Clustering Longitudinal Mixed-type Data

Francesco Amato, Julien Jacques

Clustering Longitudinal Mixed Data

Francesco Amato
francesco.amato@univ-lyon2.fr

and

Julien Jacques
julien.jacques@univ-lyon2.fr

CFE-CMStatistics 2024

16/12/2024

What is multivariate longitudinal mixed-type data?

- **Longitudinal data:** data collected from the same subjects over time.
  - Tracks changes and trends within individuals.
  - Example: Patient health metrics measured monthly.

- **Mixed-type data:** combines different data types such as:
  - **continuous:** Blood pressure, temperature.
  - **categorical binary:** COVID-19 status (yes/no).
  - **categorical ordinal:** Pain level (low, medium, high).
  - **count:** Number of asthma attacks experienced.

Clustering longitudinal mixed-type data 🤔

In the **literature** generally:

- mixed variables
  $\Longrightarrow$ assumed to be independent conditionally to the cluster belonging.
- repeated measurements along time
  $\Longrightarrow$ treated through random-effect models (heavy parametrization) or through state/cluster changes in time (ex. HMM).

What we want is a model that is able:

- to detect covariance structures among variables
- to cluster units with similar evolution in time
- to be easy to interpret by non-statisticians

From longitudinal data to three-way data

Longitudinal mixed-type data y_{ijt}:
J different variables measured T times for N units.

From longitudinal data to three-way data

Longitudinal mixed-type data y_{ijt}:
J different variables measured T times for N units.
We organize our data in a random-matrix form such that:

$$Y_i = \begin{pmatrix} y_{i,1,1} & \cdots & y_{i,1,t} & \cdots & y_{i,1,T} \\ \vdots & \ddots & \vdots & \cdots & \vdots \\ y_{i,j,1} & \cdots & y_{i,j,t} & \cdots & y_{i,j,T} \\ \vdots & \cdots & \vdots & \ddots & \vdots \\ y_{i,J,1} & \cdots & y_{i,J,t} & \cdots & y_{i,J,T} \end{pmatrix}$$

The matrix becomes our statistical units! Each matrix is an "observation".

Matrix-variate Normal 🔢

Let $Z \sim \mathcal{MN}_{(J\times T)}(M, \Phi, \Sigma)$, where:

- $M \in \mathbb{R}^{J\times T}$ matrix of means
- $\Phi \in \mathbb{R}^{T\times T}$ covariances among the T times
- $\Sigma \in \mathbb{R}^{J\times J}$ covariances among the J variables

The p.d.f. $\mathcal{MN}_{(J\times T)}(Z|M, \Phi, \Sigma)$ is:

$$(2\pi)^{-\frac{TJ}{2}}|\Phi|^{-\frac{J}{2}}|\Sigma|^{-\frac{T}{2}}\exp\left\{-\frac{1}{2}\text{tr}[\Sigma^{-1}(Z-M)\Phi^{-1}(Z-M)^{\intercal}]\right\}$$

Interpretability

- Φ express the time covariance
- Σ express the covariance among variables

Matrix-variate Normal and Multivariate Normal

The matrix-variate normal distribution is a special case of the multivariate normal distribution:

$$Z \sim \mathcal{MN}_{(J\times T)}(M, \Phi, \Sigma) \iff \text{vec}(Z) \sim \mathcal{MN}_{JT}(\text{vec}(M), \Sigma \otimes \Phi),$$

where vec(.) and $\otimes$ is the Kronecker product.

Properties

$\implies$ the multivariate and the matrix-variate normal distributions share the same properties (Gupta et al., 2000).

Parsimony

- $\Sigma \otimes \Phi$ has $J(J+1)/2 + T(T+1)/2$ parameters
- full cov. matrix of size JT has $JT(JT+1)/2$ parameters
- ex: $J = T = 5$: 30 versus 325 parameters

Mixture of Matrix-Normals 🥗

For **heterogeneous** data sets of matrix-variate data $\Longrightarrow$ **Mixture of Matrix-Normals** (MMN) (Viroli, 2011) distribution:

$$f(Z_i|\boldsymbol{\pi}, \boldsymbol{\Theta}) = \sum_{k=1}^{K} \pi_k \mathcal{MN}_{(J\times T)}(Z_i|M_k, \Phi_k, \Sigma_k),$$

where:

- K : number of mixture components
- $\boldsymbol{\pi} = \{\pi_k\}_{k=1}^K$: vector of mixing proportions, $\sum_{k=1}^K \pi_k = 1$
- $\{\Theta_k\}_{k=1}^K$ set of component-specific parameters $\Theta_k = \{M_k, \Phi_k, \Sigma_k\}$

Idea 💡

- We can use **mixture of matrix-variate normal distributions** to account for the longitudinal structure!
  $\Longrightarrow$ easy to understand and interpret!

- How can we extend it to account for **mixed data**? 🤔
  $\Longrightarrow$ Inspired by `clustMD` (McParland et al., 2016) we assume that **each observed variable** is a **realizations** of a **continuous latent variable**.

