Università Degli Studi
di Milano-Bicocca

Sparse model-based clustering of three-way data via lasso-type penalties

Andrea Borghesi | Matteo Falcone
916202 865448

Artificial Intelligence For Science and Technology - A.Y. 2024/2025

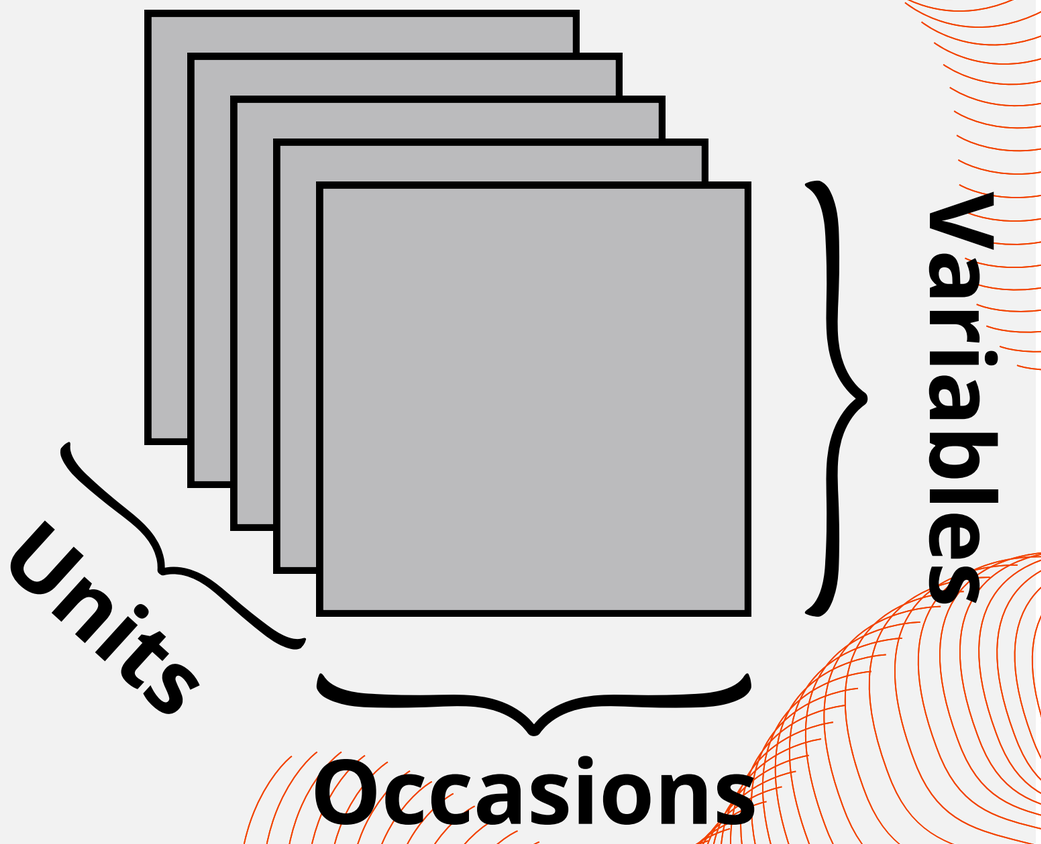
Content

Introduction

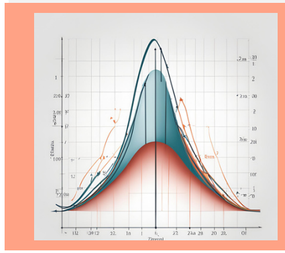
Three-way Data

Dataset composed of n statistical units:

Each unit takes the form of a pxq matrix, which means p different variables measured in q occasions



Introduction



Gaussian Mixture Models

$$f(x_i;\Theta) = \sum_{k=1}^{K} \tau_k \phi_p(x_i;\mu_k,\Sigma_k)$$

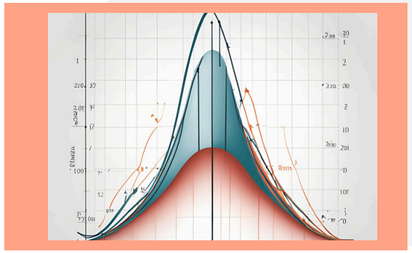
With:

$$\phi_p(x_i;\mu_k,\Sigma_k) = \frac{1}{\sqrt{(2\pi)^p\,|\Sigma_k|}} exp\{-\frac{1}{2}(x_i-\mu_k)^T {\Sigma_k}^{-1}(x_i-\mu_k)\}$$

where:

- K is the number of mixture components
- τ_k is the mixing proportion
- μ_k and Σ_k are the mean and covariance matrix of the kth component

Introduction



Matrix Gaussian Mixture Models

$$f(x_i;\Theta) = \sum_{k=1}^{K} \tau_k \phi_{p\times q}(x_i; M_k, \Sigma_k, \Psi_k)$$

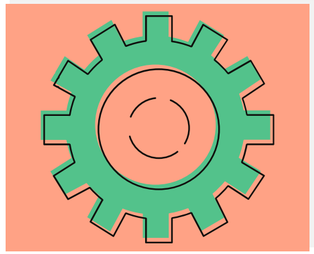
With:

$$\phi_{p\times q}(x_i; M_k, \Sigma_k, \Psi_k) = \left((2\pi)^{-\frac{pq}{2}} |\Psi_k|^{-\frac{p}{2}} |\Sigma_k|^{-\frac{q}{2}} exp\{-\frac{1}{2} tr\left({\Sigma_k}^{-1} (x_i - M_k) {\Psi_k}^{-1} (x_i - M_k)^T \right)\}\right.$$

where:

- M_k is the pxq mean matrix of the k-th component
- Σ_k and Ψ_k are the component rows and columns covariance matrices

