



Clustering : une vision unifiée pour une utilisation éclairée - Partie 3 : Formalisation par modèles de mélange

C Biernacki

► To cite this version:

C Biernacki. Clustering : une vision unifiée pour une utilisation éclairée - Partie 3 : Formalisation par modèles de mélange. Doctoral. France. 2023. hal-04370761

HAL Id: hal-04370761

<https://inria.hal.science/hal-04370761>

Submitted on 3 Jan 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Clustering : une vision unifiée pour une utilisation éclairée

—

Partie 3 Formalisation par modèles de mélange

C. Biernacki

Ateliers statistiques de la SFdS, 15 et 16 juin 2023, IHP, Paris



Need to formalize
●○○○○○

Formalizing estimation
○○○○○○○○○○○○○○○○○○○○
○○○○○○○○○○○○○○○○○○○○
○○○○○○○○○○○○○○○○○○○○

Formalizing selection
○○○○○○○○○○○○○○○○○○○○
○○○○○○○○○○○○○○○○○○○○

More advanced formalizing
○○○○○○○○○○○○

Experiments
○○○○○○○○○○○○
○○○○○○○○○○○○

To go further
○○○

Outline

1 Need to formalize

2 Formalizing estimation

3 Formalizing selection

4 More advanced formalizing

5 Experiments

6 To go further

Clustering: an ill-posed problem

- We have seen how to perform and evaluate clustering. . .
- . . . but we do not know what is a cluster
- Thus **we have built something without defining it!**

It is a serious problem

Impossible to **provide guarantees** on by-products of clustering (ex.: some user decisions) since no guarantees on clustering itself is really available

$$\mathbf{x} \longrightarrow \hat{\mathbf{z}} = f(\mathbf{x}) \longrightarrow \underbrace{\widehat{\text{decision}} = g(\hat{\mathbf{z}})}_{\text{need guarantees}}$$

Expected guarantees on clustering

- $\hat{\mathbf{z}}, \hat{\boldsymbol{\mu}}, \hat{K}$, etc. are estimates of theoretical quantities $\mathbf{z}, \boldsymbol{\mu}, K$, etc.
- It can be thus expected classical guarantees provided for any estimate in statistics
- Typically: consistency, bias, variance
- Examples:

$$p(\hat{K} = K) = 1 \quad \text{as } n \rightarrow \infty$$

$$p(\hat{\mathbf{z}} = \mathbf{z}) \text{ for finite } n$$

- In the previous lessons we were very far from such a requirement. . .

Key idea

Formalize the rigorous definition of a cluster

The model-based clustering paradigm

a cluster \longleftrightarrow a distribution

- It recasts all previous/next questions into **model design/estimation/selection**
- It takes benefits from all **theoretical statistics** environment

- How to choose the **best metric** $M_{(k)}$?
- How to choose the **number K of clusters**?
- Clusters of **different sizes** are they well estimated?
- How to choose the **data unit**?
- How to **select features**?
- How to deal with **mixed data**?
- How to deal with **missing data**?
- How to deal with **outliers**?
- ...

What about empirical clustering?

Somewhere it works pretty well even if it has the previous mentioned limits

- 1 Interesting to understand why
- 2 Interesting to overcome their limits then

In fact many empirical methods are hidden model-based clustering ones!

Reformulate K -means: the hidden estimate choice

$$W_I(\mathbf{z}) = -2\ell_c(\boldsymbol{\mu}; \mathbf{x}, \mathbf{z}) + \text{cst}$$

⇓

$$\left. \begin{array}{l} \hat{\mathbf{z}}^W = \arg \min_{\mathbf{z}} W_I(\mathbf{z}) \\ (\hat{\mathbf{z}}^{\ell_c}, \hat{\boldsymbol{\mu}}^{\ell_c}) = \arg \max_{(\mathbf{z}, \boldsymbol{\mu})} \ell_c(\boldsymbol{\mu}; \mathbf{x}, \mathbf{z}) \end{array} \right\} \Rightarrow \hat{\mathbf{z}}^W \equiv \hat{\mathbf{z}}^{\ell_c}$$

Estimate

Maximum of the so-called complete-likelihood (see later for its statistical properties)

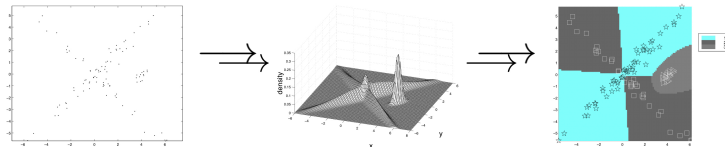
The clustering process in mixtures

- 1 Estimation of θ by $\hat{\theta}$
- 2 Estimation of the **conditional probability** that $\mathbf{x}_i \in G_k$

$$t_{ik}(\hat{\theta}) = p(Z_{ik} = 1 | \mathbf{X}_i = \mathbf{x}_i; \hat{\theta}) = \frac{\hat{\pi}_k p(\mathbf{x}_i; \hat{\alpha}_k)}{p(\mathbf{x}_i; \hat{\theta})}$$

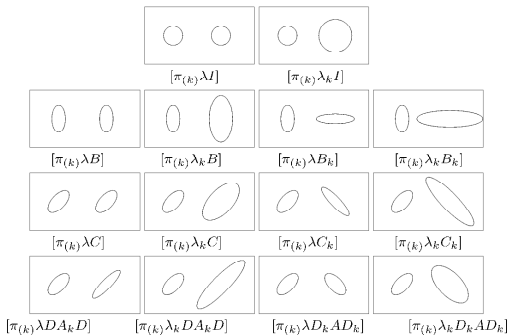
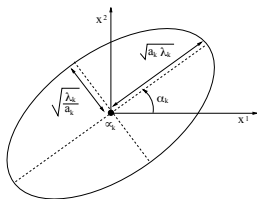
- 3 Estimation of \mathbf{z}_i by *maximum a posteriori* (**MAP**)

$$\hat{\mathbf{z}}_{ik} = \mathbb{I}_{\{k = \arg \max_{h=1, \dots, K} t_{ih}(\hat{\theta})\}}$$