- Developed for ordinal data in Amato et al., 2024 🤗

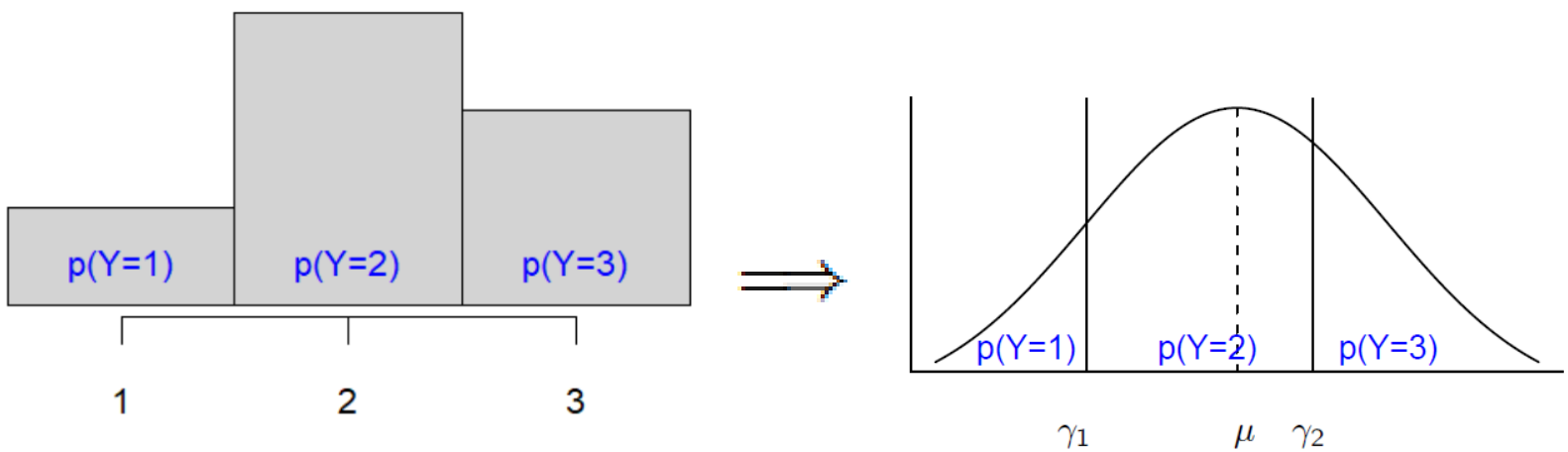
The Mixture of Mixed Matrices (MMM) Model®

- **Continuous variables** equal the latent ones. Let c be the c-th generic continuous variable.

$$y_{ict} = z_{ict}$$

- **Categorical ordinal variables** follow Amato et al., 2024. Let o be the o-th generic ordinal variable with C_o levels.

$$\gamma_{o,c_o-1} < z_{iot} < \gamma_{o,c_o} \implies y_{iot} = c_o$$



The Mixture of Mixed Matrices (MMM) Model®

- **Categorical binary variables** are considered as a special case of ordinal ones with 2 levels. The underlying threshold is set at 0.

- **Categorical nominal variables** with P levels are transformed as one-hot encoder for $P - 1$ levels and treat them as binary variables.

- **Count data** follow a matrix-variate Poisson log-normal Silva et al., 2023. Let g be the g-th generic count variables.

$$y_{igt} \sim \mathcal{P}(\exp(z_{igt}))$$

z_{igt} is a term of the latent matrix $G \times T$ that follows a matrix-variate normal.

Joint model

$Y_i = [Y_i^\alpha, Y_i^\beta, Y_i^\gamma]^\intercal$ is a block matrix:

- $Y_i^\alpha \in \mathbb{R}^{C \times T}$ contains the **continuous** variables
- $Y_i^\beta \in \mathbb{N}^{O \times T}$ collects the **categorical** ones (coded via integers)
- $Y_i^\gamma \in \mathbb{N}^{G \times T}$ is the block containing the **count** variables

$$\begin{pmatrix} \mathbb{R}^{C\times T} \\ \mathbb{N}^{O\times T} \\ \mathbb{N}^{G\times T} \end{pmatrix} \ni Y_i = \begin{pmatrix} y_{i,1,1} & \cdots & y_{i,1,t} & \cdots & y_{i,1,T} \\ \vdots & \ddots & \vdots & \cdots & \vdots \\ y_{i,C,1} & \cdots & y_{i,C,t} & \cdots & y_{i,C,T} \\ y_{i,C+1,1} & \cdots & y_{i,C+1,t} & \cdots & y_{i,C+1,T} \\ \vdots & \cdots & \vdots & \ddots & \vdots \\ y_{i,O,1} & \cdots & y_{i,O,t} & \cdots & y_{i,O,T} \\ y_{i,O+1,1} & \cdots & y_{i,O+1,t} & \cdots & y_{i,O+1,T} \\ \vdots & \cdots & \vdots & \ddots & \vdots \\ y_{i,G,1} & \cdots & y_{i,G,t} & \cdots & y_{i,G,T} \end{pmatrix} \longleftarrow \begin{pmatrix} z_{i,1,1} & \cdots & z_{i,1,t} & \cdots & z_{i,1,T} \\ \vdots & \ddots & \vdots & \cdots & \vdots \\ z_{i,C,1} & \cdots & z_{i,C,t} & \cdots & z_{i,C,T} \\ z_{i,C+1,1} & \cdots & z_{i,C+1,t} & \cdots & z_{i,C+1,T} \\ \vdots & \cdots & \vdots & \ddots & \vdots \\ z_{i,O,1} & \cdots & z_{i,O,t} & \cdots & z_{i,O,T} \\ z_{i,O+1,1} & \cdots & z_{i,O+1,t} & \cdots & z_{i,O+1,T} \\ \vdots & \cdots & \vdots & \ddots & \vdots \\ z_{i,G,1} & \cdots & z_{i,G,t} & \cdots & z_{i,G,T} \end{pmatrix} = Z_i \in \mathbb{R}^{J\times T}$$

Joint model 🧩

Each block follow a matrix-variate normal distribution. However, **each latent block is related differently to the respective observed block**.

Joint model 🧩

Each block follow a matrix-variate normal distribution. However, **each latent block is related differently to the respective observed block**.