Sparse matrix mixture models



Model specification

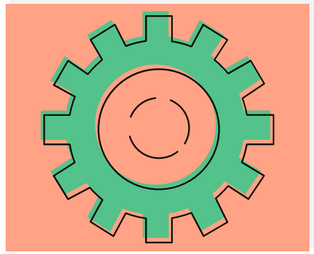
Penalized (Incomplete-data) Log-likelihood

$$\ell_P(\boldsymbol{\Theta}; \mathbf{X}) = \sum_{i=1}^{n} \log \left\{ \sum_{k=1}^{K} \tau_k \phi_{p \times q}(\mathbf{X}_i; \mathbf{M}_k, \boldsymbol{\Omega}_k, \boldsymbol{\Gamma}_k) \right\} - p_{\lambda}(\mathbf{M}_k, \boldsymbol{\Omega}_k, \boldsymbol{\Gamma}_k)$$

Where:

- $\boldsymbol{\Omega}_k = \boldsymbol{\Sigma}_k^{-1} \wedge \boldsymbol{\Gamma}_k = \boldsymbol{\Psi}_k^{-1}$ are the rows and columns precision matrices
- p_λ is the penalty term

Sparse matrix mixture models



Model specification

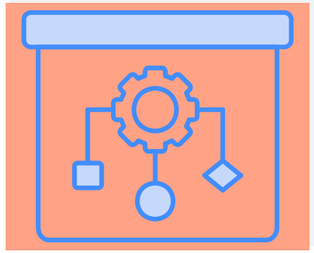
Penalty term

$$p_\lambda(\mathbf{M}_k, \mathbf{\Omega}_k, \mathbf{\Gamma}_k) = \sum_{k=1}^{K} \lambda_1 \sum_{r=1}^{p} \|\mathbf{m}_{r\cdot,k}\|_2 + \sum_{k=1}^{K} \lambda_2 \|\mathbf{P}_2 * \mathbf{\Omega}_k\|_1 + \sum_{k=1}^{K} \lambda_3 \|\mathbf{P}_3 * \mathbf{\Gamma}_k\|_1$$

Where:

- $\lambda_1, \lambda_2, \lambda_3$ are shrinkage hyper-parameters controlling the strength of the penalization
- P_2 and P_3 are symmetric matrices with non-negative entries

Model estimation



A note on related penalty specifications

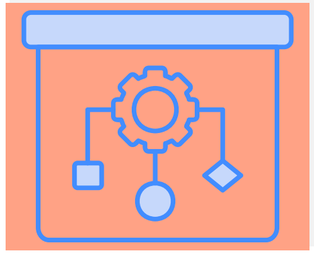
Alternative for the penalty term:

$$p_{\boldsymbol{\lambda}}(\mathbf{M}_k, \boldsymbol{\Omega}_k, \boldsymbol{\Gamma}_k) = \sum_{k=1}^{K} \lambda_1 \|\mathbf{P}_1 * \mathbf{M}_k\|_1 + \sum_{k=1}^{K} \lambda_2 \|\mathbf{P}_2 * \boldsymbol{\Omega}_k\|_1 + \sum_{k=1}^{K} \lambda_3 \|\mathbf{P}_3 * \boldsymbol{\Gamma}_k\|_1$$

Instead of using group lasso for the mean matrices
we can use **graphical lasso** as for the precision matrices

Despite this technique is presented as an alternative in the paper, the lasso method works every time, while group lasso fails sometimes

Model estimation



Model estimation

Penalized complete-data log-likelihood

$$\ell_C(\boldsymbol{\Theta}; \mathbf{X}) = \sum_{i=1}^{n}\sum_{k=1}^{K} z_{ik}\left[\log \tau_k - \frac{pq}{2}\log 2\pi + \frac{q}{2}\log|\boldsymbol{\Omega}_k| + \frac{p}{2}\log|\boldsymbol{\Gamma}_k| + \right.$$
$$\left. -\frac{1}{2}\operatorname{tr}\left\{\boldsymbol{\Omega}_k(\mathbf{X}_i - \mathbf{M}_k)\boldsymbol{\Gamma}_k(\mathbf{X}_i - \mathbf{M}_k)'\right\}\right] - p_{\boldsymbol{\lambda}}(\mathbf{M}_k, \boldsymbol{\Omega}_k, \boldsymbol{\Gamma}_k)$$

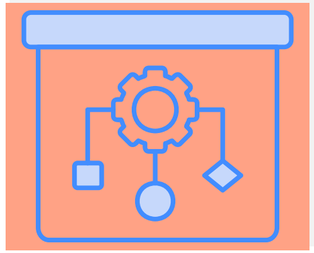
Where z_{ik} is the latent variable, distributed as

$$p(\mathbf{z}_i; \boldsymbol{\tau}) = \prod_{g=1}^{G} \tau_g^{z_{ig}}$$

Initialization:

- Hierarchical clustering (*mclust*)
- Uniform random

Model estimation



EM Algorithm

E-step

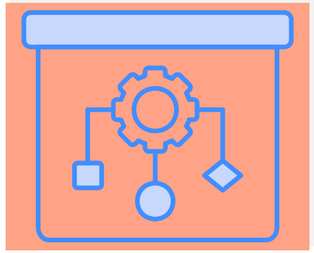
$$\hat{z}_{ik}^{(t)} = \frac{\hat{\tau}_k^{(t-1)} \phi_{p\times q}\left(\mathbf{X}_i; \hat{\mathbf{M}}_k^{(t-1)}, \hat{\mathbf{\Omega}}_k^{(t-1)}, \hat{\mathbf{\Gamma}}_k^{(t-1)}\right)}{\sum_{v=1}^{K} \hat{\tau}_v^{(t-1)} \phi_{p\times q}\left(\mathbf{X}_i; \hat{\mathbf{M}}_v^{(t-1)}, \hat{\mathbf{\Omega}}_v^{(t-1)}, \hat{\mathbf{\Gamma}}_v^{(t-1)}\right)}, \quad i = 1, \ldots, n,$$

