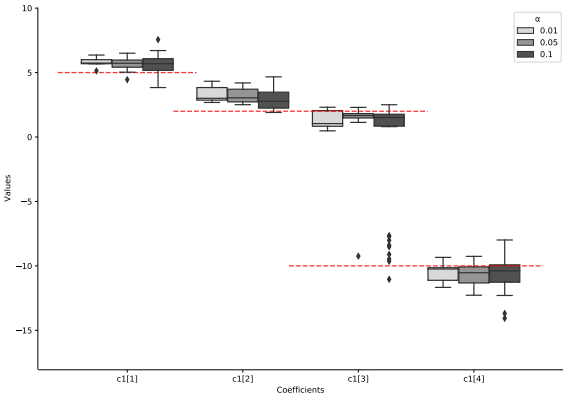
(d) $\mathbf{c}_1$ (3 groups) - *t-criterion*



(e) $\mathbf{c}_1$ (4 groups) - *α-criterion*



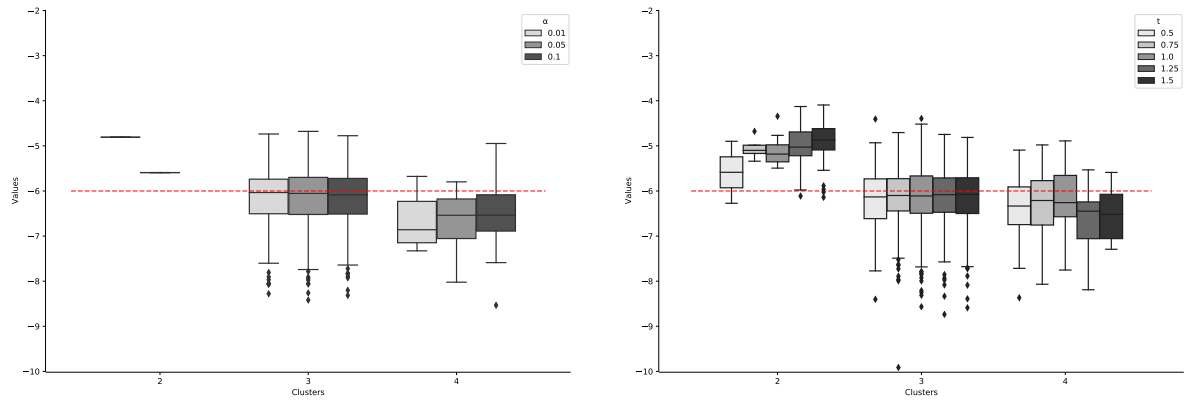
(f) $\mathbf{c}_1$ (4 groups) - *t-criterion*



Notes: For each of the 500 runs with a chosen threshold, we represent boxplots of the value for the components of the random intercept $\mathbf{c}_1$ (y-axis) according to the number of identified clusters (panels (a) and (b) for 2 clusters, panels (c) and (d) for 3 clusters, panels (e) and (f) for 4 clusters). In the left panels, we report the results of the SPGLMM run via *α-criterion*, while in the right panels the results obtained via *t-criterion*. The horizontal dotted lines indicate the simulated coefficients.

Figure S5.3: Boxplots for the fixed slope β_1 distribution for the DGP (i) for Bernoulli response, with one fixed slope.

(a) β_1 - *α-criterion* (b) β_1 - *t-criterion*



Notes: For each of the 500 runs with a chosen threshold, we represent boxplots of the value for the fixed slope β_1 (y-axis) according to the number of identified clusters (indicated on the x-axis). In the left panel, we report the results of the SPGLMM run via *α-criterion*, while in the right panel the results obtained via *t-criterion*. The horizontal dotted lines indicate the simulated coefficients.

a parametric GLMM, implemented in the function *glmer* in R package *lme4* (Bates et al. (2015), R Core Team (2022)). Also in this case, we focus on DGP (i) with one fixed slope, to be in line with the case study addressed in Section S4. We simulate 100 different DGPs as in Eq. (S5.1) and we fit, in turn, two different GLMMs, as described in Section 4.1. Moreover, we fit the SPGLMM with *α-criterion* with $\alpha = 0.05$, knowing that with such a value the algorithm identifies 3 clusters in the 96.6% of the times, as highlighted in Table S7.3. For each of the three models (i.e., GLMM with 10 random intercepts, GLMM with 3 random intercepts, SPGLMM), we represent through boxplots the distribution of the obtained coefficients across the 100 DGPs, emphasizing the true values through dotted lines, respectively in Figure S5.4, panels (a), (b) and (c). We report in Table S5.1 the summary statistics of the Goodness-of-Fit (GoF) metrics (Sensitivity, Specificity and Accuracy) retrieved by the confusion matrix for each model, which is computed comparing y_{ij} of the DGP and the estimated $\hat{y}_{ij}$, which assumes value 1 if $\hat{\mu}_{ij} > 0.5$ and 0 otherwise.

Results in Table S5.1 show that the SPGLMM performs almost as well as the GLMM in which 3 clusters are provided to the parametric model. Similar conclusions to the Poisson case can be drawn. The case in which we run a GLMM with 10 groups performs slightly better, as expected since the models have more flexibility to adapt to the differences in each group. Nevertheless, the difference in the performance can be appreciated only at the third or fourth decimal number in Sensitivity, Specificity and Accuracy. Concerning the computation of the coefficients, in the boxplots in Figure S5.4 we can appreciate that the actual values are correctly identified in all the cases, with a slightly higher presence of outliers in panels (a) and (c).

Table S5.1: Summary statistics of the GoF metrics estimates for DGP (i), Bernoulli response, with GLMM (10 groups and 3 clusters) and SPGLMM (3 clusters, α-*criterion*, $\alpha = 0.05$).

Model	Metric	Mean	Std. dev.	Quantile		
				25%	50%	75%
GLMM, 10 groups	*Sensitivity*	0.9421	0.0088	0.9367	0.9414	0.9475
	Specificity	0.9390	0.0109	0.9306	0.9403	0.9466
	Accuracy	0.9406	0.0082	0.9348	0.9418	0.9460
GLMM, 3 clusters	*Sensitivity*	0.9403	0.0082	0.9343	0.9410	0.9459
	Specificity	0.9370	0.0111	0.9301	0.9376	0.9450
	Accuracy	0.9387	0.0083	0.9338	0.9390	0.9442
SPGLMM, 3 clusters	*Sensitivity*	0.9400	0.0083	0.9341	0.9410	0.9450
	Specificity	0.9370	0.0111	0.9302	0.9372	0.9444
	Accuracy	0.9385	0.0083	0.9338	0.9386	0.9442

S6 The entropy and the elbow method

S6.1 Definition of the (average) entropy

As mentioned in Masci et al. (2022), the uncertainty of classification - with which the algorithm classifies groups into clusters - can be evaluated by measuring the entropy of the rows of the conditional weights matrix $\mathbf{W}$ (see Eq. (5)). For each group i, for $i = 1, ..., N$, the *entropy* E_i of each array $[W_{i1}, ..., W_{iM}]$ for $i = 1, ..., N$ is defined as $E_i = -\sum_{m=1}^{M} W_{im} \ln W_{im}$. The *average entropy*[*] $E = \frac{1}{N}\sum_{i=1}^{N} E_i$ assesses the level of uncertainty for which each group (i.e., the element at the higher hierarchical level) is assigned to a cluster: the closer to 0, the less uncertain the assignment to the cluster is. In fact, in the best case, the algorithm assigns each group i to a cluster m with probability 1 and each row of the matrix $\mathbf{W}$ would be composed of $M - 1$ values equal to 0 and one value equal to 1 and the entropy E_i would assume value 0.