$\Longrightarrow$ To link the blocks' distributions, we resort to **condition one block to another**, such that $p(Y_i) = p(Y_i^{\alpha}) \cdot p(Y_i^{\beta|\alpha}) \cdot p(Y_i^{\gamma|\beta,\alpha})$.

Joint model 🧩

Each block follow a matrix-variate normal distribution. However, **each latent block is related differently to the respective observed block**.

$\Longrightarrow$ To link the blocks' distributions, we resort to **condition one block to another**, such that $p(Y_i) = p(Y_i^{\alpha}) \cdot p(Y_i^{\beta|\alpha}) \cdot p(Y_i^{\gamma|\beta,\alpha})$.

$$Y_i \sim \sum_{k=1}^{K} \pi_k \, \mathcal{MN}_{(C\times T)}(Z_i^{\alpha}|\Theta_k^{\alpha}) \cdot \int_{\Omega_r} \mathcal{MN}_{(O\times T)}(Z_i^{\beta}|\Theta_k^{\beta|\alpha}) dZ_i^{\beta}$$

$$\cdot \int_{\mathbb{R}} \prod_{t}^{T} \prod_{g}^{G} \mathcal{P}(y_{igt}^{\gamma}|\exp(z_{igt}^{\gamma})) \times \mathcal{MN}_{(G\times T)}(Z_i^{\gamma}|\Theta_k^{\gamma|\alpha,\beta}) dZ_i^{\gamma}$$

where $\Theta_k^{\gamma|\alpha,\beta} := \{M_k^{\gamma|\alpha,\beta}, \Phi_k, \Sigma_k^{\gamma|\alpha,\beta}\}$, $\Theta_k^{\beta|\alpha} := \{M_k^{\beta|\alpha}, \Phi_k, \Sigma_k^{\beta|\alpha}\}$.

Conditioning is easy thanks to properties 😉

$$M_k = \begin{pmatrix} M_k^\alpha \\ M_k^\beta \\ M_k^\gamma \end{pmatrix} \; ; \; \Sigma_k = \begin{pmatrix} \Sigma_k^{\alpha\alpha} & \Sigma_k^{\alpha\beta} & \Sigma_k^{\alpha\gamma} \\ \Sigma_k^{\beta\alpha} & \Sigma_k^{\beta\beta} & \Sigma_k^{\beta\gamma} \\ \Sigma_k^{\gamma\alpha} & \Sigma_k^{\gamma\beta} & \Sigma_k^{\gamma\gamma} \end{pmatrix}$$

$\Theta_k^{\beta|\alpha} := \{M_k^{\beta|\alpha}, \Phi_k, \Sigma_k^{\beta|\alpha}\}$:

- $M_k^{\beta|\alpha} = M_k^\beta + \Sigma_{k,\beta\alpha}\Sigma_{k,\alpha\alpha}^{-1}(Y_i^\alpha - M_k^\alpha)$
- $\Sigma_k^{\beta|\alpha} = \Sigma_{k,\beta\beta} - \Sigma_{k,\beta\alpha}\Sigma_{k,\alpha\alpha}^{-1}\Sigma_{k,\alpha\beta}$.

$\Theta_k^{\gamma|\alpha,\beta} := \{M_k^{\gamma|\alpha,\beta}, \Phi_k, \Sigma_k^{\gamma|\alpha,\beta}\}$:

- $M_k^{\gamma|\alpha,\beta} = M_k^\gamma + \Sigma_{k,\gamma\cdot}\Sigma_{k,\cdot\cdot}^{-1}(Z_i^{\alpha,\beta} - M_k^{\alpha,\beta})$
- $\Sigma_k^{\gamma|\alpha,\beta} = \Sigma_{k,\gamma\gamma} - \Sigma_{k,\gamma\cdot}\Sigma_{k,\cdot\cdot}^{-1}\Sigma_{k,\cdot\gamma}$

Inference: EM algorithm ✍️

Starting from an initialization $\boldsymbol{\Theta}^{(0)}$, The **EM algorithm is an iterative algorithm** that alternates

- E step (*Expectation*): computes

$$\mathcal{Q}(\boldsymbol{\Theta}, \hat{\boldsymbol{\Theta}}^{(s)}) := \mathbb{E}(\log \mathcal{L}_C(\boldsymbol{\Theta}; \mathbf{Y}, \mathbf{Z}, \boldsymbol{\ell}) | \hat{\boldsymbol{\Theta}}^{(s)}, \mathbf{Y})$$

- M step (*Maximisation*): computes

$$\hat{\boldsymbol{\Theta}}^{(s+1)} = \arg\max_{\boldsymbol{\Theta}} \mathcal{Q}(\boldsymbol{\Theta}, \hat{\boldsymbol{\Theta}}^{(s)})$$

until convergence on the observed log-likelihood.

Inference: EM algorithm - E step

In the E-step, we need to compute:

$\mathbb{E}(Z_i^{\beta}|\Theta^{(s),\beta|\alpha})$ and $\mathbb{E}(Z_i^{\beta}Z_i^{\beta\intercal}|\Theta^{(s),\beta|\alpha})$, the first two moments of a truncated matrix-variate distribution.

$\Longrightarrow$ Vectorization of Z_i and use of a **Gibbs sampler**.

$\mathbb{E}(Z_i^{\gamma}|\Theta^{(s),\gamma|\alpha,\beta})$ and $\mathbb{E}(Z_i^{\gamma}Z_i^{\gamma\intercal}|\Theta^{(s),\gamma|\alpha,\beta})$ the first two moments of a "prior" matrix-variate distribution.