Geometric interpretation of Σ_k

$$\Sigma_k = \underbrace{\lambda_k}_{\text{volume}} \cdot \underbrace{\mathbf{D}_k}_{\text{orientation}} \cdot \underbrace{\mathbf{A}_k}_{\text{shape}} \cdot \mathbf{D}_k'$$



Estimation of θ by *complete*-likelihood

Maximize the *complete-likelihood* over (θ, \mathbf{z})

$$\ell_c(\theta; \mathbf{x}, \mathbf{z}) = \sum_{i=1}^n \sum_{k=1}^K z_{ik} \ln \{ \pi_k p(\mathbf{x}_i; \alpha_k) \}$$

- **Equivalent** to traditional methods

Metric	$\mathbf{M} = \mathbf{I}$	\mathbf{M} free	\mathbf{M}_k free
Gaussian model	$[\pi \lambda I]$	$[\pi \lambda C]$	$[\pi \lambda_k C_k]$

- **Bias** of $\hat{\theta}$: heavy if poor separated clusters
- Associated optimization algorithm: **CEM** (see later)
- CEM with $[\pi \lambda I]$ is **strictly** equivalent to K -means
- CEM is simple et fast (convergence with few iterations)

Estimation of θ by *observe*-likelihood

Maximize the *observe*-likelihood on θ

$$\ell(\theta; \mathbf{x}) = \sum_{i=1}^n \ln p(\mathbf{x}_i; \theta)$$

- **Convergence** of $\hat{\theta}$, asymptotic **efficiency**, asymptotically **unbiased**
- **General** algorithm for missing data: **EM**
- EM is simple but slower than CEM
- Interpretation: it is a kind of **fuzzy clustering**

Principle of EM and CEM

- Initialization: θ^0
- Iteration $n^o q$:
 - Step E: estimate probabilities $\mathbf{t}^q = \{t_{ik}(\theta^q)\}$
 - Step C: classify by setting $\mathbf{t}^q = \text{MAP}(\{t_{ik}(\theta^q)\})$
 - Step M: maximize $\theta^{q+1} = \arg \max_{\theta} \ell_c(\theta; \mathbf{x}, \mathbf{t}^q)$
- Stopping rule: iteration number or criterion stability

Properties

- \oplus : simplicity, monotony, low memory requirement
- \ominus : local maxima (depends on θ^0), linear convergence (EM)

Gaussian M-step

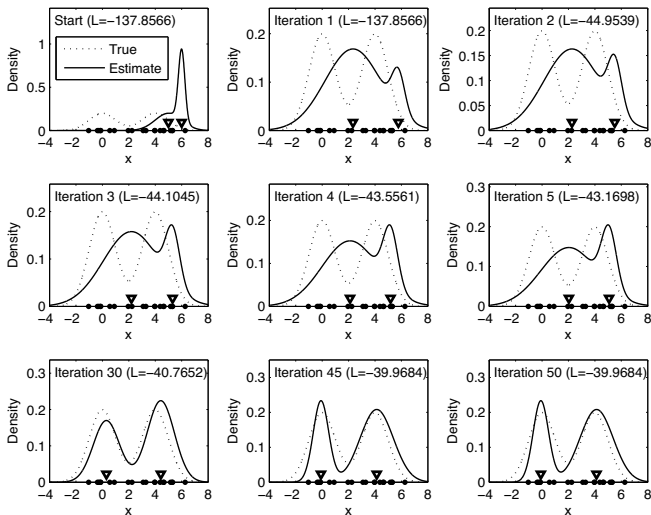
$$n_k^{(q)} = \sum_{i=n^l+1}^n t_{ik}(\boldsymbol{\theta}^{(q)})$$

$$\pi_k^{(q+1)} = \frac{n_k^{(q)}}{n}$$

$$\boldsymbol{\mu}_k^{(q+1)} = \frac{1}{n_k^{(q)}} \left(\sum_{i=1}^n t_{ik}(\boldsymbol{\theta}^{(q)}) \mathbf{x}_i \right)$$

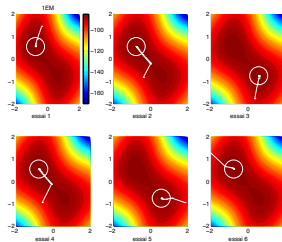
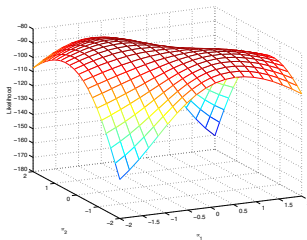
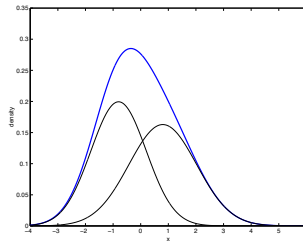
$$\boldsymbol{\Sigma}_k^{(q+1)} = \frac{1}{n_k^{(q)}} \left(\sum_{i=1}^n t_{ik}(\boldsymbol{\theta}^{(q)}) (\mathbf{x}_i - \boldsymbol{\mu}_k^{(q+1)}) (\mathbf{x}_i - \boldsymbol{\mu}_k^{(q+1)})' \right)$$