M-step

Three Q-functions to maximize:

- $Q_M\left(M_k\right) = f\left(X, M_k\right)$
- $Q_\Omega\left(\Omega_k\right) = f\left(X, \Omega_k\right)$
- $Q_\Gamma\left(\Gamma_k\right) = f\left(X, \Gamma_k\right)$

Model estimation



Sparse estimation of the mean matrices

$$Q_M(\mathrm{M}_k) = \sum_{i=1}^{n} \hat{z}_{ik}^{(t)} \left[\mathrm{tr}\left\{ \hat{\Omega}_k^{(t-1)} \mathrm{X}_i \hat{\Gamma}_k^{(t-1)} \mathrm{M}_k' \right\} - \frac{1}{2} \mathrm{tr}\left\{ \hat{\Omega}_k^{(t-1)} \mathrm{M}_k \hat{\Gamma}_k^{(t-1)} \mathrm{M}_k' \right\} \right] - \lambda_1 \sum_{r=1}^{p} \|\mathbf{m}_{r\cdot,k}\|_2$$

$$= \mathrm{tr}\left\{ \hat{\Omega}_k^{(t-1)} \boldsymbol{S}_M \hat{\Gamma}_k^{(t-1)} \mathrm{M}_k' \right\} - \frac{\hat{n}_k^{(t)}}{2} \mathrm{tr}\left\{ \hat{\Omega}_k^{(t-1)} \mathrm{M}_k \hat{\Gamma}_k^{(t-1)} \mathrm{M}_k' \right\} - \lambda_1 \sum_{r=1}^{p} \|\mathbf{m}_{r\cdot,k}\|_2,$$

solved via **proximal gradient descent** algorithm:

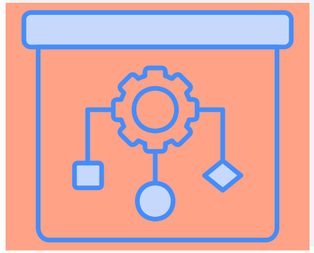
- f is convex and differentiable
- g may not be differentiable in every point

$$\underset{\mathrm{M}_k}{\text{minimize}}\, f(\mathrm{M}_k) + g(\mathrm{M}_k),$$

where

$$f(\mathrm{M}_k) = \frac{\hat{n}_k^{(t)}}{2} \mathrm{tr}\left\{ \hat{\Omega}_k^{(t-1)} \mathrm{M}_k \hat{\Gamma}_k^{(t-1)} \mathrm{M}_k' \right\} - \mathrm{tr}\left\{ \hat{\Omega}_k^{(t-1)} \boldsymbol{S}_M \hat{\Gamma}_k^{(t-1)} \mathrm{M}_k' \right\} \quad \text{and} \quad g(\mathrm{M}_k) = \lambda_1 \sum_{r=1}^{p} \|\mathbf{m}_{r\cdot,k}\|_2.$$

Model estimation



Sparse estimation of the mean matrices

update for the l-th row of the mean matrix:

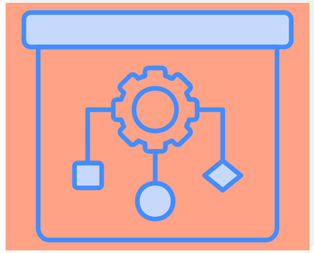
$$\boldsymbol{b} = \mathbf{m}_{l\cdot,k} - \nu \nabla \mathbf{m}_{l\cdot,k},$$

$$\hat{\mathbf{m}}_{l\cdot,k} = \mathrm{prox}_{\nu\lambda_1}(\boldsymbol{b}),$$

where:

- ν is a step size parameter
- $\bigtriangledown m_{l;k} = \left(\frac{df(M_k)}{dM_k} \right)_{l\cdot}$
- $$\mathrm{prox}_{\nu\lambda_1}(\boldsymbol{b}) = \begin{cases} \boldsymbol{b}\left(1 - \frac{\lambda_1 \nu}{\|\boldsymbol{b}\|_2}\right) & \text{if } \|\boldsymbol{b}\|_2 > \lambda_1 \nu, \\ \mathbf{0} & \text{if } \|\boldsymbol{b}\|_2 \leq \lambda_1 \nu. \end{cases}$$

Model estimation



Sparse estimation of the row-precision matrices

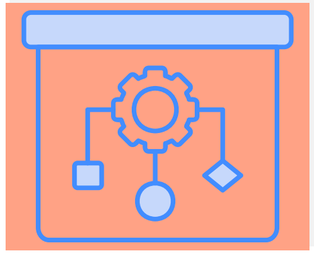
$$Q_\Omega(\boldsymbol{\Omega}_k) = \log|\boldsymbol{\Omega}_k| - \operatorname{tr}\{\boldsymbol{\Omega}_k \boldsymbol{S}_\Omega\} - \frac{2}{\hat{n}_k q}\lambda_2 ||\mathbf{P}_2 * \boldsymbol{\Omega}_k||_1,$$

where:

$$\boldsymbol{S}_\Omega = \sum_{i=1}^{n} \hat{z}_{ik}^{(t)} \frac{\left(\mathbf{X}_i - \hat{\mathbf{M}}_k^{(t)}\right) \hat{\boldsymbol{\Gamma}}_k^{(t-1)} \left(\mathbf{X}_i - \hat{\mathbf{M}}_k^{(t)}\right)'}{\hat{n}_k^{(t)} q}.$$

Maximization of this **graphical Lasso** problem is solved through the coordinate descent algorithm (*glassoFast* package)