$\Longrightarrow$ More powerful samplers used in Bayesian statistics, such as **No-U-Turn sampler** implemented in `stan` (Stan Development Team, 2024).

Numerical study on simulated data 💻

Simulation setup:

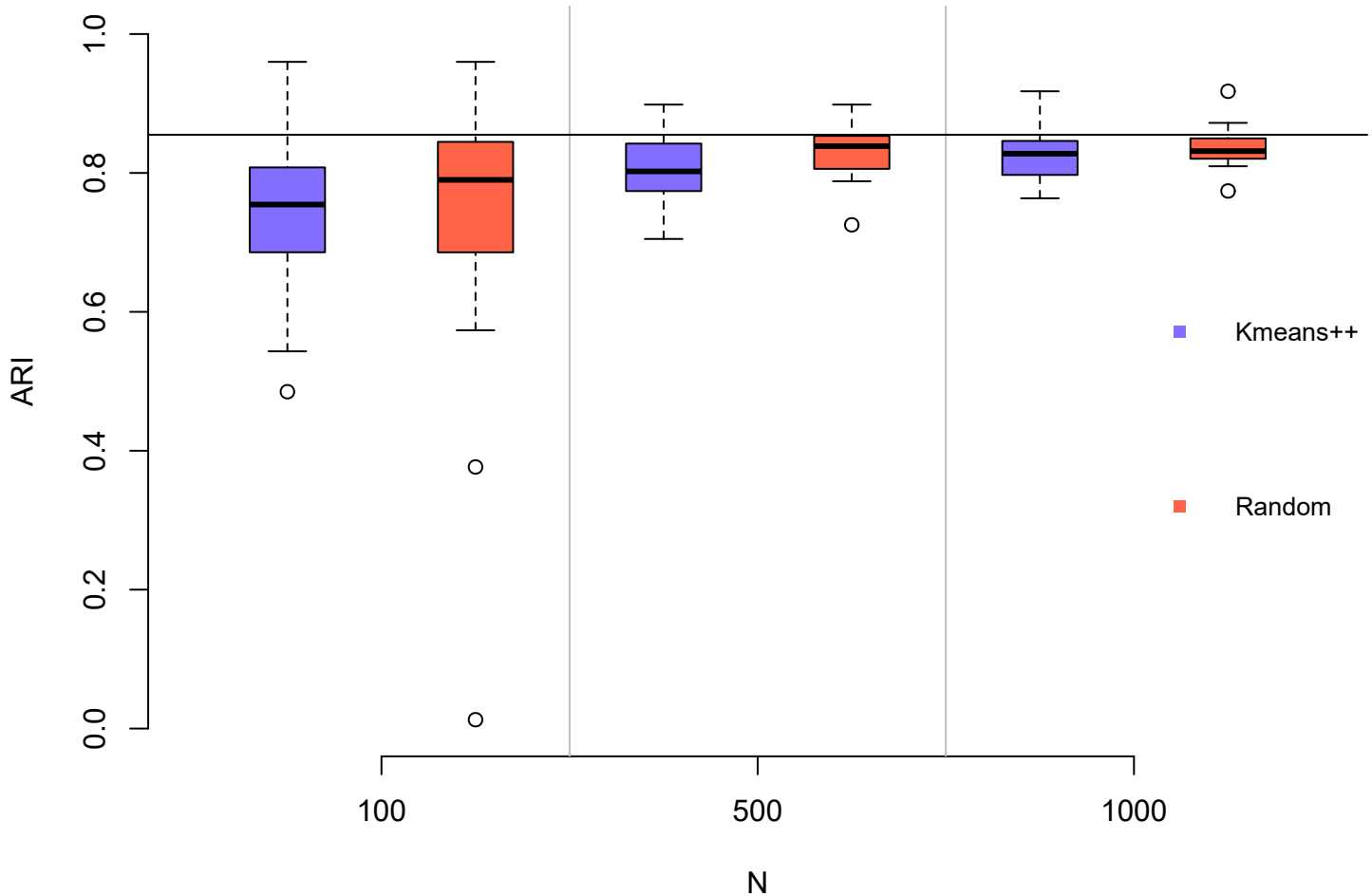
- 20 data sets simulated according to the MMM model:
  - $K = 2$, $J = 4, T = 3$, $\pi = (.4, .6)$
  - One continuous, one ordinal, one binary and one count variable.
- $N \in \{100, 500, 1000\}$
- in each data set, a proportion of noise $\tau \in \{0, 0.1, 0.2\}$ is added using a $\mathcal{N}(0, 0.5)$.

Goals:

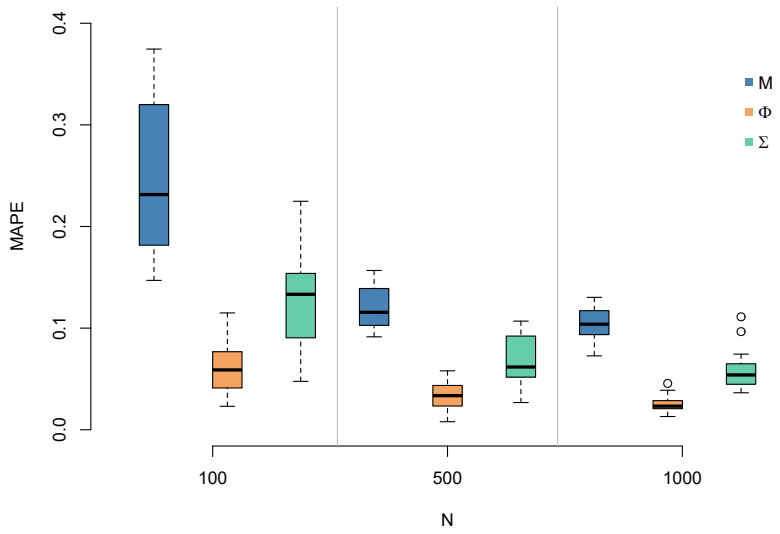
- compare the different initialization strategies
- check that parameter estimation is consistent with N
- robustness to noise
- evaluate the efficiency of BIC to choose K
- comparison with competitors (*continuous* model)