Example of an EM in the univariate case



Note : low at the beginning but increase of the log-likelihood

Local maxima



Comparison EM-CEM in practice

n	overlap ¹	$\hat{KL}(\theta, \hat{\theta})$		$\text{err}(z, \hat{z})$	
		EM	CEM	EM	CEM
20	low	0.2770	0.2771	0.3383	0.3217
	middle	0.4916	0.3699	0.2050	0.1700
	high	0.4108	0.3132	0.0983	0.0667
200	low	0.0209	0.0822	0.3342	0.3188
	middle	0.0187	0.0425	0.1638	0.1587
	high	0.0172	0.0209	0.0530	0.0500
2000	low	0.0014	0.0454	0.3112	0.3113
	middle	0.0017	0.0246	0.1620	0.1619
	high	0.0017	0.0059	0.0509	0.0510

¹high: 30%, middle: 15%, low:5%

Categorical variables: latent class model

- **Categorical variables:** d variables with m_j modalities each, $\mathbf{x}_i^j \in \{0, 1\}^{m_j}$ and

$$\mathbf{x}_i^{jh} = 1 \quad \Leftrightarrow \quad \text{variable } j \text{ of } \mathbf{x}_i \text{ takes level } h$$

- **Conditional independence:**

$$p(\mathbf{x}_i; \boldsymbol{\alpha}_k) = \prod_{j=1}^d \prod_{h=1}^{m_j} (\alpha_k^{jh})^{x_i^{jh}}$$

and

$$\alpha_k^{jh} = p(\mathbf{x}_i^{jh} = 1 | z_{ik} = 1)$$

with $\boldsymbol{\alpha}_k = (\alpha_k^{jh}; j = 1, \dots, d; h = 1, \dots, m_j)$

Integer: Poisson mixture model

- integer variables: d variables $\mathbf{x}_i^j \in \mathbb{N}$
- Intra conditional independence:

$$p(\mathbf{x}_i^{int}; \boldsymbol{\alpha}_k^{int}) = \prod_{j=1}^d \frac{(\alpha_k^j)^{x_i^j}}{\alpha_k^j!} e^{-\alpha_k^j}$$

SPAM E-mail Database³

- $n = 4601$ e-mails composed by 1813 “spams” and 2788 “good e-mails”
- $d = 48 + 6 = 54$ continuous descriptors²
 - 48 percentages that a given **word** appears in an e-mail (“make”, “you’...)
 - 6 percentages that a given **char** appears in an e-mail (“;”, “\$” ...)
- Transformation of continuous descriptors into **binary descriptors**

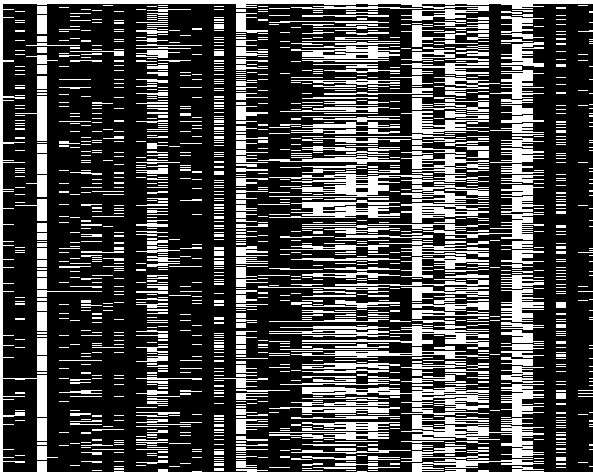
$$x_i^j = \begin{cases} 1 & \text{if word/char } j \text{ appears in e-mail } i \\ 0 & \text{otherwise} \end{cases}$$