Model estimation



Sparse estimation of the column-precision matrices

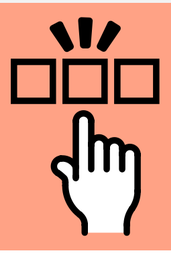
$$Q_\Gamma(\mathbf{\Gamma}_k) = \log|\mathbf{\Gamma}_k| - \mathrm{tr}\left\{\mathbf{\Gamma}_k \boldsymbol{S}_\Gamma\right\} - \frac{2}{\hat{n}_k p}\lambda_3 ||\mathbf{P}_3 * \mathbf{\Gamma}_k||_1,$$

where:

$$\boldsymbol{S}_\Gamma = \sum_{i=1}^{n} \hat{z}_{ik}^{(t)} \frac{\left(\mathbf{X}_i - \hat{\mathbf{M}}_k^{(t)}\right)' \hat{\mathbf{\Omega}}_k^{(t)} \left(\mathbf{X}_i - \hat{\mathbf{M}}_k^{(t)}\right)}{\hat{n}_k^{(t)} p}.$$

Maximization of this **graphical Lasso** problem is solved through the coordinate descent algorithm (*glassoFast* package)

Model selection



Bayesian Information Criterion

$$BIC = 2 \log L \left(\begin{matrix} \Lambda \\ \Theta \end{matrix} \right) - d_0 \log (n)$$

- d_0 is the number of parameters of the model not shrunk to 0
- $log\ L(\hat{\Theta})$ is the log-likelihood evaluated at $\hat{\Theta}$

We aim to maximize this measure to estimate the best λ_1, λ_2, λ_3, P_2, P_3 and K

Application

Dataset: Air Quality in Madrid (2001-2018)

Several sensing stations sparsed in Madrid collected hourly air-quality data over several years

Challenges

- Inconsistent equipment across stations
- Inconsistent deployment time across stations

→ *High rate of missing data*

- Natively non-aggregated data

Application

Dataset: Air Quality in Madrid (2001-2018)

Several sensing stations sparsed in Madrid collected hourly air-quality data over several years

Challenges & Solutions

- Inconsistent equipment across stations → Intersection of columns + discarding statations with high rate of missing data
- Inconsistent deployment time across stations → Kept the timeframe with the lowest missing data rate
- Natively non-aggregated data → Transformed the data into aggregated three-way data

Application

From Time-series to Three-way aggregated data

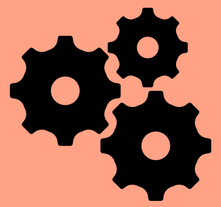
Initial datasets:

- 18 datasets (one per year)
- 33 stations
- 14-17 chemicals monitored (inconsistently)
- ~217k rows per dataset

Transformed dataset:

- Aggregated data
- Three-way: Chemicals (14) x Years (9) x Stations (27)
- Missing data imputed through median over stations dimension
- Normalized through Z-scoring

Application



Model selection and algorithm arguments

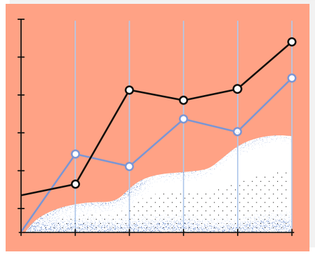
- **K** = 2, ..., 15
- Uniform penalty matrices **P2**, **P3** with values in {0, 0.01, 0.05, 0.1} while **P1** only with 0.05
- The **diagonals** are penalized
- **Graphical Lasso** penalty for Mean matrices M

---

240 models tested → **56% failed** due to errors within the provided library →

- poor handling of edge cases
- numerical instabilities
- half working implementations

Application



- Introduction
- Sparse matrix mixture models
- Model estimation
- Model selection
- Application
- Conclusions

Results

Best model:

K = 13

P1 = 0.05

P2 = 0.01

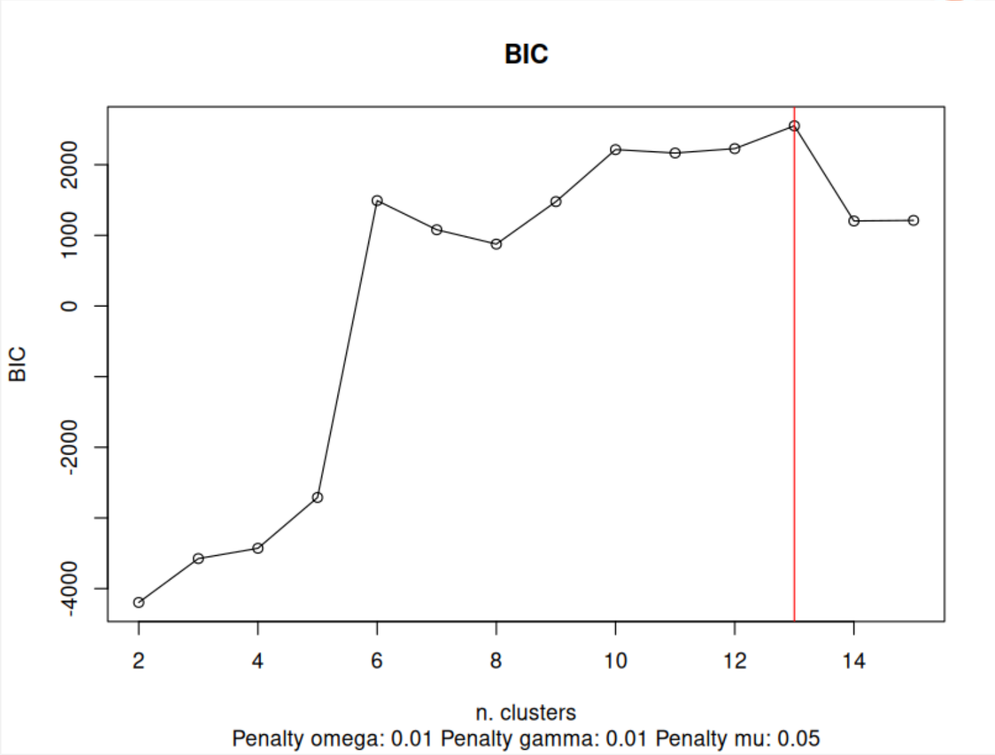
P3 = 0.01

Lasso type penalty for M_k

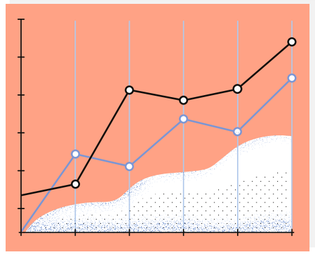
BIC = ~2500

parameters:

- **Ω**: 358 (74% sparse)
- **Γ**: 385 (34% sparse)
- **M**: 1628 (0.006% sparse)



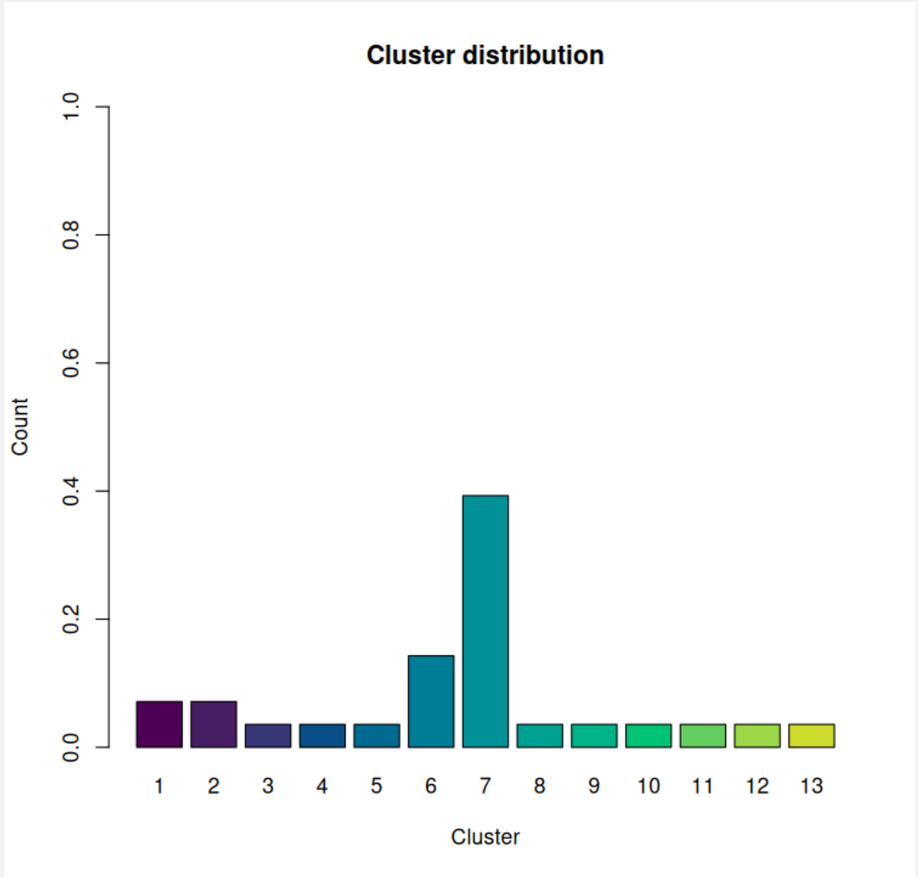
Application



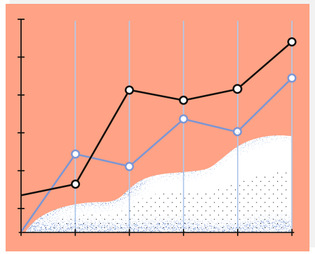
Results: Cluster distribution

From K >= 6 we tend to find

- Dominant cluster
- Secondary cluster