Compare the different initialization strategies

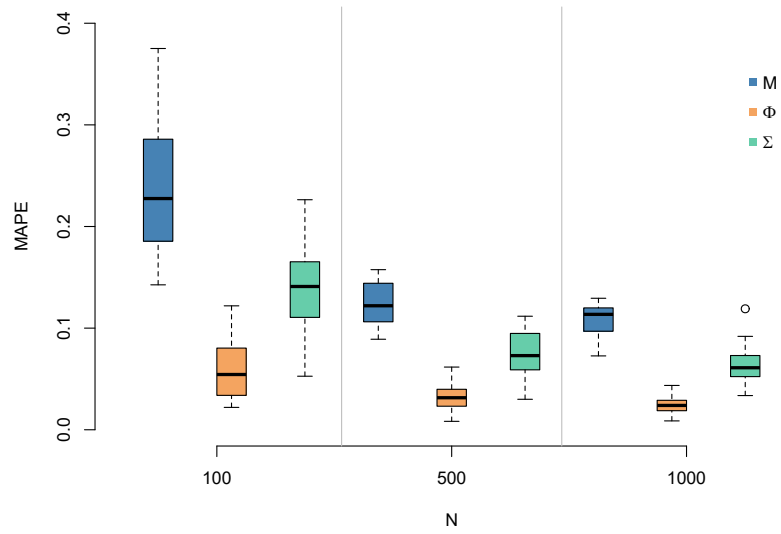
The horizontal line represents the estimated optimal ARI.