²There are 3 other continuous descriptors we do not use

³<https://archive.ics.uci.edu/ml/machine-learning-databases/spambase/>

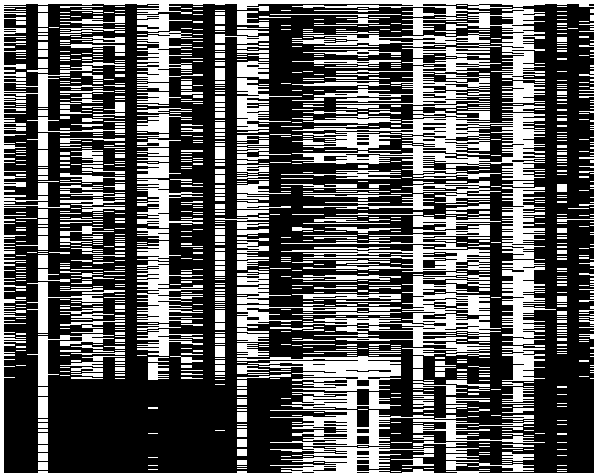
An EM run with a binary data set

Initial binary data



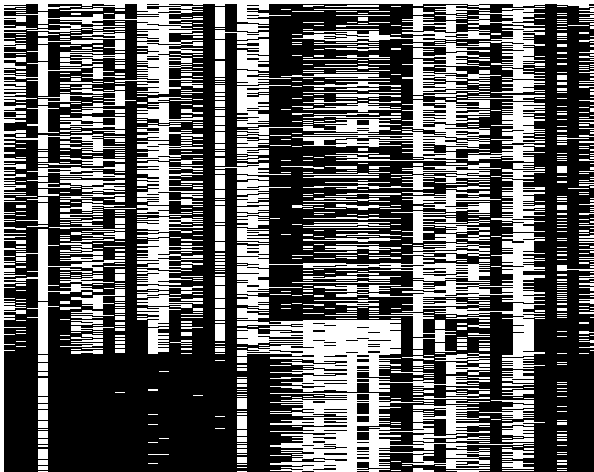
An EM run with a binary data set

Iteration 1



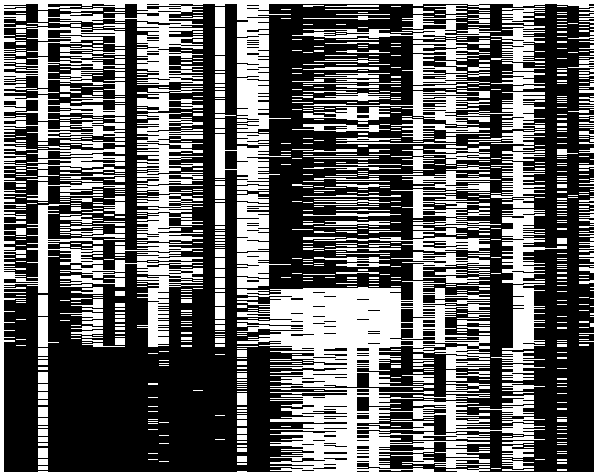
An EM run with a binary data set

Iteration 2



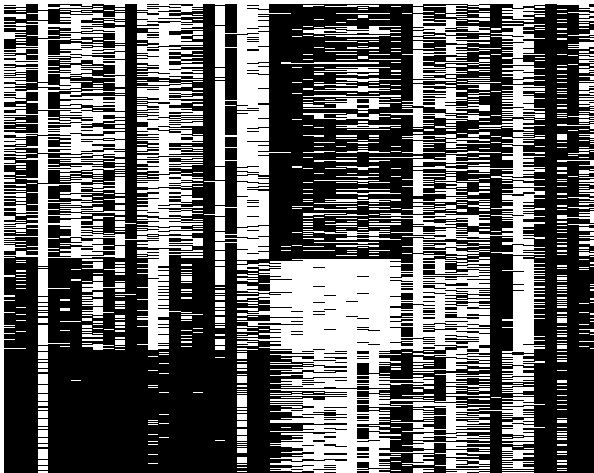
An EM run with a binary data set

Iteration 3



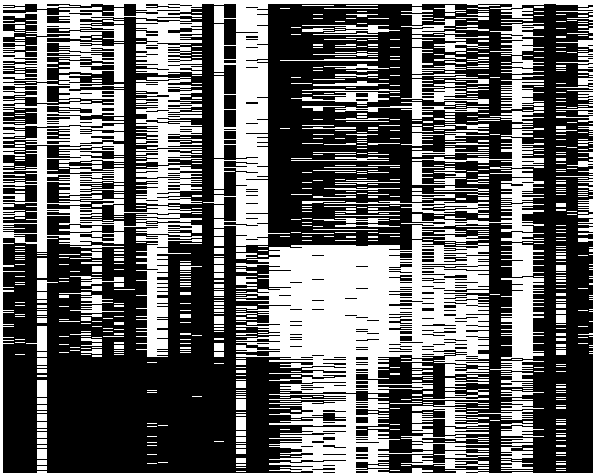
An EM run with a binary data set

Iteration 4



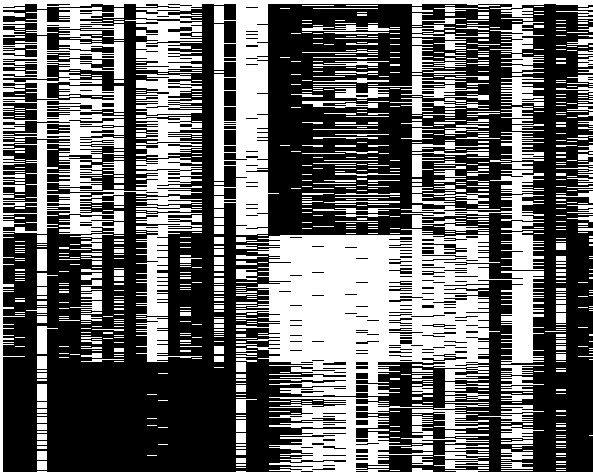
An EM run with a binary data set

Iteration 5



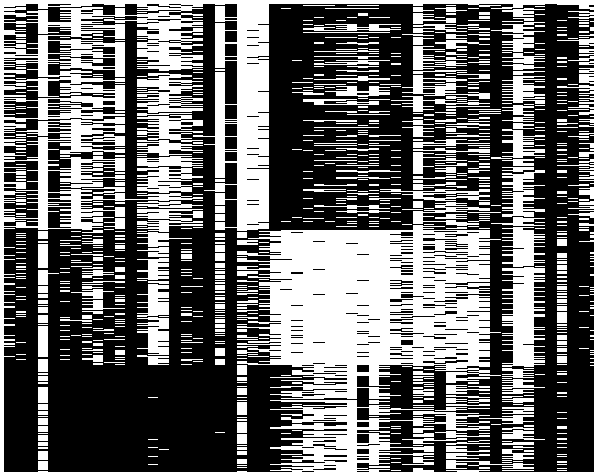
An EM run with a binary data set

Iteration 6



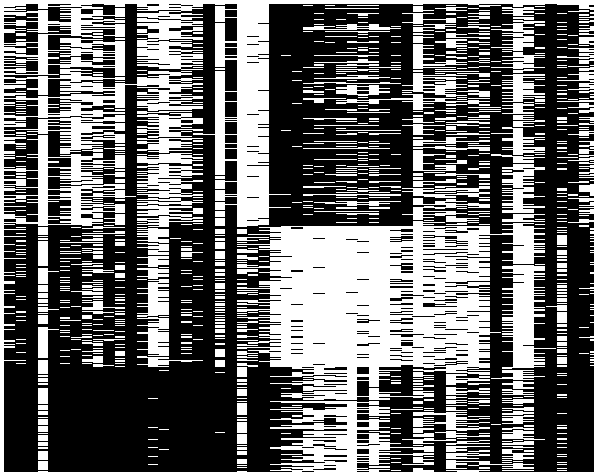
An EM run with a binary data set

Iteration 7



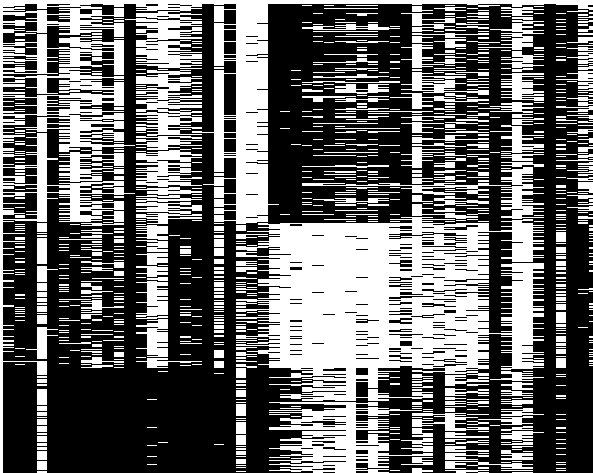
An EM run with a binary data set

Iteration 8



An EM run with a binary data set

Iteration 9



Need to formalize
○○○○○○○

Formalizing estimation
○○○○○○○○○○○○○○○○○○○○●○○
○○○○○○○○○○○○○○○○○○○○
○○○○○○○○○○○○○○○○○○○○

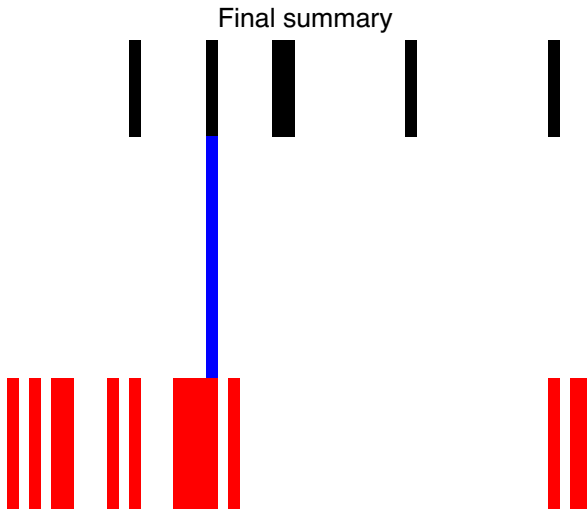
Formalizing selection
○○○○○○○○○○○○○○○○○○○○
○○○○○○○○○○○○○○○○○○○○

More advanced formalizing
○○○○○○○○○○○○○

Experiments
○○○○○○○○○○○
○○○○○○○○○○○

To go further
○○○

An EM run with a binary data set



Mixed data: classical approaches

Usually, unify data type by transformation :

- Quantify continuous variables: [loose some information](#)
- MCA dof categorical variable: [loose the meaning](#)
- ...

Proposal

Model-based directly on [raw data](#)

Mixed data: conditional independence everywhere

The aim is to combine continuous, categorical and integer data

$$\mathbf{x}_1 = (\mathbf{x}_1^{cont}, \mathbf{x}_1^{cat}, \mathbf{x}_1^{int})$$

The proposed solution is to mixed all types by **inter-type conditional independence**

$$p(\mathbf{x}_1; \boldsymbol{\alpha}_k) = p(\mathbf{x}_1^{cont}; \boldsymbol{\alpha}_k^{cont}) \times p(\mathbf{x}_1^{cat}; \boldsymbol{\alpha}_k^{cat}) \times p(\mathbf{x}_1^{int}; \boldsymbol{\alpha}_k^{int})$$

In addition, for symmetry between types, **intra-type conditional independence**

Only need to define the univariate pdf for each variable type!