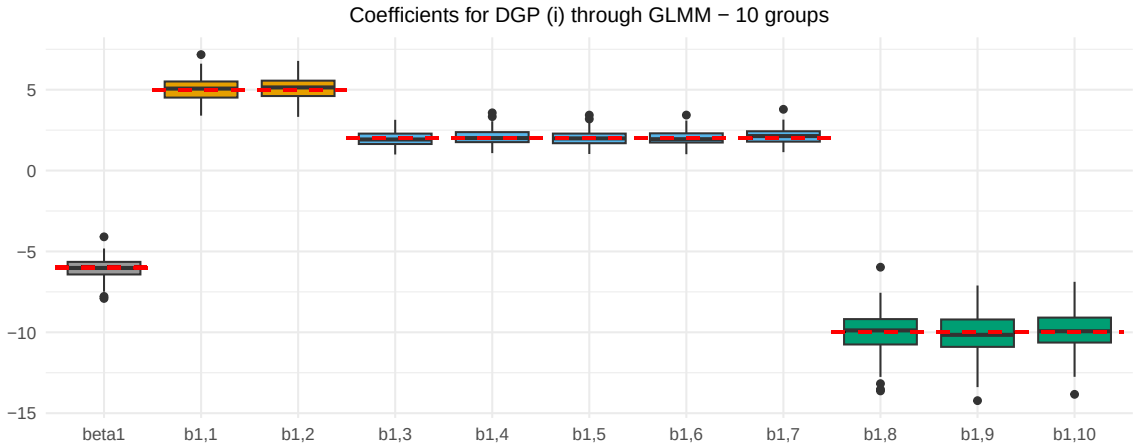
S6.2 Discussion on the entropy results in the simulation studies

For the sake of completeness, we report in Figure S6.1 the boxplots of the average entropy distributions for the simulated cases both for Poisson and Bernoulli responses. This information can be also found, respectively, in Tables S7.1, S7.3, S7.5 and S7.6. We observe that the entropy is very low for all the cases, for both α and *t-criteria*. Moreover, we notice that the entropy assumes higher values in all the cases in which the algorithm identifies 4 clusters (one cluster more than the true number), compared to the cases in which the algorithm identifies 2 and 3 clusters. These plots should help in understanding

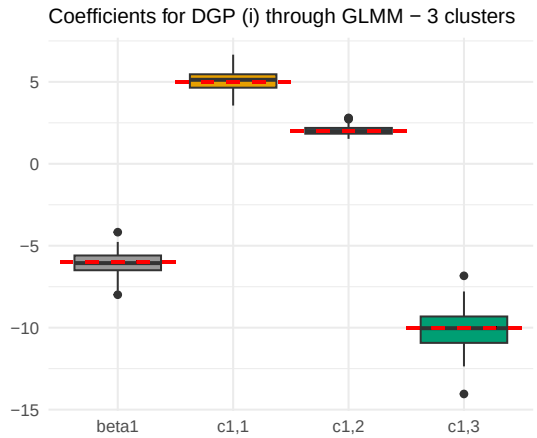
[*]Simply named *entropy* in the following, without any distinction.

Figure S5.4: Boxplots representing the distribution of the fixed slope β_1 and random intercepts for DGP (i) for Bernoulli response, for 100 iterations of the DGP.

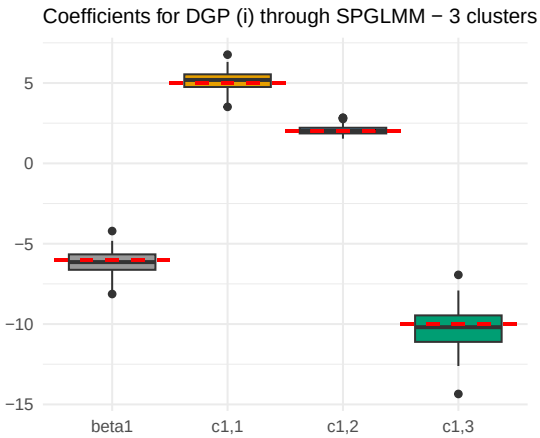
(a) GLMM (10 groups)



(b) GLMM (3 clusters)



(c) SPGLMM (3 clusters)



Notes: Random intercepts are denoted by b_{1i} for $i = 1, ..., 10$ in panel (a) and c_{1m} for $m = 1, 2, 3$ in panels (b) and (c). The horizontal dotted lines indicate the simulated coefficients.

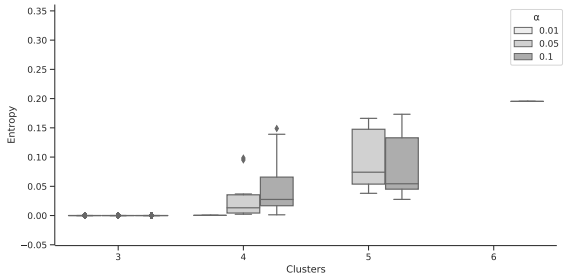
the interpretation given in Section S6.1: when the algorithm identifies the true number of clusters (i.e., the latent structure induced by the simulation), the entropy assumes a value of approximately zero. This suggests that, as addressed in the next section, the plot of the entropy across different values of t identifies an elbow which could be considered a good indicator for the identification of the optimal number of clusters (equal to 3 in our case).

S6.3 The elbow method

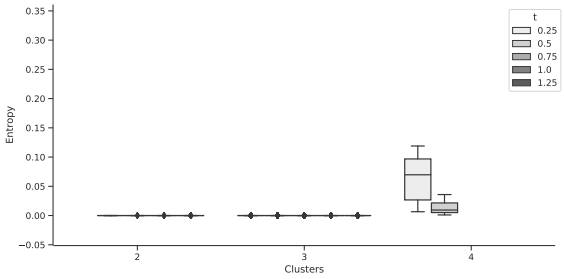
As anticipated, the selection of the best performing t within the *t-criterion* is not trivial: the SPGLMM outputs could be very sensitive to it. The average entropy could serve as a

Figure S6.1: Average entropy plot for each of the four simulated cases with one fixed slope.

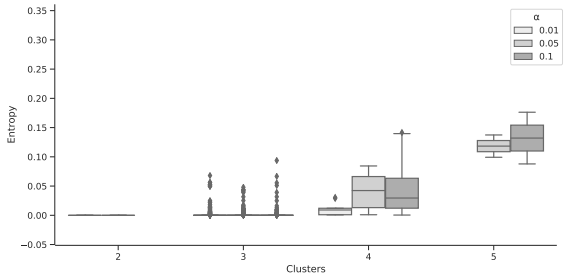
(a) DGP - Poisson - α-*criterion*



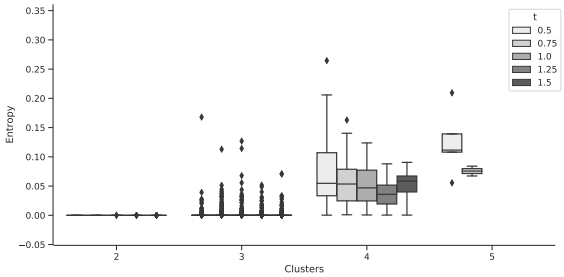
(b) DGP - Poisson - *t-criterion*



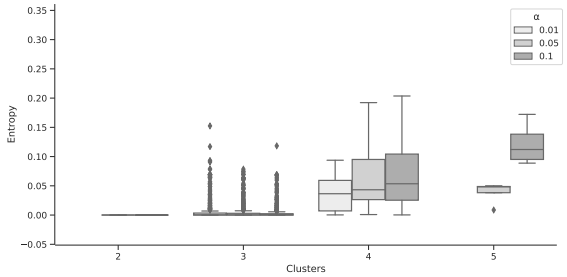
(c) DGP (i) - Bernoulli - α-*criterion*



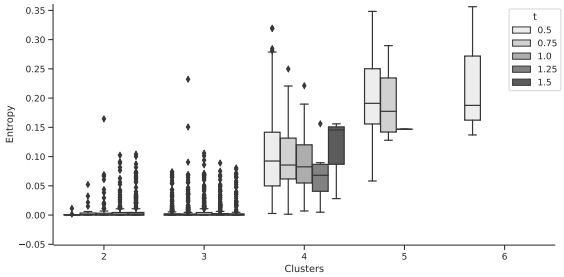
(d) DGP (i) - Bernoulli - *t-criterion*



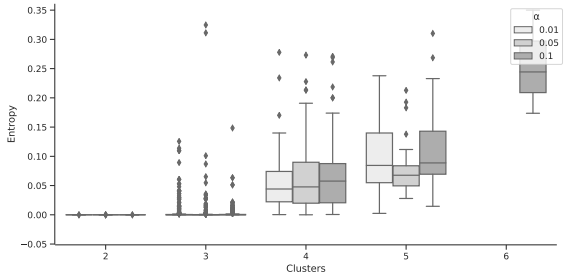
(e) DGP (ii) - Bernoulli - α-*criterion*



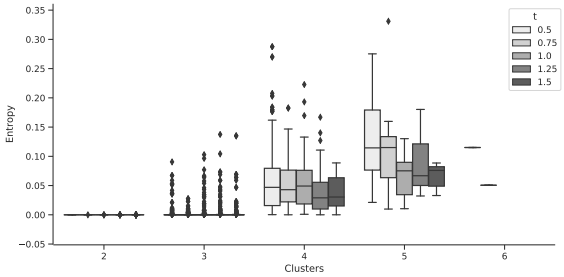
(f) DGP (ii) - Bernoulli - *t-criterion*



(g) DGP (iii) - Bernoulli - α-*criterion*



(h) DGP (iii) - Bernoulli - *t-criterion*



Notes: The first row (panels (a) and (b)) concerns the Poisson response, while the last three rows (panels (c-h)) represent results obtained by fitting an SPGLMM with a Bernoulli response. For each of the 500 runs with a chosen threshold (see the legend in the right-up corner), we represent boxplots of the average entropy (y-axis) according to the number of clusters M identified by the algorithm (indicated on the x-axis). In the left panels, we report the results of the SPGLMM run via α-*criterion*, while in the right panels the results obtained via *t-criterion*.

driver for the selection of the best-performing t. In order to identify such a t, we could fit distinct SPGLMMs across different values of t, and compute, for each of them, the entropy $E(t)$. By plotting $E(t)$ in function of t, inspired by the *elbow*-method in classical clustering algorithms, we could identify the best t, namely t^*, in correspondence of an elbow in the piecewise-continuous diagram of $E(t)$. The rationale is to choose a threshold t^* for which the induced number of clusters has low average entropy $E(t^*)$, so that the increase of the threshold to $t^* + \epsilon$ would not make $E(t^* + \epsilon)$ significantly lower than $E(t^*)$ and would not bring particular improvements to the modelling, aside from identifying fewer clusters, which might not be what we are interested in. In other words, we should select the first value of E (in increasing order) after which the decrease in the entropy is negligible.