Check that parameter estimation is consistent with N

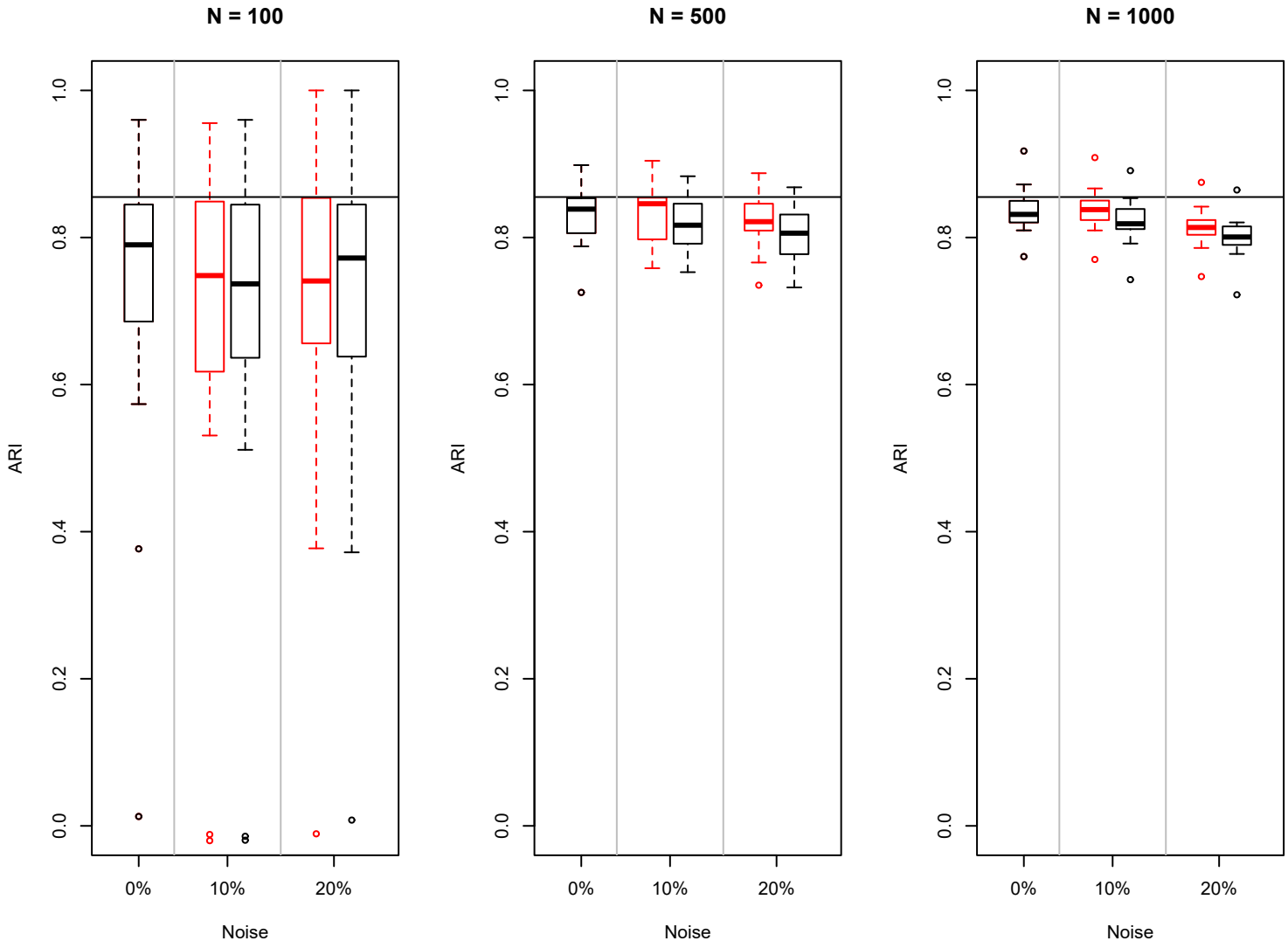


K-means



Random

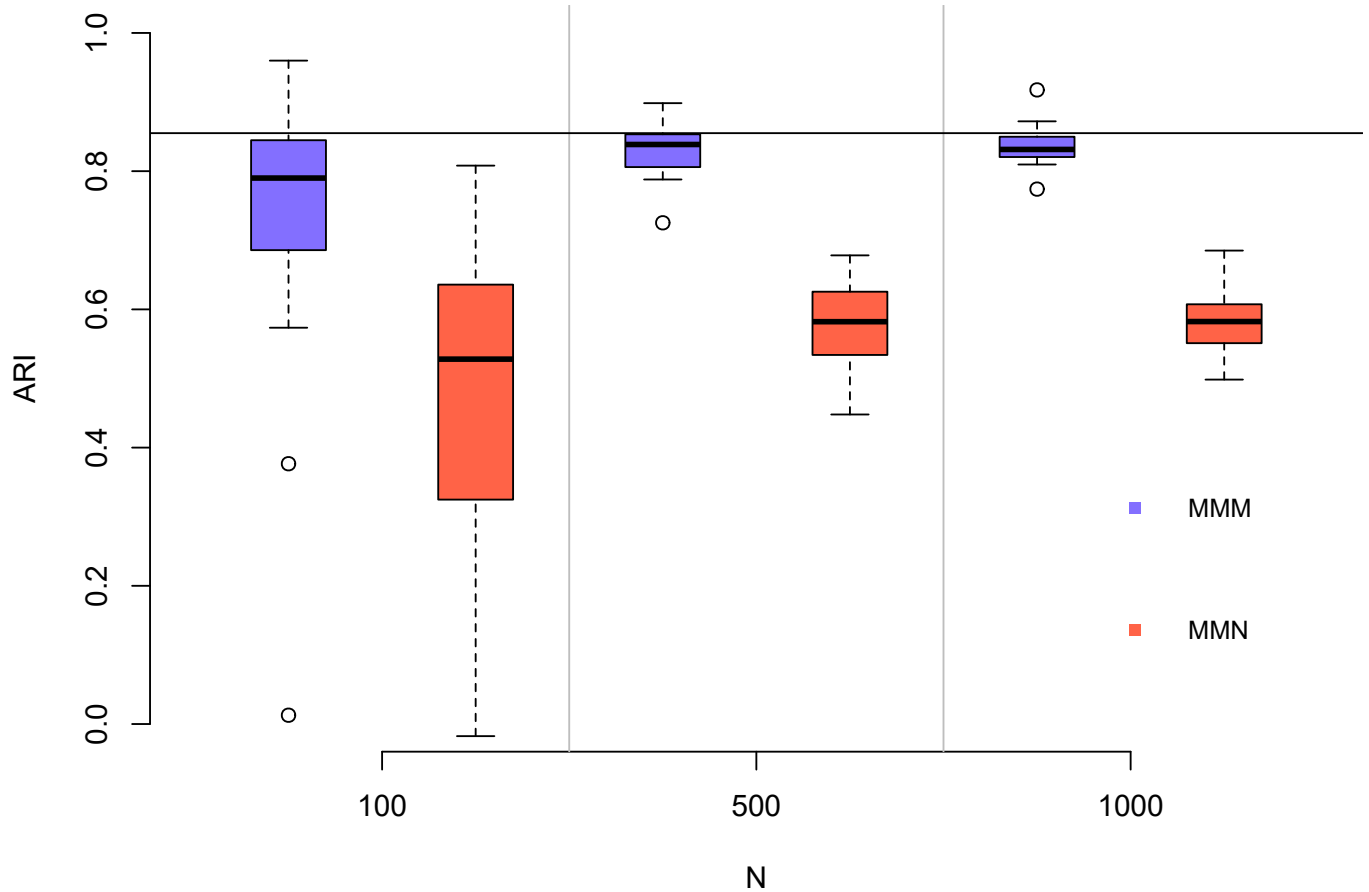
Robustness to noise



Evaluate the efficiency of BIC to choose K

N/K	Scenario $\tau = 0$				Scenario $\tau = 0.1$				Scenario $\tau = 0.2$			
	1	**2**	3	4	1	**2**	3	4	1	**2**	3	4
100	**14**	6	0	0	**13**	7	0	0	**12**	8	0	0
500	0	**19**	1	0	0	**20**	0	0	0	**20**	0	0
1000	0	**17**	3	0	0	**17**	2	1	0	**18**	2	0

Comparison with competitors (*continuous* model)