- **Continuous**: Gaussian
- **Categorical**: multinomial
- **Integer**: Poisson

Missing data: a seminal paper

Biometrika (1976), **63**, 3, pp. 581–92

581

Printed in Great Britain

Inference and missing data

BY DONALD B. RUBIN

Educational Testing Service, Princeton, New Jersey

SUMMARY

When making sampling distribution inferences about the parameter of the data, θ , it is appropriate to ignore the process that causes missing data if the missing data are ‘missing at random’ and the observed data are ‘observed at random’, but these inferences are generally conditional on the observed pattern of missing data. When making direct-likelihood or Bayesian inferences about θ , it is appropriate to ignore the process that causes missing data if the missing data are missing at random and the parameter of the missing data process is ‘distinct’ from θ . These conditions are the weakest general conditions under which ignoring the process that causes missing data always leads to correct inferences.

Some key words: Bayesian inference; Incomplete data; Likelihood inference; Missing at random; Missing data; Missing values; Observed at random; Sampling distribution inference.

Missing data: current solutions

X_1	X_2	X_3	Cluster
1.23	?	3.42	?
?	?	4.10	?
4.53	1.50	5.35	?
?	5.67	?	?

Discarded solutions

- Suppress units and/or variables with missing data \Rightarrow **loss of information**
- Imputation of the missing data by the mean or more evolved methods \Rightarrow **uncertainty of the prediction not taken into account**

Retained solution

Use an **integrated approach** which allows to take into account all the available information to perform clustering

Missing data: notations and MNAR assumption

- $O_i \subseteq \{1, \dots, d\}$ the set of the observed variables from sample i
- \mathbf{x}_i^O the observed data from sample i
- M_i the set of the missing variables for sample i
- μ_{ik}^O the sub-vector of μ_k associated to index O_i (the same for M_i)
- Σ_{ik}^{OM} the sub-matrix of Σ_k associated to row O_i and columns M_i (the same for any other combination)

Assumption on the missingness mechanism

Missing At Random (MAR): the probability that a variable is missing does not depend on its own value given the observed variables.