In Figure S6.2 we provide the application of such method to the simulated data (DGP for Poisson response and DGP (i) for Bernoulli response described in Section S5). The results find a match in the Tables S7.1, S7.3, S7.5 and S7.6, proving that this method heuristically seems to work, though it could result in being computationally expensive because requires fitting the model multiple times until when an elbow is clearly identified.

S7 Tables with simulation study outputs

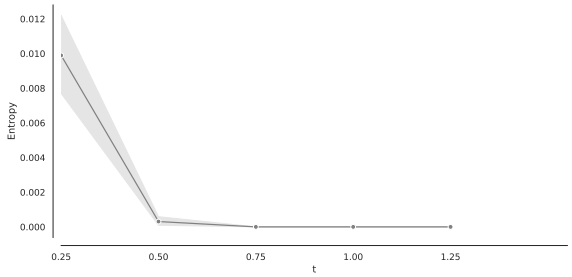
We report extended simulation results with:

- **Poisson response** for DGP in Tables S7.1 and S7.2 (with one and two fixed slopes, respectively);
- **Bernoulli response** for DGP(i) in Tables S7.3 and S7.4 (with one and two fixed slopes, respectively), for DGP(ii) in Table S7.5 (with one fixed slope) and for DGP(iii) in Table S7.6 (with one fixed slope).

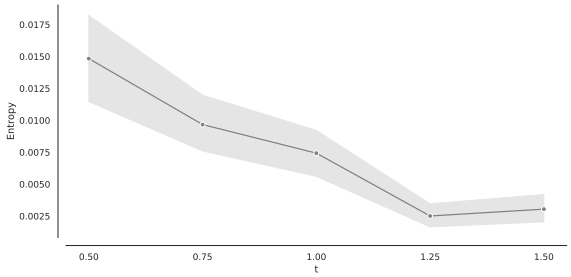
In each table, estimates of the proportion of identified clusters, entropy, weights $\hat{\omega}$, random and fixed coefficients are reported in terms of mean and standard deviation (sd) across the 500 iterations.

Figure S6.2: Average entropy plot in function of t for each of the four simulated cases with one fixed slope.

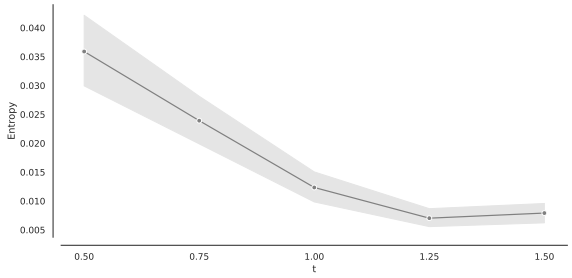
(a) DGP - Poisson



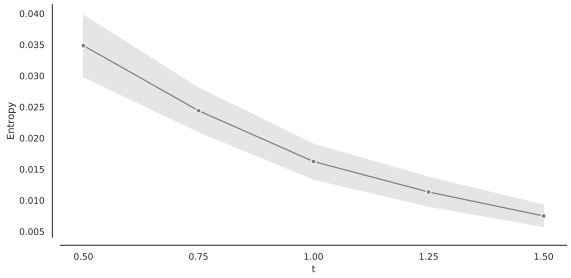
(b) DGP (i) - Bernoulli



(c) DGP (ii) - Bernoulli



(d) DGP (iii) - Bernoulli



Notes: The grey full line represents the average entropy averaged across the 500 runs with a certain t (x-axis), while the shadow grey is the 95% confidence interval for the mean.
In panel (a), we clearly identify the elbow at $t = 0.50$, which corresponds to the level at which with maximum Proportion in Table S7.1 the true number of clusters is identified (95 %).
Similarly, in panels (b) and (c), we can identify elbows at $t = 1.25$ and $t = 1.00$, finding confirmation in Tables S7.3 and S7.5 with a maximum Proportion of 92.2% and 76%, respectively.
The identification an elbow in the plot in panel (d) is less trivial. This is due to the fact that, as shown in Table S7.6, the maximum Proportion (74.6%) can be identified at $t = 1.50$. This means that in order to clearly visualize the elbow we should run the model for $t \geq 1.5$.

Table S7.1: Results obtained by SPGLMM algorithm for the **Poisson response** through DGP for $t = 0.25, 0.50, 0.75, 1.00, 1.25$, $\alpha = 0.01, 0.05, 0.10$, with **1 fixed slope**.

	N clusters	Proportion (out of 500)	Entropy(sd)	$\hat{\omega}$(sd)	$\hat{c}_1$(sd)	$\hat{\beta}_1$(sd)
	2	0.012	0.000 (0.00)	[0.50 (0.00), 0.30 (0.00)]	[1.01 (0.02), -0.98 (0.10)]	0.29 (0.03)
$\alpha = 0.01$	**3**	**0.986**	**0.000 (0.00)**	**[0.20 (0.00), 0.50 (0.00), 0.30 (0.00)]**	**[2.50 (0.02), 1.00 (0.03), -1.01 (0.10)]**	**0.30 (0.02)**
	4	0.002	0.000 (-)	[0.20 (0.00), 0.50 (0.00), 0.10 (0.00), 0.20 (0.00)]	[2.51 (-), 0.98 (-), -0.66 (-), -1.47 (-)]	0.30 (-)
	2	0.018	0.000 (0.00)	[0.50 (0.00), 0.30 (0.00)]	[1.00 (0.04), -0.98 (0.11)]	0.30 (0.03)
$\alpha = 0.05$	**3**	**0.948**	**0.000 (0.00)**	**[0.20 (0.00), 0.50 (0.00), 0.30 (0.00)]**	**[2.50 (0.02), 1.00 (0.03), -0.99 (0.10)]**	**0.30 (0.02)**
	4	0.024	0.029 (0.03)	[0.20 (0.00), 0.38 (1.3), 0.19 (0.86), 0.23 (0.85)]	[2.51 (0.02), 1.06 (0.05), 0.19 (0.86), -1.20 (0.28)]	0.29 (0.02)
	2	0.018	0.000 (0.00)	[0.50 (0.00), 0.30 (0.00)]	[1.00 (0.03), -1.11 (0.13)]	0.31 (0.02)
$\alpha = 0.10$	**3**	**0.872**	**0.000 (0.00)**	**[0.20 (0.10), 0.50 (0.19), 0.30 (0.00)]**	**[2.50 (0.07), 1.00 (0.03), -1.00 (0.10)]**	**0.30 (0.02)**
	4	0.094	0.047 (0.04)	[0.20 (0.20), 0.35 (1.57), 0.22 (1.15), 0.23 (0.83)]	[2.50 (0.03), 1.11 (0.29), 0.21 (0.84), -1.13 (0.23)]	0.30 (0.02)
	2	-	-	-	-	-
$t = 0.25$	**3**	**0.846**	**0.000 (0.00)**	**[0.20 (0.00), 0.50 (0.00), 0.30 (0.00)]**	**[2.50 (0.02), 1.00 (0.03), -1.01 (0.10)]**	**0.30 (0.02)**
	4	0.154	0.064 (0.04)	[0.20 (0.00), 0.50 (0.00), 0.16 (0.48), 0.14 (0.48)]	[2.49 (0.02), 1.00 (0.03), -0.87 (0.11), -1.30 (0.16)]	0.30 (0.02)
	2	0.028	0.000 (0.00)	[0.60 (0.93), 0.30 (0.00)]	[1.36 (0.34), -0.97 (0.12)]	0.30 (0.02)
$t = 0.50$	**3**	**0.950**	**0.000 (0.00)**	**[0.20 (0.00), 0.50 (0.00), 0.30 (0.00)]**	**[2.50 (0.02), 1.00 (0.03), -1.01 (0.10)]**	**0.30 (0.02)**
	4	0.022	0.014 (0.01)	[0.20 (0.00), 0.50 (0.00), 0.15 (0.50), 0.15 (0.50)]	[2.51 (0.02), 1.00 (0.02), -0.77 (0.18), -1.39 (0.21)]	0.31 (0.02)
	2	0.066	0.000 (0.00)	[0.58 (1.34), 0.33 (1.19)]	[1.43 (0.42), -0.90 (0.42)]	0.31 (0.03)
$t = 0.75$	**3**	**0.934**	**0.000 (0.00)**	**[0.20 (0.00), 0.50 (0.00), 0.30 (0.00)]**	**[2.50 (0.02), 1.00 (0.03), -1.01 (0.10)]**	**0.30 (0.02)**
	4	-	-	-	-	-
	2	0.110	0.000 (0.00)	[0.61 (1.33), 0.33 (1.14)]	[1.50 (0.39), -0.94 (0.40)]	0.30 (0.04)
$t = 1.00$	**3**	**0.890**	**0.000 (0.00)**	**[0.20 (0.00), 0.50 (0.00), 0.30 (0.00)]**	**[2.50 (0.02), 1.00 (0.03), -1.01 (0.10)]**	**0.30 (0.02)**
	4	-	-	-	-	-
	2	0.296	0.000 (0.00)	[0.50 (2.16), 0.46 (2.31)]	[1.79 (0.55), -0.50 (0.76)]	0.30 (0.03)
$t = 1.25$	**3**	**0.704**	**0.000 (0.00)**	**[0.20 (0.00), 0.50 (0.00), 0.30 (0.00)]**	**[2.50 (0.02), 1.00 (0.03), -1.00 (0.09)]**	**0.30 (0.02)**
	4	-	-	-	-	-
TV				[0.2, 0.5, 0.3]	[2.5, 1, −1]	0.3