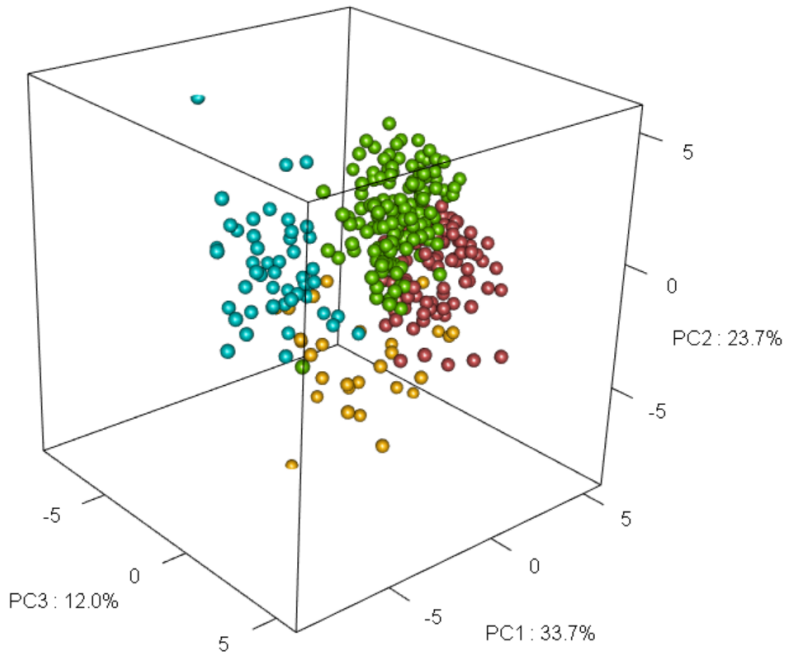


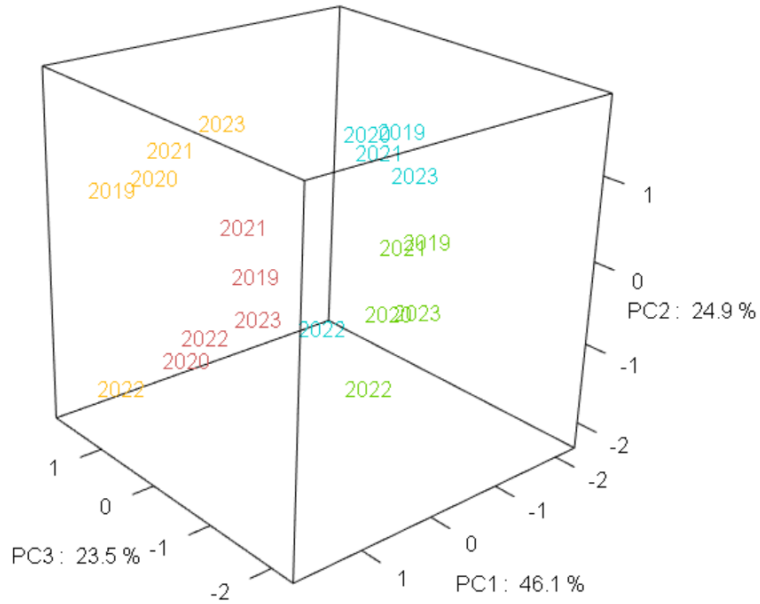
Application on financial data 💱

We collected for years 2019-2023 and for 330 of listed company comprising the S&P500 index:

- **LogReturns:** continuous variable. The logarithm of the yearly return of the stock.
- **Grades:** ordinal variable. The investment grade of the stock expressed by institutional investment banks. The grades have three levels: “Underperform”, “Neutral” and “Buy”.
- **Dividends:** binary variable. Whether the stock gave right to a dividend during the fiscal year or not, regardless of the amount .
- **Volume:** count variable. The total volume of stocks exchanged during the year, expressed as per millions of stocks exchanged. Securities with higher volume are more liquid.

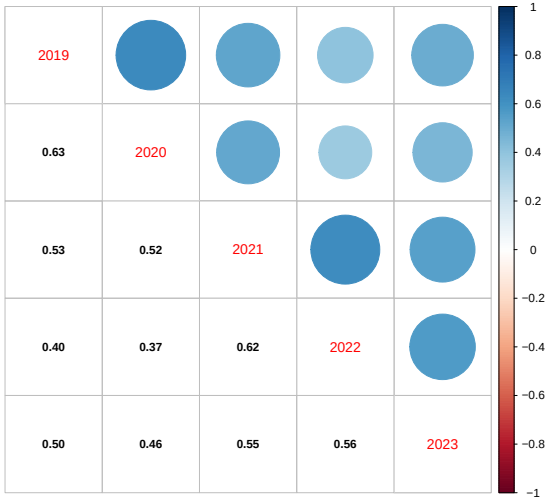
Units and cluster means represented through PCA