Missing data: maximum likelihood estimator

Observed log-likelihood...

$$\ell(\theta; \mathbf{x}^O) = \sum_{i=1}^n \log \left(\sum_{k=1}^K \pi_k p(\mathbf{x}_i^O; \alpha_k) \right) = \ln \left[\sum_{k=1}^K \pi_k \underbrace{\int_{\mathbf{x}_i^M} p(\mathbf{x}_i^O, \mathbf{x}_i^M; \alpha_k) d\mathbf{x}_i^M}_{\text{MAR assumption}} \right]$$

EM and Gaussian case: E step

θ and θ^+ the parameters for two successive steps (*idem* for missing data)

$$z_{ik}^+ = P(Z_{ik} = 1 | \mathbf{x}_i^O; \theta) = \frac{\pi_k \phi(\mathbf{x}_i^O; \Sigma_k)}{\sum_{\ell=1}^K \pi_\ell \phi(\mathbf{x}_i^O; \Sigma_\ell)}$$
$$\mathbf{x}_{ik}^{M+} = E \left[\mathbf{X}_i^M \mid \mathbf{x}_i^O, Z_{ik} = 1; \theta \right] = \boldsymbol{\mu}_{ik}^M + \Sigma_{ik}^{MO} \left(\Sigma_{ik}^{OO} \right)^{-1} (\mathbf{x}_i^O - \boldsymbol{\mu}_{ik}^O).$$

Interpretation

- z_{ik}^+ : class posterior probability membership given the available information \mathbf{x}_i^O .
- \mathbf{x}_{ik}^{M+} : conditional imputation of the missing data given the cluster.

EM and Gaussian case: M step

$$\pi_k^+ = \frac{1}{n_k^+} \sum_{i=1}^n z_{ik}^+, \quad \mu_k^+ = \frac{1}{n_k^+} \sum_{i=1}^n z_{ik}^+ \mathbf{x}_{ik}^+$$

$$\Sigma_k^+ = \frac{1}{n_k^+} \sum_{i=1}^n z_{ik}^+ \left[(\mathbf{x}_{ik}^+ - \mu_k^+) (\mathbf{x}_{ik}^+ - \mu_k^+)' + \Sigma_{ik}^+ \right]$$

where $n_k^+ = \sum_{i=1}^n z_{ik}^+$, $\mathbf{x}_{ik}^+ = \begin{pmatrix} \mathbf{x}_i^O \\ \mathbf{x}_{ik}^{M^+} \end{pmatrix}$, $\Sigma_{ik}^+ = \begin{pmatrix} 0_i^O & 0_i^{OM} \\ 0_i^{MO} & \Sigma_{ik}^{M^+} \end{pmatrix}$ with 0 the $d \times d$ null matrix, and $\Sigma_{ik}^{M^+} = \Sigma_{ik}^{MO} (\Sigma_{ik}^O)^{-1} \Sigma_{ik}^{OM}$.

Interpretation of $\Sigma_{ik}^{M^+}$

Variance correction due to the under-estimation of variability caused by the imputation of missing data.

Missing data: SEM algorithm

A SEM algorithm to estimate θ by maximizing the **observed**-data log-likelihood

- Initialisation: $\theta^{(0)}$
- Iteration nb q :
 - **E-step**: compute conditional probabilities $p(\mathbf{x}^M, \mathbf{z} | \mathcal{D}; \theta^{(q)})$
 - **S-step**: draw $(\mathbf{x}^{M(q)}, \mathbf{z}^{(q)})$ from $p(\mathbf{x}^M, \mathbf{z} | \mathbf{x}^O; \theta^{(q)})$
 - **M-step**: maximize $\theta^{(q+1)} = \arg \max_{\theta} \ln p(\mathbf{x}^O, \mathbf{x}^{M(q)}, \mathbf{z}^{(q)}; \theta)$
- Stopping rule: iteration number

Properties: simpler than EM and interesting properties!

- Avoid possibly difficult E-step in an EM
- Classical M steps
- Avoids local maxima
- The mean of the sequence $(\theta^{(q)})$ approximates $\hat{\theta}$
- The variance of the sequence $(\theta^{(q)})$ gives confidence intervals

Missing data: SE algorithm

A SE algorithm estimates then $(\mathbf{x}^M, \mathbf{z}^M)$

- Iteration nb q :
 - **E-step**: compute conditional probabilities $p(\mathbf{x}^M, \mathbf{z}^M | \mathcal{D}; \hat{\theta})$
 - **S-step**: draw $(\mathbf{x}^{M(q)}, \mathbf{z}^{M(q)})$ from $p(\mathbf{x}^M, \mathbf{z}^M | \mathcal{D}; \hat{\theta})$
- Stopping rule: iteration number

Properties

- simplicity because of conditional independence
- the mean/mode of the sequence $(\mathbf{x}^{M(q)}, \mathbf{z}^{M(q)})$ estimates $(\mathbf{x}^M, \mathbf{z}^M)$
- confidence intervals are also derived

Missing data: illustration with the cancer data set (1/2)

- **Strategy “mice⁴ + mixture”**: mixture on the dataset completed by mice

```
> data.imp=mice(data)
> data.comp.mice=complete(data.imp)
```

- **Strategy “full mixture”**: mixture on the observed (no completed) dataset

⁴<http://cran.r-project.org/web/packages/mice/mice.pdf>

Missing data: illustration with the cancer data set (2/2)

Strategy	mice + mixture	full mixture
% misclassified	12.8	8.1

Avoid to complete missing data ([imputation depends on the purpose](#))

Need to formalize
○○○○○○○

Formalizing estimation
○○○○○○○○○○○○○○○○
○○○○○○○○○○○○○○○○
○○○○○○○○○○

Formalizing selection
●○○○○○○○○○○○○○
○○○○○○○○○○○○○

More advanced formalizing
○○○○○○○○○○○

Experiments
○○○○○○○○○○○
○○○○○○○○○

To go further
○○○

Outline

- 1 Need to formalize
- 2 Formalizing estimation
- 3 Formalizing selection**
- 4 More advanced formalizing
- 5 Experiments
- 6 To go further

Keep in mind

George E.P. Box (1987)

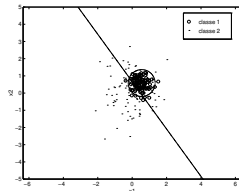
“Essentially, all models are **wrong**, but some are **useful**”