Notes: For each case, 500 runs were performed and results are reported for the cases in which the algorithm identifies 2, 3 and 4 clusters. Estimates of entropy, weights, random and fixed coefficients are reported in terms of mean (sd). True Values (TV) of the coefficients used to simulate data are reported under the relative estimates. Results related to the true number of clusters (i.e. 3) are reported in bold. The cases for which the algorithm identifies 1 or more than 4 clusters are not reported in table, but can be identified by complementing with 1 the sum of the three reported Proportions.

Table S7.2: Results obtained by SPGLMM algorithm for the **Poisson response** through DGP for $t = 0.50, 0.75, 1.00, 1.25, 1.50$, $\alpha = 0.01, 0.05, 0.10$, with **2 fixed slopes**.

	N clusters	Proportion (out of 500)	Entropy(sd)	$\hat{\omega}$(sd)	$\hat{c}_1$(sd)	$\hat{\beta}_1$(sd)	$\hat{\beta}_2$(sd)
	2	0.040	0.000 (0.00)	[0.50 (0.00), 0.30 (0.00)]	[0.99 (0.04), -0.99 (0.14)]	0.29 (0.02)	0.90 (0.03)
$\alpha = 0.01$	**3**	**0.956**	**0.000 (0.00)**	**[0.20 (0.00), 0.50 (0.00), 0.30 (0.00)]**	**[2.50 (0.02), 1.00 (0.03), -1.01 (0.08)]**	**0.30 (0.01)**	**0.90 (0.02)**
	4	0.004	0.006 (0.00)	[0.20 (0.00), 0.20 (0.00), 0.30 (0.00), 0.30 (0.00)]	[2.52 (0.00), 1.14 (0.00), 0.94 (0.00), -1.02 (0.00)]	0.33 (0.00)	0.88 (0.00)
	2	0.028	0.000 (0.00)	[0.50 (0.00), 0.30 (0.00)]	[1.00 (0.02), -1.06 (0.09)]	0.31 (0.02)	0.90 (0.02)
$\alpha = 0.05$	**3**	**0.934**	**0.000 (0.00)**	**[0.20 (0.00), 0.50 (0.00), 0.30 (0.00)]**	**[2.50 (0.02), 1.00 (0.03), -1.00 (0.08)]**	**0.30 (0.01)**	**0.90 (0.02)**
	4	0.036	0.015 (0.02)	[0.19 (0.31), 0.37 (1.76), 0.24 (1.34), 0.21 (0.85)]	[2.50 (0.02), 1.18 (0.45), -0.11 (0.88), -1.26 (0.26)]	0.30 (0.01)	0.90 (0.01)
	2	0.028	0.000 (0.00)	[0.50 (0.00), 0.30 (0.00)]	[0.99 (0.02), -0.95 (0.06)]	0.30 (0.01)	0.91 (0.02)
$\alpha = 0.10$	**3**	**0.870**	**0.000 (0.00)**	**[0.20 (0.21), 0.50 (0.31), 0.30 (0.07)]**	**[2.49 (0.12), 0.99 (0.12), -0.99 (0.09)]**	**0.30 (0.01)**	**0.90 (0.02)**
	4	0.098	0.043 (0.04)	[0.19 (0.27), 0.35 (1.40), 0.20 (1.25), 0.26 (0.64)]	[2.50 (0.03), 1.16 (0.40), 0.38 (0.80), -1.08 (0.14)]	0.30 (0.02)	0.90 (0.01)
	2	0.024	0.000 (0.00)	[0.51 (0.28), 0.30 (0.00)]	[1.02 (0.13), -1.00 (0.10)]	0.30 (0.01)	0.89 (0.04)
$t = 0.50$	**3**	**0.968**	**0.000 (0.00)**	**[0.20 (0.14), 0.50 (0.18), 0.30 (0.05)]**	**[2.50 (0.07), 1.00 (0.08), -1.00 (0.09)]**	**0.30 (0.02)**	**0.90 (0.02)**
	4	0.008	0.000 (0.00)	[0.20 (0.00), 0.50 (0.00), 0.20 (0.00), 0.10 (0.00)]	[2.50 (0.01), 1.00 (0.02), -0.84 (0.02), -1.41 (0.03)]	0.30 (0.02)	0.90 (0.01)
	2	0.088	0.000 (0.00)	[0.50 (1.71), 0.40 (2.0)]	[1.28 (0.39), -0.91 (0.35)]	0.29 (0.04)	0.86 (0.05)
$t = 0.75$	**3**	**0.912**	**0.000 (0.00)**	**[0.20 (0.00), 0.50 (0.00), 0.30 (0.00)]**	**[2.50 (0.02), 1.00 (0.03), -1.00 (0.09)]**	**0.30 (0.01)**	**0.90 (0.02)**
	4	-	-	-	-	-	-
	2	0.172	0.000 (0.00)	[0.56 (1.19), 0.32 (1.05)]	[1.35 (0.42), -0.91 (0.35)]	0.29 (0.04)	0.86 (0.05)
$t = 1.00$	**3**	**0.828**	**0.000 (0.00)**	**[0.20 (0.00), 0.50 (0.00), 0.30 (0.00)]**	**[2.50 (0.02), 1.00 (0.03), -1.01 (0.08)]**	**0.30 (0.02)**	0.90 (0.02) n
	4	-	-	-	-	-	-
	2	0.340	0.000 (0.00)	[0.50 (1.71), 0.40 (2.00)]	[1.52 (0.57), -0.64 (0.64)]	0.30 (0.03)	0.88 (0.05)
$t = 1.25$	**3**	**0.660**	**0.000 (0.00)**	**[0.20 (0.00), 0.50 (0.00), 0.30 (0.00)]**	**[2.50 (0.02), 1.01 (0.03), -1.00 (0.08)]**	**0.30 (0.01)**	**0.90 (0.02)**
	4	-	-	-	-	-	-
	2	0.808	0.000 (0.00)	[0.51 (1.64), 0.39 (1.89)]	[1.52 (0.54), -0.67 (0.61)]	0.29 (0.04)	0.87 (0.05)
$t = 1.50$	**3**	**0.192**	**0.000 (0.00)**	**[0.20 (0.00), 0.50 (0.00), 0.30 (0.00)]**	**[2.51 (0.02), 0.99 (0.02), -1.01 (0.09)]**	**0.30 (0.01)**	**0.90 (0.02)**
	4	-	-	-	-	-	-
TV				[0.2, 0.5, 0.3]	[2.5, 1, −1]	0.3	0.9

Notes: For each case, 500 runs were performed and results are reported for the cases in which the algorithm identifies 2, 3 and 4 clusters. Estimates of entropy, weights, random and fixed coefficients are reported in terms of mean (sd). True Values (TV) of the coefficients used to simulate data are reported under the relative estimates. Results related to the true number of clusters (i.e. 3) are reported in bold. The cases for which the algorithm identifies 1 or more than 4 clusters are not reported in table, but can be identified by complementing with 1 the sum of the three reported Proportions.