- Several small clusters

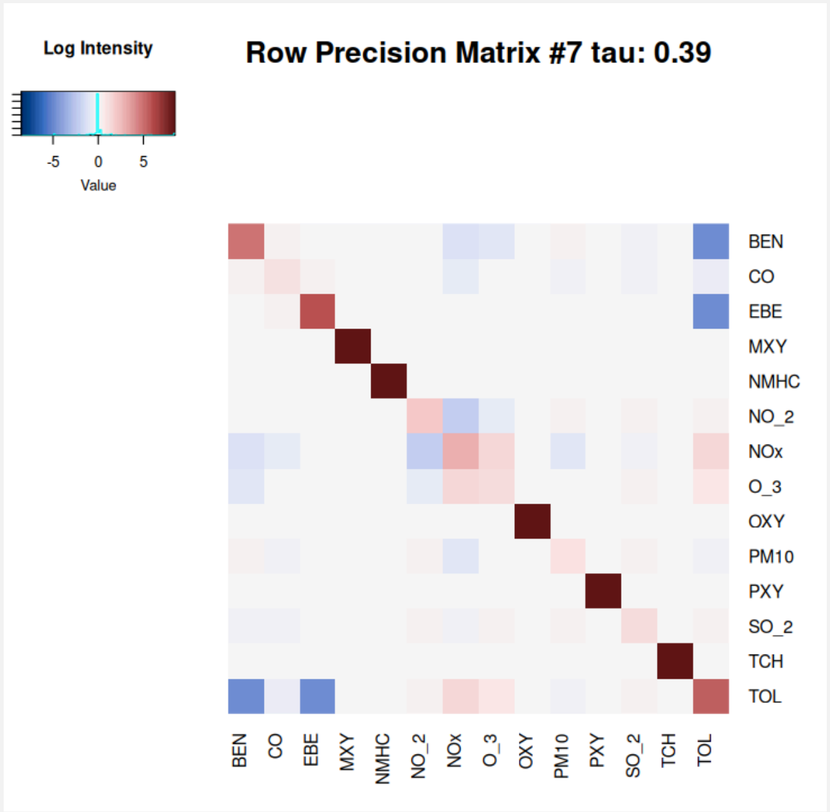


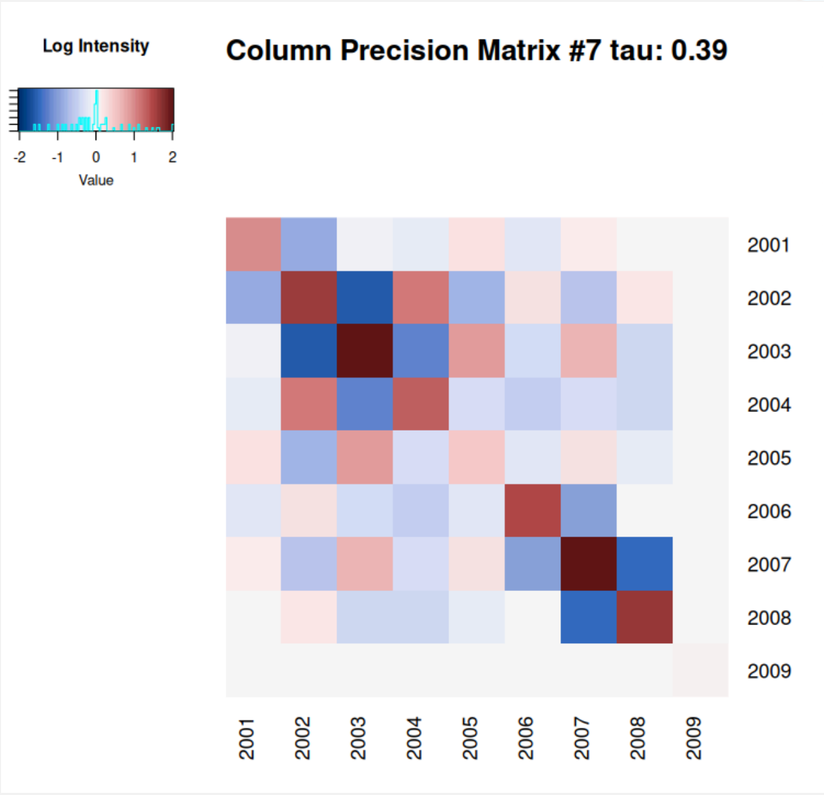
Application



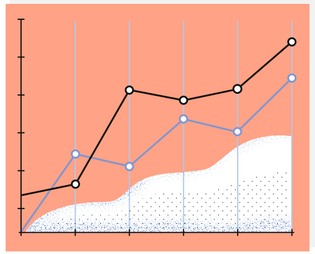
Results: Sparsity

Dominant cluster



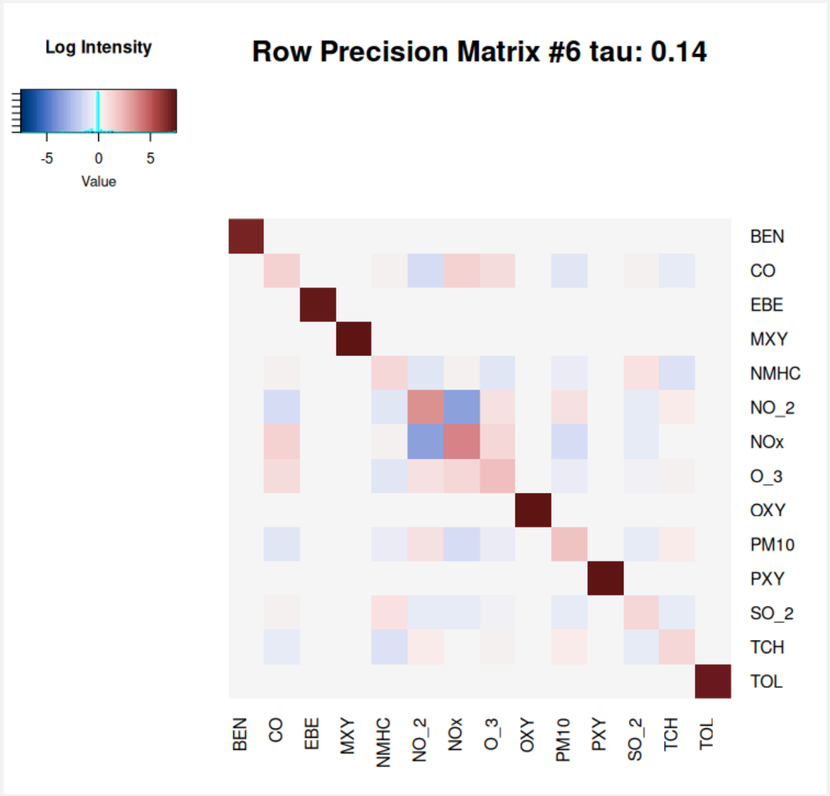


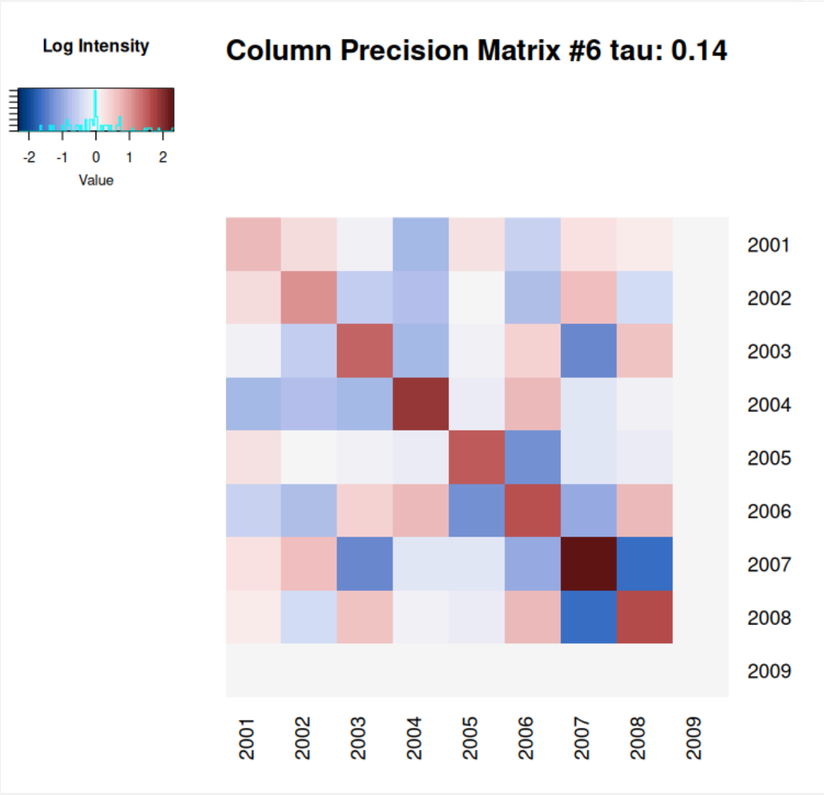
Application