- $\mathcal{M} = \{\mathbf{m}\}$ will denote the set of competing models
- The true distribution p is **not** necessarily in \mathcal{M}
- **Density estimation**: AIC, BIC
- **Clustering**: ICL, CL, NEC

Importance of model selection: example

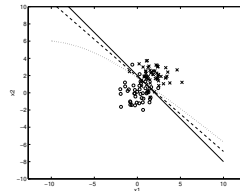
Model = number of clusters + parametric structure of clusters

Too simple model: | bias



true modèle: $[\pi \lambda_k I]$
 too simple model: $[\pi \lambda I]$

Too complex model: variance



— true borderline
 - - - borderline with $[\pi \lambda I]$
 . . . borderline with $[\pi \lambda_k C_k]$

Importance of model selection: bias/variance trade-off

- **Partition error rate:** $\text{err}(\mathbf{z}_1, \mathbf{z}_2) \geq 0$ a distance-like between two partitions $\mathbf{z}_1, \mathbf{z}_2$
- **Gap between true and model partition:**

$$\theta_m^* = \arg \min_{\theta \in \Theta_m} \text{err}(\mathbf{z}, \mathbf{z}(\theta))$$

- **MLE:**

$$\hat{\theta}_m = \arg \max_{\theta \in \Theta} \ell(\theta; \mathcal{D})$$

- **Fundamental decomposition of $\text{err}(\mathbf{z}, \mathbf{z}(\hat{\theta}_m))$:**

$$\begin{aligned} \text{err}(\mathbf{z}, \mathbf{z}(\hat{\theta}_m)) &= \left\{ \text{err}(\mathbf{z}, \mathbf{z}(\theta_m^*)) - \text{err}(\mathbf{z}, \mathbf{z}) \right\} + \left\{ \text{err}(\mathbf{z}, \mathbf{z}(\hat{\theta}_m)) - \text{err}(\mathbf{z}, \mathbf{z}(\theta_m^*)) \right\} \\ &= \left\{ \text{bias}_m \right\} + \left\{ \text{variance}_m \right\} \end{aligned}$$

Importance of model selection: illustration of the variance effect

30 samples from a bivariate mixture with two components

$$\pi_1 = \pi_2 = 0.5, \quad \mu_1 = (0, 0)', \quad \mu_2 = (2, 2)', \quad \Sigma_1 = \Sigma_2 = \mathbf{I}$$

$$\mathcal{M} = \{\text{spherical, general}\}$$

n	\mathbf{m}	$\text{err}(\mathbf{z}, \hat{\mathbf{z}}_{\mathbf{m}})$
40	spherical	0.0967
	general	0.1100
200	spherical	0.0840
	general	0.0872

Some heuristics entropy-based criteria: examples

- A fundamental decomposition of $\ell(\boldsymbol{\theta}; \mathbf{x})$: for any “fuzzy partition” $\mathbf{c} = \{c_{ik}\}$

$$\begin{aligned}\ell(\boldsymbol{\theta}; \mathbf{x}) &= \sum_{i=1}^n \sum_{k=1}^K c_{ik} \ln \{\pi_k p(\mathbf{x}_i; \boldsymbol{\alpha}_k)\} - \sum_{i=1}^n \sum_{k=1}^K c_{ik} \ln t_{ik}(\boldsymbol{\theta}) \\ &= \ell_c(\boldsymbol{\theta}; \mathbf{x}, \mathbf{c}) + \xi(\boldsymbol{\theta}; \mathbf{c}) \\ &= \text{complete-data log-likelihood} + \text{entropy}\end{aligned}$$

- NEC criterion (*Normalized Entropy Criterion*): retain \mathbf{m} minimizing

$$\text{NEC}_K = \begin{cases} \frac{\xi(\hat{\boldsymbol{\theta}}_K; \mathbf{t}(\hat{\boldsymbol{\theta}}_K))}{\ell(\hat{\boldsymbol{\theta}}_K; \mathbf{x}) - \ell(\hat{\boldsymbol{\theta}}_1; \mathbf{x})} & \text{if } K > 1 \\ 1 & \text{if } K = 1 \end{cases}$$

- CL criterion (*Completed Likelihood*): retain \mathbf{m} maximizing

$$\text{CL} = \ell_c(\hat{\boldsymbol{\theta}}; \mathbf{x}, \hat{\mathbf{z}}) = \underbrace{\ell(\hat{\boldsymbol{\theta}}; \mathbf{x})}_{\text{model adequacy}} - \underbrace{\xi(\hat{\boldsymbol{\theta}}; \hat{\mathbf{z}})}_{\text{partition evidence}}$$

- Behaviour: not completely satisfactory but something happens...

Need to formalize
○○○○○○○

Formalizing estimation
○○○○○○○○○○○○○○
○○○○○○○○○○○○○○
○○○○○○○○○○○○

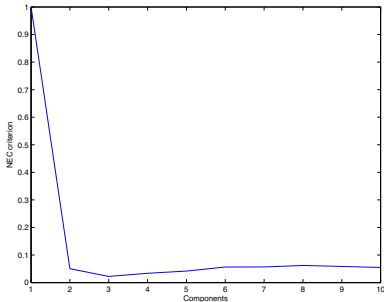
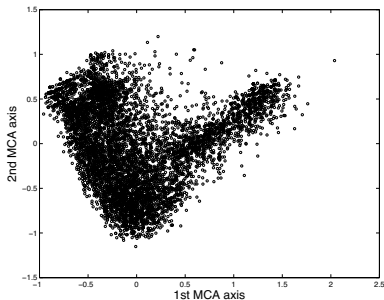
Formalizing selection
○○○○○○●○○○○○
○○○○○○○○○○○