Table S7.3: Results obtained by SPGLMM algorithm for the **Bernoulli response** through DGP(i) in Eq. (S5.1), for $t = 0.50, 0.75, 1.00, 1.25, 1.50$, $\alpha = 0.01, 0.05, 0.10$, with **1 fixed slope**.

	N clusters	Proportion (out of 500)	Entropy(sd)	$\hat{\omega}$(sd)	$\hat{c}_1$(sd)	$\hat{\beta}_1$(sd)
	2	0.002	0.000 (-)	[0.70 (-), 0.30 (-)]	[2.21 (-), -8.13 (-)]	-4.81 (-)
$\alpha = 0.01$	**3**	**0.980**	**0.002 (0.01)**	**[0.20 (0.16), 0.50 (0.16), 0.30 (0.00)]**	**[5.13 (0.62), 2.05 (0.28), -9.93 (1.01)]**	**-6.13 (0.62)**
	4	0.018	0.010 (0.01)	[0.17 (0.47), 0.29 (1.29), 0.24 (1.71), 0.30 (0.00)]	[5.78 (0.33), 3.25 (0.60), 1.28 (0.71), -10.42 (0.80)]	-6.62 (0.62)
	2	0.002	0.000 (-)	[0.70 (-), 0.30 (-)]	[2.54 (-), -8.76 (-)]	-5.59 (-)
$\alpha = 0.05$	**3**	**0.966**	**0.001 (0.01)**	**[0.20 (0.14), 0.50 (0.14), 0.30 (0.00)]**	**[5.15 (0.62), 2.04 (0.28), -9.93 (1.05)]**	**-6.14 (0.63)**
	4	0.028	0.041 (0.03)	[0.18 (0.41), 0.24 (1.34), 0.29 (1.39), 0.29 (0.26)]	[5.65 (0.52), 3.20 (0.55), 0.91 (2.94), -10.67 (1.03)]	-6.66 (0.69)
	2	-	-	-	-	-
$\alpha = 0.10$	**3**	**0.916**	**0.001 (0.01)**	**[0.20 (0.10), 0.50 (0.10), 0.30 (0.00)]**	**[5.14 (0.59), 2.05 (0.29), -9.98 (0.99)]**	**-6.16 (0.59)**
	4	0.080	0.045 (0.04)	[0.18 (0.42), 0.30 (1.48), 0.25 (1.48), 0.27 (0.55)]	[5.63 (0.69), 2.93 (0.81), -0.76 (4.45), -10.62 (1.23)]	-6.50 (0.71)
	2	0.004	0.000 (0.00)	[0.70 (0.00), 0.30 (0.00)]	[2.42 (0.10), -9.20 (1.23)]	-5.59 (0.97)
$t = 0.50$	**3**	**0.812**	**0.001 (0.00)**	**[0.20 (0.00), 0.50 (0.00), 0.30 (0.00)]**	**[5.14 (0.59), 2.05 (0.27), -10.03 (0.97)]**	**-6.19 (0.61)**
	4	0.174	0.074 (0.06)	[0.18 (0.42), 0.33 (1.71), 0.25 (1.60), 0.24 (0.81)]	[5.54 (0.76), 2.88 (1.07), -2.78 (5.54), -10.66 (1.29)]	-6.35 (0.69)
	2	0.008	0.000 (0.00)	[0.70 (0.00), 0.30 (0.00)]	[2.35 (0.15), -8.66 (0.63)]	-5.05 (0.28)
$t = 0.75$	**3**	**0.858**	**0.002 (0.01)**	**[0.20 (0.27), 0.50 (0.19), 0.30 (0.10)]**	**[5.09 (0.59), 2.02 (0.54), -9.92 (0.99)]**	**-6.14 (0.60)**
	4	0.130	0.057 (0.02)	[0.18 (0.41), 0.39 (1.62), 0.22 (1.5), 0.22 (0.81)]	[5.40 (0.70), 2.61 (0.92), -4.53 (5.39), -10.73 (1.13)]	-6.28 (0.67)
	2	0.016	0.000 (0.00)	[0.70 (0.00), 0.30 (0.00)]	[2.36 (0.19), -8.44 (0.60)]	-5.10 (0.39)
$t = 1.00$	**3**	**0.876**	**0.002 (0.01)**	**[0.20 (0.27), 0.50 (0.19), 0.30 (0.10)]**	**[5.10 (0.64), 2.00 (0.60), -9.94 (1.02)]**	**-6.13 (0.65)**
	4	0.108	0.051 (0.03)	[0.19 (0.37), 0.43 (1.38), 0.19 (1.26), 0.20 (0.79)]	[5.34 (0.73), 2.41 (0.93), -5.93 (4.96), -10.68 (1.26)]	-6.23 (0.65)
	2	0.040	0.000 (0.00)	[0.70 (0.00), 0.30 (0.00)]	[2.41 (0.21), -8.30 (0.81)]	-5.03 (0.50)
$t = 1.25$	**3**	**0.922**	**0.001 (0.01)**	**[0.20 (0.27), 0.50 (0.23), 0.30 (0.05)]**	**[5.14 (0.64), 2.01 (0.50), -9.92 (0.97)]**	**-6.13 (0.59)**
	4	0.038	0.039 (0.03)	[0.18 (0.36), 0.43 (1.45), 0.23 (1.22), 0.15 (0.82)]	[5.88 (0.67), 2.60 (0.97), -7.56 (5.10), -12.03 (1.66)]	-6.64 (0.70)
	2	0.064	0.000 (0.00)	[0.70 (0.00), 0.30 (0.00)]	[2.35 (0.24), -8.16 (0.74)]	-4.95 (0.50)
$t = 1.50$	**3**	**0.900**	**0.001 (0.01)**	**[0.20 (0.20), 0.50 (0.20), 0.30 (0.00)]**	**[5.13 (0.59), 2.03 (0.27), -9.95 (1.03)]**	**-6.14 (0.61)**
	4	0.036	0.050 (0.03)	[0.19 (0.31), 0.46 (1.11), 0.19 (1.03), 0.16 (0.68)]	[5.46 (0.59), 2.41 (0.57), -8.30 (3.76), -11.64 (1.21)]	-6.52 (0.59)
TV				[0.2, 0.5, 0.3]	$[5, 2, -10]$	-6

Notes: For each case, 500 runs were performed and results are reported for the cases in which the algorithm identifies 2, 3 and 4 clusters. Estimates of entropy, weights, random and fixed coefficients are reported in terms of mean (sd). True Values (TV) of the coefficients used to simulate data are reported under the relative estimates. Results related to the true number of clusters (i.e. 3) are reported in bold. The cases for which the algorithm identifies 1 or more than 4 clusters are not reported in table, but can be identified by complementing with 1 the sum of the three reported Proportions.

Table S7.4: Results obtained by SPGLMM algorithm for the **Bernoulli response** through DGP(i) in Eq. (S5.1), for $t = 0.50, 0.75, 1.00, 1.25, 1.50$, $\alpha = 0.01, 0.05, 0.10$, with **2 fixed slopes**.