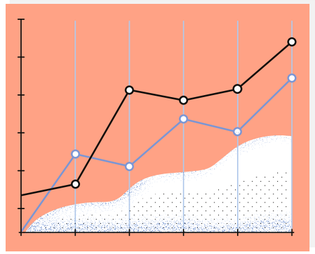
Results: Sparsity

Secondary cluster





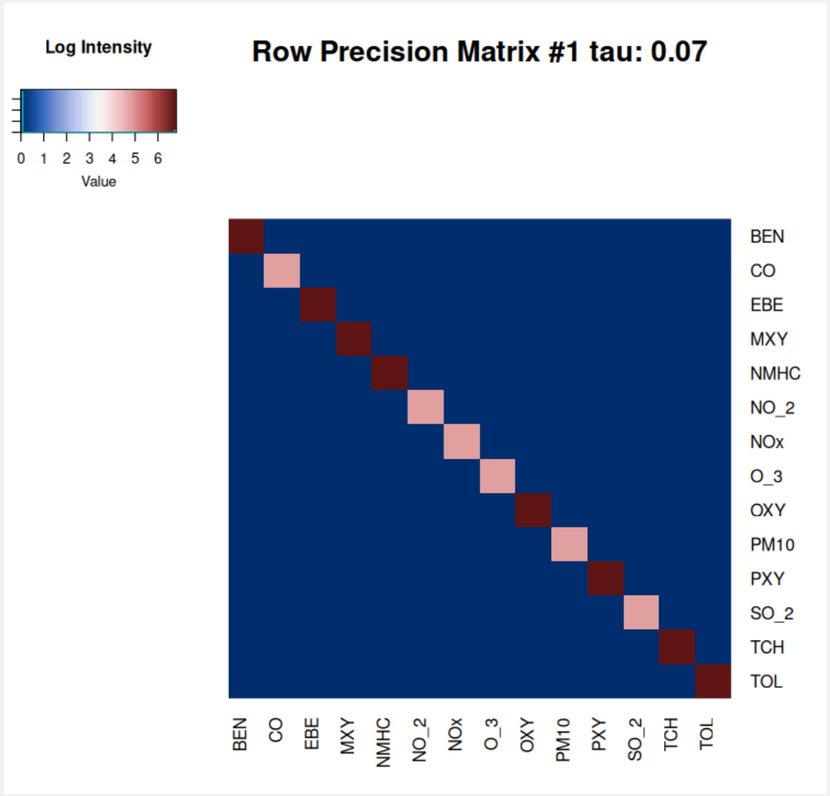
Application

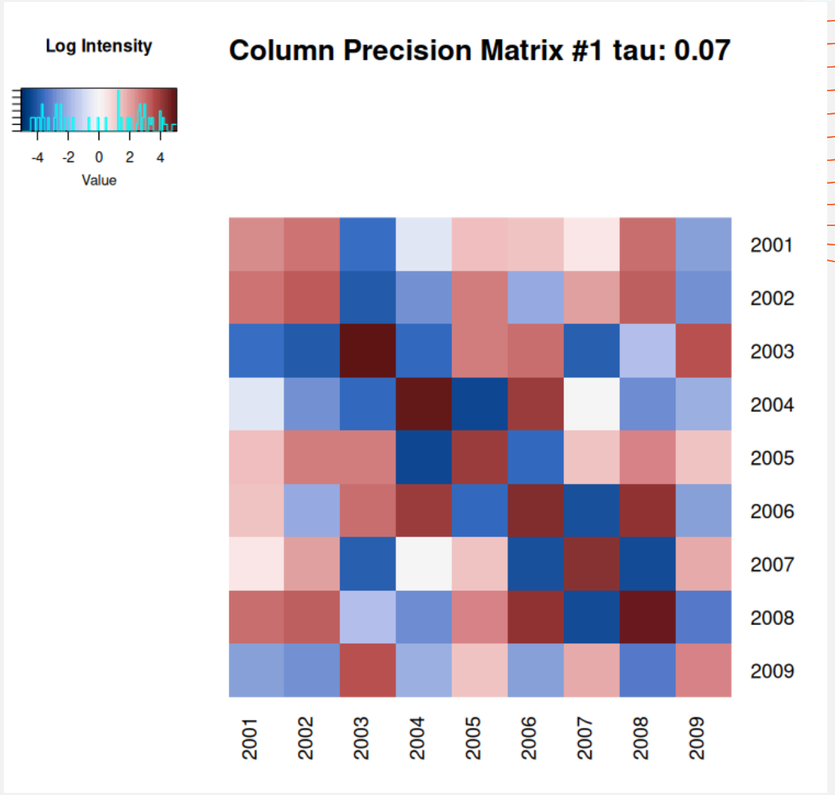


- Introduction
- Sparse matrix mixture models
- Model estimation
- Model selection
- Application
- Conclusions