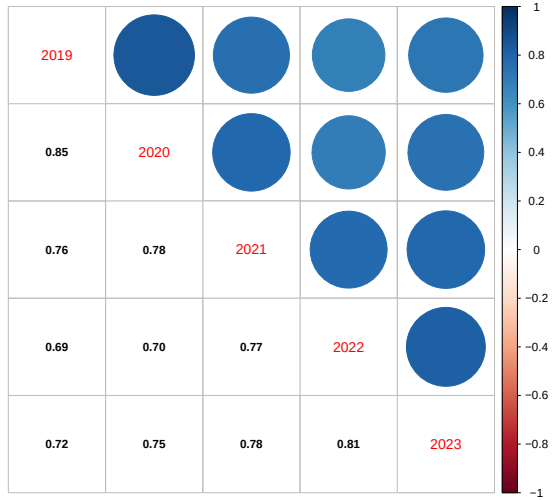


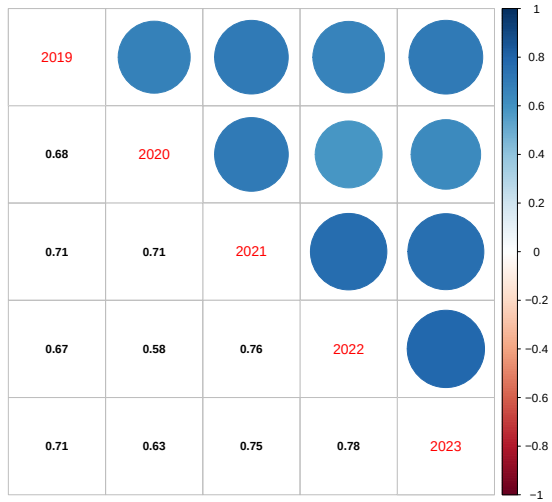


Cluster 1 Cluster 2 Cluster 3 Cluster 4

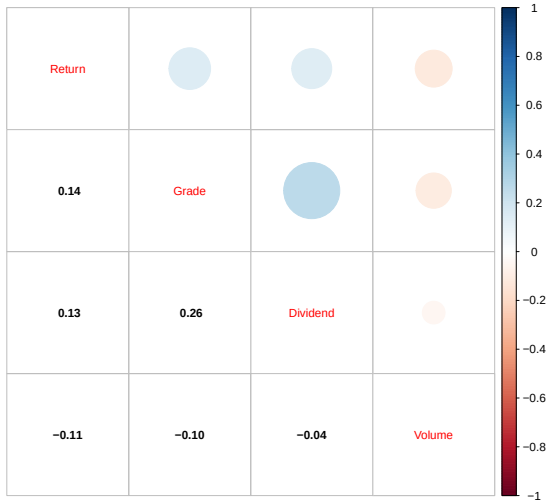
Years correlations

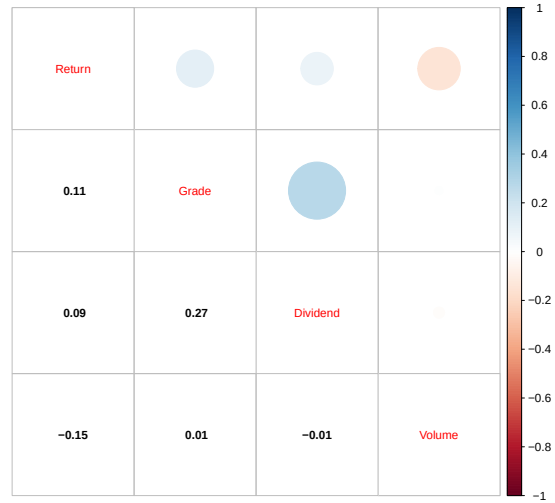


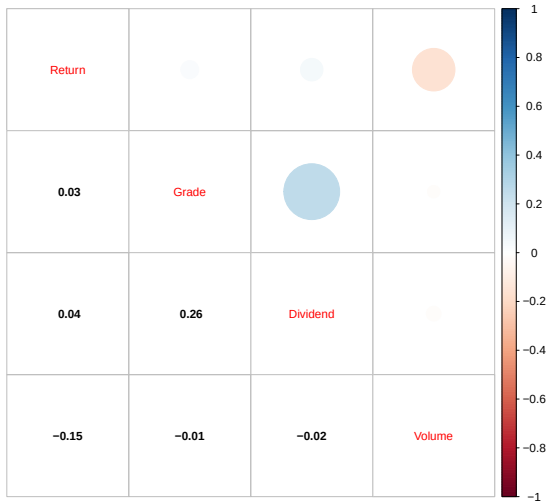


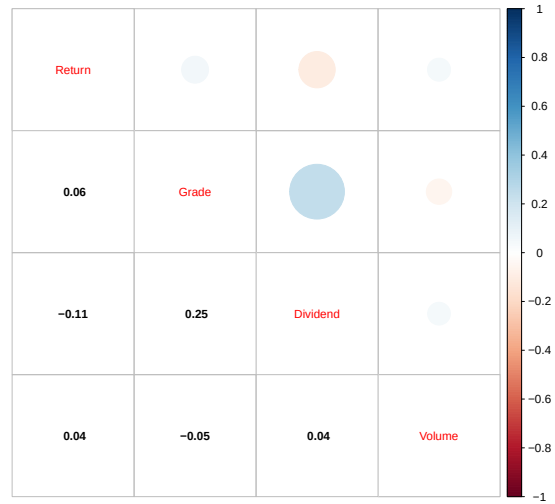


Variables correlations









Conclusions and future works 🧐

Conclusions

- **model-based clustering** for **longitudinal mixed-type data**
- no need for conditional independence
- cluster together similar evolutions
- parsimonious modeling
- good interpretability
- R package in progress 🛠️
- pre-print: https://hal.science/hal-04807626

Future works

- studying more parsimonious models by decomposing the covariance matrices
- missing data

Thanks for your attention! 🤗

[1] Arjun Kumar Gupta and Daya Krishna Nagar. *Matrix Variate Distributions*. Chapman and Hall/CRC, 2000.

[2] Cinzia Viroli. "Finite mixtures of matrix normal distributions for classifying three-way data". In: *Statistics and Computing* 21.4 (Oct. 2011), pp. 511–522. ISSN: 1573-1375. DOI: 10.1007/s11222-010-9188-x.

[3] Damien McParland and Isobel Claire Gormley. "Model based clustering for mixed data: clustMD". In: *Advances in Data Analysis and Classification* 10.2 (June 2016), pp. 155–169. ISSN: 1862-5355. DOI: 10.1007/s11634-016-0238-x.

[4] Anjali Silva et al. "Finite Mixtures of Matrix Variate Poisson-Log Normal Distributions for Three-Way Count Data". In: *Bioinformatics* (Apr. 2023), btad167. ISSN: 1367-4811. DOI: 10.1093/bioinformatics/btad167.

[5] Francesco Amato, Julien Jacques, and Isabelle Prim-Allaz. "Clustering longitudinal ordinal data via finite mixture of matrix-variate distributions". In: *Statistics and Computing*

34.2 (Apr. 2024). ISSN: 1573-1375. DOI: 10.1007/s11222-024-10390-z.

[6] Stan Development Team. *RStan: the R interface to Stan*. R package version 2.32.6. 2024. URL: https://mc-stan.org/.