	N clusters	Proportion (out of 500)	Entropy(sd)	$\hat{\omega}$(sd)	$\hat{c}_1$(sd)	$\hat{\beta}_1$(sd)	$\hat{\beta}_2$(sd)
	2	0.008	0.000 (0.00)	[0.70 (0.00), 0.30 (0.00)]	[2.46 (0.24), -8.98 (1.06)]	-5.38 (0.50)	2.79 (0.31)
$\alpha = 0.01$	**3**	**0.970**	**0.002 (0.01)**	**[0.20 (0.16), 0.50 (0.16), 0.30 (0.00)]**	**[5.16 (0.63), 2.05 (0.29), -10.23 (1.09)]**	**-6.20 (0.64)**	**3.11 (0.36)**
	4	0.022	0.024 (0.02)	[0.19 (0.29), 0.40 (0.85), 0.17 (0.62), 0.24 (0.88)]	[5.19 (0.52), 2.58 (0.42), -2.90 (5.22), -11.30 (1.17)]	-6.34 (0.48)	3.14 (0.30)
	2	-	-	-	-	-	-
$\alpha = 0.05$	**3**	**0.952**	**0.002 (0.01)**	**[0.20 (0.22), 0.50 (0.22), 0.30 (0.00)]**	[5.17 (0.63), 2.05 (0.30), -10.16 (1.03)]	-6.19 (0.62)	3.10 (0.36)
	4	0.048	0.032 (0.03)	[0.19 (0.28), 0.38 (1.30), 0.18 (1.00), 0.25 (0.71)]	[5.38 (0.65), 2.56 (0.58), -2.20 (5.03), -10.92 (1.39)]	-6.41 (0.66)	3.13 (0.36)
	2	0.004	0.000 (0.00)	[0.70 (0.00), 0.30 (0.00)]	[2.72 (0.36), -8.83 (1.15)]	-5.33 (0.79)	2.29 (0.11)
$\alpha = 0.10$	**3**	**0.906**	**0.002 (0.01)**	**[0.20 (0.12), 0.50 (0.12), 0.30 (0.00)]**	**[5.15 (0.62), 2.06 (0.29), -10.12 (1.06)]**	**-6.19 (0.62)**	3.06 (0.35)
	4	0.080	0.039 (0.03)	[0.19 (0.33), 0.38 (1.44), 0.20 (1.32), 0.23 (0.85)]	[5.36 (0.83), 2.67 (0.88), -3.02 (5.27), -10.9 (1.04)]	-6.35 (0.51)	3.17 (0.26)
	2	-	-	-	-	-	-
$t = 0.50$	**3**	**0.780**	**0.002 (0.01)**	**[0.20 (0.40), 0.50 (0.34), 0.30 (0.07)]**	**[5.14 (0.66), 2.00 (0.72), -10.22 (1.16)]**	**-6.18 (0.68)**	**3.09 (0.35)**
	4	0.196	0.078 (0.05)	[0.18 (0.4), 0.33 (1.69), 0.26 (1.43)]	[5.53 (0.82), 2.82 (1.05), -2.91 (5.55), -10.90 (1.20)]	-6.44 (0.63)	3.20 (0.37)
	2	0.008	0.000 (0.00)	[0.70 (0.00), 0.30 (0.00)]	[2.36 (0.11), -8.42 (0.30)]	-5.05 (0.30)	2.55 (0.37)
$t = 0.75$	**3**	**0.824**	**0.003 (0.01)**	**[0.20 (0.18), 0.50 (0.18), 0.30 (0.00)]**	**[5.09 (0.65), 2.02 (0.30), -9.99 (1.03)]**	**-6.11 (0.63)**	**3.03 (0.32)**
	4	0.154	0.053 (0.04)	[0.18 (0.41), 0.37 (1.65), 0.24 (1.46), 0.21 (0.83)]	[5.33 (0.71), 2.61 (1.00), -4.52 (5.44), -10.72 (1.19)]	-6.14 (0.65)	3.07 (0.33)
	2	0.010	0.000 (0.00)	[0.70 (0.00), 0.30 (0.00)]	[2.56 (0.19), -8.47 (0.50)]	-5.09 (0.37)	2.45 (0.34)
$t = 1.00$	**3**	**0.880**	**0.002 (0.01)**	**[0.20 (0.18), 0.50 (0.18), 0.30 (0.00)]**	**[5.16 (0.61), 2.06 (0.31), -10.17 (1.12)]**	**-6.19 (0.68)**	**3.10 (0.37)**
	4	0.110	0.064 (0.04)	[0.20 (0.36), 0.44 (1.19), 0.16 (1.00), 0.19 (0.70)]	[5.29 (0.69), 2.25 (0.56), -7.17 (4.44), -11.23 (0.87)]	-6.43 (0.54)	3.21 (0.30)
	2	0.044	0.000 (0.00)	[0.70 (0.00), 0.30 (0.00)]	[2.38 (0.17), -8.23 (0.59)]	-4.84 (0.38)	2.41 (0.26)
$t = 1.25$	**3**	**0.910**	**0.002 (0.01)**	**[0.20 (0.24), 0.50 (0.24), 0.30 (0.00)]**	**[5.22 (0.70), 2.04 (0.29), -10.16 (1.06)]**	**-6.19 (0.62)**	**3.09 (0.36)**
	4	0.046	0.049 (0.03)	[0.20 (0.00), 0.50 (0.00), 0.15 (0.50), 0.15 (0.50)]	[5.45 (0.62), 2.20 (0.28), -9.61 (0.96), -11.95 (1.40)]	-6.60 (0.60)	3.33 (0.32)
	2	0.064	0.000 (0.00)	[0.70 (0.00), 0.30 (0.00)	[2.40 (0.24), -8.38 (0.72)]	-5.05 (0.48)	2.55 (0.29)
$t = 1.50$	**3**	**0.898**	**0.002 (0.01)**	**[0.20 (0.18), 0.50 (0.18), 0.30 (0.00)]**	**[5.25 (0.66), 2.09 (0.31), -10.40 (1.22)]**	**-6.30 (0.70)**	**3.15 (0.41)**
	4	0.038	0.046 (0.02)	[0.19 (0.31), 0.49 (0.51), 0.17 (0.57), 0.15 (0.60)]	[5.46 (0.70), 2.26 (0.25), -9.56 (3.11), -12.47 (1.99)]	-6.58 (0.80)	3.25 (0.47)
TV				[0.2,0.5,0.3]	$[5, 2, -10]$	-6	3

Notes: For each case, 500 runs were performed and results are reported for the cases in which the algorithm identifies 2, 3 and 4 clusters. Estimates of entropy, weights, random and fixed coefficients are reported in terms of mean (sd). True Values (TV) of the coefficients used to simulate data are reported under the relative estimates. Results related to the true number of clusters (i.e. 3) are reported in bold. The cases for which the algorithm identifies 1 or more than 4 clusters are not reported in table, but can be identified by complementing with 1 the sum of the three reported Proportions.

Table S7.5: Results obtained by SPGLMM algorithm for the **Bernoulli response** through DGP(ii) in Eq. (S5.2), for $t = 0.50, 0.75, 1.00, 1.25, 1.50$, $\alpha = 0.01, 0.05, 0.10$, with **1 fixed slope**.