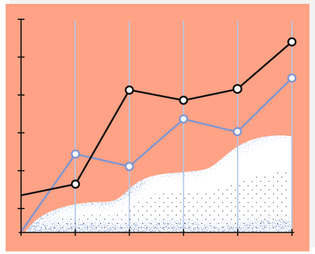
Results: Sparsity

Small cluster





Application

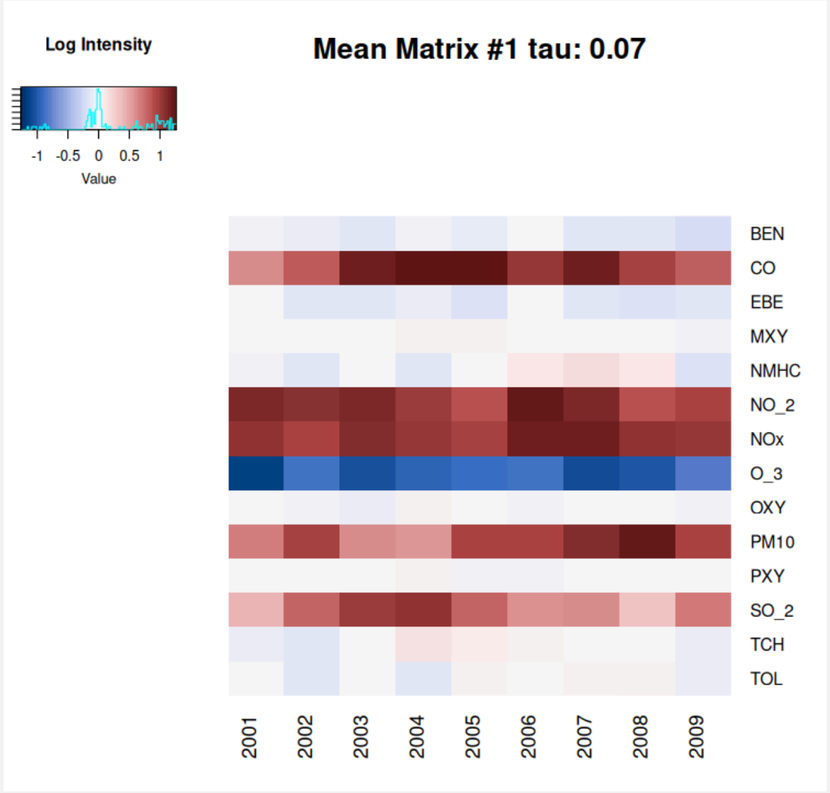


- Introduction
- Sparse matrix mixture models
- Model estimation
- Model selection
- Application
- Conclusions

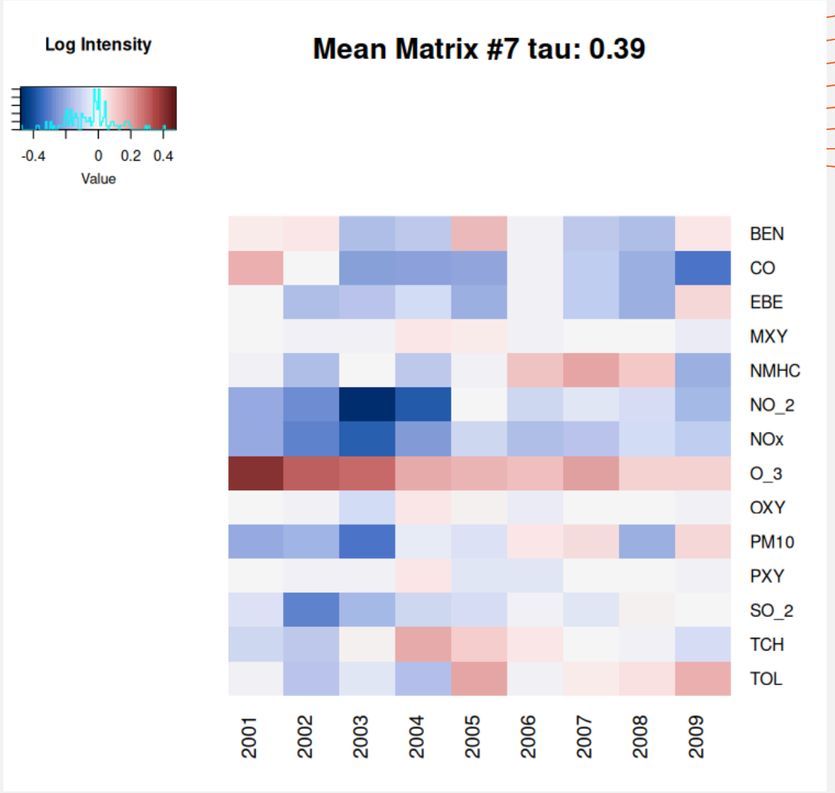
Results: Sparsity

Mean matrices comparison

Small cluster



Dominant cluster



Conclusions

Summary

- We explored the use of three-way data
- We ventured into multidimensional Gaussian distributions and Mixture models
- We studied the problem of overparametrization in multidimensional Mixture models and a solution through a flexible penalization approach
- We applied said techniques in real-world case study and discussed its results and plausible (implementation) flaws

THANK YOU