	N clusters	Proportion (out of 500)	Entropy(sd)	$\hat{\omega}$(sd)	$\hat{c}_1$(sd)	$\hat{\beta}_1$(sd)	$\hat{\beta}_2$(sd)
	2	-	-	-	-	-	-
$\alpha = 0.01$	**3**	**0.986**	**0.008 (0.02)**	**[0.20 (0.16), 0.50 (0.25), 0.30 (0.19)]**	**[10.51 (1.61), 5.25 (0.77), 0.01 (0.50)]**	**10.49 (1.47)**	**-6.38 (0.95)**
	4	0.014	0.038 (0.04)	[0.20 (0.00), 0.39 (1.25), 0.20 (0.76), 0.21 (0.64)]	[12.79 (2.72), 6.76 (0.94), 3.24 (0.95), -0.96 (1.20)]	12.25 (2.21)	-7.56 (1.29)
	2	0.002	0.000 (-)	[0.70 (0.00), 0.30 (0.00)]	[6.35 (-), -0.25 (-)]	10.87 (-)	-6.65 (-)
$\alpha = 0.05$	**3**	**0.920**	**0.006 (0.02)**	**[0.20 (0.13), 0.50 (0.17), 0.30 (0.14)]**	**[10.46 (1.64), 5.19 (0.83), 0.06 (0.55)]**	**10.36 (1.53)**	**-6.31 (0.97)**
	4	0.070	0.065 (0.05)	[0.19 (0.28), 0.35 (1.46), 0.23 (1.26), 0.22 (0.83)]	[11.40 (1.93), 6.45 (1.45), 3.27 (1.84), -0.61 (1.16)]	11.16 (1.76)	-6.82 (1.14)
	2	0.004	0.000 (0.00)	[0.70 (0.00), 0.30 (0.00)]	[4.78 (0.42), -0.03 (0.17)]	7.39 (0.54)	-4.29 (0.46)
$\alpha = 0.10$	**3**	**0.830**	**0.005 (0.01)**	**[0.20 (0.11), 0.50 (0.15), 0.30 (0.11)]**	**[10.45 (1.50), 5.20 (0.77), -0.02 (0.51)]**	**10.38 (1.38)**	**-6.32 (0.91)**
	4	0.158	0.066 (0.05)	[0.19 (0.28), 0.34 (1.46), 0.22 (1.21), 0.25 (0.76)]	[11.43 (1.94), 6.47 (1.54), 3.44 (1.79), -0.51 (1.03)]	11.23 (1.92)	-6.81 (1.18)
	2	0.022	0.001 (0.00)	[0.70 (0.00), 0.30 (0.00)]	[4.49 (0.57), 0.08 (0.44)]	7.21 (0.93)	-4.22 (0.59)
$t = 0.50$	**3**	**0.704**	**0.005 (0.01)**	**[0.20 (0.09), 0.50 (0.16), 0.30 (0.13)]**	**[10.58 (1.72), 5.28 (0.87), -0.01 (0.54)]**	**10.57 (1.56)**	**-6.44 (0.98)**
	4	0.230	0.100 (0.07)	[0.19 (0.29), 0.36 (1.63), 0.22 (1.34), 0.23 (0.82)]	[10.91 (1.72), 6.06 (1.36), 3.06 (1.91), -0.50 (0.87)]	10.73 (1.58)	-6.56 (1.05)
	2	0.058	0.005 (0.01)	[0.68 (0.93), 0.32 (0.93)]	[4.57 (0.55), 0.22 (0.45)]	7.13 (0.68)	-4.21 (0.40)
$t = 0.75$	**3**	**0.754**	**0.005 (0.02)**	**[0.20 (0.41), 0.50 (0.31), 0.30 (0.26)]**	**[10.45 (1.67), 5.22 (0.89), -0.01 (0.57)]**	**10.48 (1.49)**	**-6.39 (0.96)**
	4	0.174	0.099 (0.06)	[0.20 (0.19), 0.38 (1.39), 0.20 (0.99), 0.22 (0.84)]	[10.84 (1.81), 5.81 (1.50), 2.73 (2.15), -0.74 (0.83)]	10.79 (1.90)	-6.58 (1.20)
	2	0.178	0.006 (0.02)	[0.63 (1.57), 0.37 (1.57)]	[4.93 (1.03), 0.34 (0.78)]	7.33 (0.85)	-4.38 (0.59)
$t = 1.00$	**3**	**0.760**	**0.007 (0.02)**	**[0.21 (0.71), 0.49 (0.54), 0.30 (0.28)]**	**[10.29 (1.81), 5.09 (1.04), -0.03 (0.54)]**	**10.33 (1.59)**	**-6.29 (1.04)**
	4	0.060	0.088 (0.05)	[0.20 (0.18), 0.40 (1.38), 0.21 (0.89), 0.19 (0.85)]	[11.46 (1.79), 6.09 (1.06), 2.30 (2.22), -1.33 (1.15)]	11.48 (1.66)	-7.00 (1.04)
	2	0.368	0.007 (0.02)	[0.66 (1.29), 0.34 (1.29)]	[4.78 (1.01), 0.28 (0.75)]	7.27 (0.81)	-4.32 (0.54)
$t = 1.25$	**3**	**0.614**	**0.005 (0.01)**	**[0.20 (0.43), 0.50 (0.37), 0.30 (0.11)]**	**[10.51 (1.74), 5.26 (0.93), -0.03 (0.52)]**	**10.50 (1.60)**	**-6.39 (1.03)**
	4	0.018	0.068 (0.04)	[0.19 (0.31), 0.40 (1.41), 0.20 (1.33), 0.21 (0.74)]	[13.57 (3.51), 7.25 (2.12), 2.65 (2.55), -1.49 (1.25)]	12.93 (2.53)	-7.96 (1.55)
	2	0.574	0.008 (0.02)	[0.63 (1.61), 0.37 (1.61)]	[4.93 (1.10), 0.39 (0.84)]	7.29 (0.78)	-4.35 (0.53)
$t = 1.50$	**3**	**0.420**	**0.007 (0.02)**	**[0.20 (0.27), 0.49 (0.29), 0.30 (0.19)]**	**[10.44 (1.52), 5.23 (0.77), -0.09 (0.48)]**	**10.38 (1.31)**	**-6.30 (0.84)**
	4	0.006	0.110 (0.07)	[0.20 (0.00), 0.37 (0.94), 0.20 (0.00), 0.23 (0.94)]	[11.63 (1.32), 6.70 (0.35), 3.05 (3.10), -1.96 (2.20)]	12.05 (1.12)	-7.24 (0.69)
TV				[0.2, 0.5, 0.3]	[10, 5, 0]	10	−6

Notes: For each case, 500 runs were performed and results are reported for the cases in which the algorithm identifies 2, 3 and 4 clusters. Estimates of entropy, weights, random and fixed coefficients are reported in terms of mean (sd). True Values (TV) of the coefficients used to simulate data are reported under the relative estimates. Results related to the true number of clusters (i.e. 3) are reported in bold. The cases for which the algorithm identifies 1 or more than 4 clusters are not reported in table, but can be identified by complementing with 1 the sum of the three reported Proportions.

Table S7.6: Results obtained by SPGLMM algorithm for the **Bernoulli response** through DGP(iii) in Eq. (S5.3), for $t = 0.50, 0.75, 1.00, 1.25, 1.50$, $\alpha = 0.01, 0.05, 0.10$, with **1 fixed slope**.

	N clusters	Proportion (out of 500)	Entropy(sd)	$\hat{\omega}$(sd)	$\hat{c}_1$(sd)	$\hat{c}_2$(sd)	$\hat{\beta}_1$(sd)
	2	0.016	0.000 (0.00)	[0.70 (0.00), 0.30 (0.00)]	[2.14 (0.11), -7.86 (0.49)]	[4.79 (0.26), -0.09 (0.53)]	-4.68 (0.26)
$\alpha = 0.01$	**3**	**0.664**	**0.005 (0.02)**	**[0.21 (0.56), 0.49 (0.41), 0.3 (0.19)]**	**[5.24 (0.93), 2.02 (0.98), -10.22 (1.20)]**	**[10.57 (1.73), 5.22 (0.82), -0.04 (0.54)]**	**-6.28 (0.73)**
	4	0.268	0.054 (0.04)	[0.19 (0.31), 0.40 (1.48), 0.20 (1.17), 0.2 (0.88)]	[5.56 (1.13), 2.49 (0.78), -5.29 (5.85), -10.99 (1.76)]	[11.04 (1.95), 5.64 (1.54), 2.22 (2.49), 0.14 (1.41)]	-6.41 (0.88)
	2	0.032	0.000 (0.00)	[0.70 (0.00), 0.30 (0.00)]	[2.25 (0.28), -8.15 (0.66)]	[5.02 (0.83), 0.11 (0.50)]	-5.02 (0.52)
$\alpha = 0.05$	**3**	**0.602**	**0.003 (0.01)**	**[0.20 (0.48), 0.50 (0.39), 0.3 (0.13)]**	**[5.30 (0.96), 2.08 (0.83), -10.37 (1.28)]**	**[10.71 (1.89), 5.27 (0.80), -0.02 (0.48)]**	**-6.41 (0.82)**
	4	0.316	0.061 (0.06)	[0.20 (0.18), 0.38 (1.58), 0.21 (1.14), 0.21 (0.9)]	[5.44 (0.94), 2.40 (0.64), -4.57 (5.94), -10.99 (1.40)]	[10.92 (1.41), 5.69 (1.53), 2.51 (2.91), 0.08 (1.19)]	-6.53 (0.72)
	2	0.002	0.000 (0.00)	[0.70 (0.00), 0.30 (0.00)]	[2.11 (0.28), -8.14 (0.93)]	[4.72 (0.41), -0.16 (0.23)]	-4.92 (0.58)
$\alpha = 0.10$	**3**	**0.584**	**0.003 (0.01)**	**[0.20 (0.20), 0.50 (0.20), 0.30 (0.00)]**	**[5.33 (0.90), 2.14 (0.36), -10.39 (1.25)]**	**[10.62 (1.59), 5.36 (0.69), -0.07 (0.53)]**	**-6.40 (0.77)**
	4	0.332	0.063 (0.05)	[0.19 (0.29), 0.39 (1.51), 0.21 (1.18), 0.21 (0.89)]	[5.56 (1.04), 2.53 (0.78), -4.69 (5.87), -10.95 (1.37)]	[10.99 (1.73), 5.73 (1.31), 2.31 (2.92), 0.09 (1.34)]	-6.51 (0.67)
	2	0.014	0.000 (0.00)	[0.70 (0.00), 0.30 (0.00)]	[2.10 (0.18), -8.01 (0.76)]	[4.89 (0.37), 0.03 (0.45)]	-4.86 (0.52)
$t = 0.50$	**3**	**0.548**	**0.003 (0.01)**	**[0.20 (0.13), 0.50 (0.13), 0.30 (0.00)]**	**[5.25 (0.75), 2.09 (0.32), -10.11 (1.07)]**	**[10.48 (1.34), 5.19 (0.62), -0.01 (0.52)]**	**-6.24 (0.69)**
	4	0.358	0.063 (0.06)	[0.19 (0.32), 0.35 (1.76), 0.24 (1.33), 0.22 (0.83)]	[5.65 (1.10), 2.58 (0.79), -4.56 (6.10), -10.97 (1.73)]	[11.13 (1.82), 5.70 (1.67), 2.55 (2.79), -0.08 (1.33)]	-6.48 (0.96)
	2	0.022	0.000 (0.00)	[0.70 (0.00), 0.30 (0.00)]	[2.01 (0.29), -7.71 (0.52)]	[4.72 (0.33), -0.09 (0.28)	-4.56 (0.49)
$t = 0.75$	**3**	**0.572**	**0.001 (0.00)**	**[0.20 (0.17), 0.50 (0.17), 0.30 (0.00)]**	**[5.37 (0.83), 2.09 (0.32), -10.28 (1.18)]**	**[10.82 (1.60), 5.28 (0.62), 0.01 (0.49)]**	**-6.36 (0.74)**
	4	0.346	0.051 (0.04)	[0.19 (0.33), 0.39 (1.66), 0.22 (1.29), 0.20 (0.86)]	[5.68 (0.99), 2.63 (0.97), -5.43 (5.82), -11.15 (1.52)]	[11.11 (2.03), 6.03 (1.88), 2.11 (2.87), -0.13 (1.30)]	-6.55 (0.74)
	2	0.005	0.000 (0.00)	[0.70 (0.00), 0.30 (0.00)]	[2.16 (0.27), -8.56 (0.94)]	[5.14 (0.60), -0.06 (0.44)]	-5.19 (0.58)
$t = 1.00$	**3**	**0.700**	**0.003 (0.01)**	**[0.21 (0.66), 0.49 (0.48), 0.30 (0.21)]**	**[5.16 (0.91), 1.90 (1.40), -10.17 (1.15)]**	**[10.40 (1.64), 5.16 (0.92), 0.01 (0.52)]**	**-6.27 (0.72)**
	4	0.232	0.055 (0.04)	[0.19 (0.28), 0.39 (1.56), 0.20 (1.17), 0.21 (0.83)]	[5.71 (1.17), 2.47 (0.80), -5.21 (5.84), -11.05 (1.23)]	[11.36 (2.20), 5.86 (1.65), 2.13 (2.74), 0.01 (1.28)]	-6.54 (0.76)
	2	0.076	0.000 (0.00)	[0.70 (0.00), 0.30 (0.00)]	[2.14 (0.27), -8.22 (0.61)]	[4.85 (0.38), -0.06 (0.48)]	-4.96 (0.36)
$t = 1.25$	**3**	**0.708**	**0.004 (0.01)**	**[0.21 (0.75), 0.49 (0.59), 0.30 (0.20)]**	**[5.26 (0.96), 1.92 (1.38), -10.25 (1.19)]**	**[10.46 (1.68), 5.15 (0.95), 0.03 (0.47)]**	**-6.31 (0.72)**
	4	0.202	0.037 (0.03)	[0.19 (0.41), 0.37 (1.69), 0.23 (1.38), 0.21 (0.87)]	[5.80 (1.28), 2.70 (1.06), -4.81 (5.94), -11.22 (1.44)]	[11.52 (2.35), 6.14 (2.14), 2.45 (2.89), 0.30 (1.40)]	-6.56 (0.79)
	2	0.126	0.000 (0.00)	[0.70 (0.00), 0.30 (0.00)]	[2.08 (0.25), -7.98 (0.85)]	[4.80 (0.48), -0.01 (0.39)]	-4.75 (0.49)
$t = 1.50$	**3**	**0.746**	**0.003 (0.01)**	**[0.21 (0.85), 0.49 (0.68), 0.30 (0.23)]**	**[5.24 (1.01), 1.83 (1.64), -10.18 (1.19)]**	**[10.56 (1.98), 5.12 (0.96), -0.04 (0.62)]**	**-6.26 (0.73)**
	4	0.114	0.038 (0.03)	[0.20 (0.86), 0.38 (1.67), 0.22 (1.39), 0.20 (0.85)]	[5.79 (1.38), 2.65 (1.15), -5.48 (5.69), -11.02 (1.34)]	[11.15 (2.13), 6.34 (2.57), 1.86 (3.04), 0.24 (1.51)]	-6.49 (0.60)
TV				[0.2, 0.5, 0.3]	[5, 2, −10]	[10, 5, 0]	−6

Notes: For each case, 500 runs were performed and results are reported for the cases in which the algorithm identifies 2, 3 and 4 clusters. Estimates of entropy, weights, random and fixed coefficients are reported in terms of mean (sd). True Values (TV) of the coefficients used to simulate data are reported under the relative estimates. Results related to the true number of clusters (i.e. 3) are reported in bold. The cases for which the algorithm identifies 1 or more than 4 clusters are not reported in table, but can be identified by complementing with 1 the sum of the three reported Proportions.

References

Aitkin, M. (1999). A general maximum likelihood analysis of variance components in generalized linear models. *Biometrics 55*(1), 117–128.

Azzimonti, L., F. Ieva, and A. M. Paganoni (2013). Nonlinear nonparametric mixed-effects models for unsupervised classification. *Computational Statistics 28*(4), 1549–1570.

Bates, D., M. Mächler, B. Bolker, and S. Walker (2015). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software 67*(1), 1–48.

Bock, R. D. and M. Aitkin (1981). Marginal maximum likelihood estimation of item parameters: Application of an em algorithm. *Psychometrika 46*(4), 443–459.

Breslow, N. E. and D. G. Clayton (1993). Approximate inference in generalized linear mixed models. *Journal of the American statistical Association 88*(421), 9–25.

Casella, G. and R. L. Berger (2021). *Statistical inference*. Cengage Learning.

Dempster, A. P., N. M. Laird, and D. B. Rubin (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological) 39*(1), 1–22.

Evered, L. J. (1990). Innumeracy: Mathematical illiteracy and its consequences. by john allen paulos. *The American Mathematical Monthly 97*(1), 88–91.

Gałecki, A., T. Burzykowski, A. Gałecki, and T. Burzykowski (2013). *Linear mixed-effects model*. Springer.

Gilitschenski, I. and U. D. Hanebeck (2012). A robust computational test for overlap of two arbitrary-dimensional ellipsoids in fault-detection of kalman filters. In *2012 15th International Conference on Information Fusion*, pp. 396–401. IEEE.

Johnson, R. and D. Wichern (2002). *Applied multivariate statistical analysis* (5. ed ed.). Upper Saddle River, NJ: Prentice Hall.

King, G. (1998). *Unifying political methodology: The likelihood theory of statistical inference*. University of Michigan Press.

Lindsay, B. G. (1983a). The geometry of mixture likelihoods, part ii: the exponential family. *The Annals of Statistics 11*(3), 783–792.

Lindsay, B. G. (1983b). The Geometry of Mixture Likelihoods: A General Theory. *The Annals of Statistics 11*(1), 86 – 94.

Long, J. S. and J. Freese (2006). *Regression models for categorical dependent variables using Stata*, Volume 7. Stata press.

Masci, C., F. Ieva, T. Agasisti, and A. M. Paganoni (2021). Evaluating class and school effects on the joint student achievements in different subjects: A bivariate semiparametric model with random coefficients. *Computational Statistics 36*, 2337–2377.

Masci, C., F. Ieva, and A. M. Paganoni (2022). Semiparametric multinomial mixed-effects models: A university students profiling tool. *The Annals of Applied Statistics 16*(3), 1608 – 1632.

Masci, C., A. M. Paganoni, and F. Ieva (2019). Semiparametric mixed effects models for unsupervised classification of italian schools. *Journal of the Royal Statistical Society: Series A (Statistics in Society) 182*(4), 1313–1342.

McCullagh, P. and J. Nelder (1983). *Generalized Linear Models*. Monographs on Statistics and Applied Probability. Springer US.

McCullagh, P. and J. Nelder (1989). *Generalized Linear Models* (2nd ed.). Chapman and Hall.

Mohammed, M. A. and J. J. Deeks (2008). In the context of performance monitoring, the caterpillar plot should be mothballed in favor of the funnel plot. *The Annals of thoracic surgery 86*(1), 348.

Nelder, J. A. and R. W. Wedderburn (1972). Generalized linear models. *Journal of the Royal Statistical Society: Series A (General) 135*(3), 370–384.

OECD (2019). Pisa 2018 results (volume i, ii, & iii): Combined executive summary.

Parlett, B. N. (1998). *The symmetric eigenvalue problem*. SIAM.

Pinheiro, J. C. and D. M. Bates (2000). *Mixed-Effects Models in S and S-PLUS*. New York: Springer.

R Core Team (2022). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.

Van Rossum, G. and F. L. Drake (2009). *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace.

Wilk, M. B. and R. Gnanadesikan (1968). Probability plotting methods for the analysis for the analysis of data. *Biometrika 55*(1), 1–17.