More advanced formalizing
○○○○○○○○○○○

Experiments
○○○○○○○○○○○
○○○○○○○○○○○

To go further
○○○

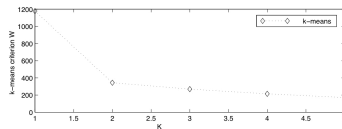
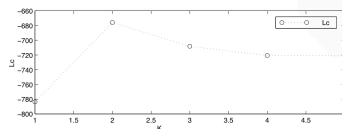
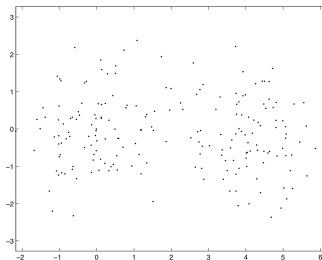
Some heuristics entropy-based criteria: NEC illustration



Some heuristics entropy-based criteria: CL illustration

Interpretation as a penalized within-cluster inertia criterion:

$$CL_{([p\lambda I], K)} = -\frac{nd}{2} \ln(W_K) - n \ln(K) + \text{cst}$$



Theoretical model selection criteria

The most widespread principle

$$\underbrace{\text{Criterion}}_{\text{to be maximized}} = \underbrace{\text{maximum log-likelihood}}_{\text{model-data adequacy}} - \underbrace{\text{penalty}}_{\text{"cost" of the model}}$$

criterion	penalty	interpretation	user purpose
-----------	---------	----------------	--------------

general criteria in statistics

AIC	ν	model complexity	prediction
BIC	$0.5\nu \ln(n)$	model complexity	identification

specific criterion for the clustering aim

ICL	$0.5\nu \ln(n)$ $-\sum_{i,k} \hat{z}_{ik} \ln t_{ik}(\hat{\theta})$	model complexity + partition entropy	well-separated clusters
-----	--	---	-------------------------

BIC criterion: integrated likelihood

■ Posterior likelihood of \mathbf{m} :

$$p(\mathbf{m}|\mathcal{D}) \propto p(\mathcal{D}|\mathbf{m}) \underbrace{p(\mathbf{m})}_{\text{prior on } \mathbf{m}}$$

■ Ideal model in a Bayesian context:

$$\hat{\mathbf{m}}^* \in \arg \max_{\mathbf{m} \in \mathcal{M}} p(\mathbf{m}|\mathcal{D})$$

■ Integrated likelihood: if $p(\mathbf{m}) = \text{cst}$, it is equivalent to maximize

$$p(\mathcal{D}|\mathbf{m}) = \int_{\Theta} p(\mathcal{D}; \theta, \mathbf{m}) \underbrace{p(\theta|\mathbf{m})}_{\text{prior on } \theta} d\theta$$

■ Difficulties:

- Choose the prior $p(\theta|\mathbf{m})$
- Evaluate the integral

BIC criterion: genesis

- **Laplace-Metropolis approximation**: under standard regularity conditions, we have

$$\ln p(\mathcal{D}|\mathbf{m}) = \ell(\hat{\boldsymbol{\theta}}; \mathcal{D}) - \frac{\nu}{2} \ln(n) + O_p(1)$$

- **BIC criterion (*Bayesian Information Criterion*)**: retain \mathbf{m} maximizing

$$\text{BIC}_{\mathbf{m}} = \ell(\hat{\boldsymbol{\theta}}_{\mathbf{m}}; \mathcal{D}) - \frac{\nu_{\mathbf{m}}}{2} \ln(n)$$

BIC criterion: consistency

- **Consistency:** BIC asymptotically selects the best

$$\mathbf{m}^* = \arg \inf_{\mathbf{m} \in \mathcal{M}} \text{KL}(\mathbf{p}, \mathbf{p}_{\theta_{\mathbf{m}}^*})$$

- **Theoretical illustration of consistency:** $\mathbf{m}_1 \subseteq \mathbf{m}_2$, \mathbf{m}_1 being the true model, $\Delta\nu = \nu_2 - \nu_1$, $\Delta\ell = \ell(\hat{\theta}_2; \mathcal{D}) - \ell(\hat{\theta}_1; \mathcal{D})$, we have

$$2(\text{BIC}_2 - \text{BIC}_1) + \Delta\nu \ln(n) = 2\Delta\ell \xrightarrow{d} \chi_{\Delta\nu}^2$$

With $\mu = \Delta\nu$ and $\sigma^2 = 2\Delta\nu$ the mean and the variance of $\chi_{\Delta\nu}^2$

$$\mathbf{p}(\chi_{\Delta\nu}^2 > \Delta\nu \ln(n)) \leq \mathbf{p}(|\chi_{\Delta\nu}^2 - \mu| > \Delta\nu \ln(n) - \mu) \leq \frac{\sigma^2}{(\Delta\nu \ln(n) - \mu)^2} \xrightarrow{n \rightarrow \infty} 0$$

by using the Chebyshev inequality. Thus, asymptotically, BIC will select \mathbf{m}_1

- **Special case of K :** be careful on the χ^2 approximation validity. . .

ICL criterion: genesis

- Revisiting the fundamental decomposition: if \mathbf{z} known, retain \mathbf{m} maximizing

$$\underbrace{\ln p(\mathbf{x}, \mathbf{z} | \mathbf{m})}_{\text{all data evidence}} = \underbrace{\ln p(\mathbf{x} | \mathbf{m})}_{\text{data } \mathbf{x} \text{ evidence}} + \underbrace{\ln p(\mathbf{z} | \mathbf{x}, \mathbf{m})}_{\text{partition } \mathbf{z} \text{ evidence}}$$

Thus models leading to overlapping groups are more penalized (low \mathbf{z} evidence)

- ICL criterion (*Integrated Classification Likelihood*): replace \mathbf{z} by $\hat{\mathbf{z}}$

$$\text{ICL} = \ln p(\mathbf{x}, \hat{\mathbf{z}} | \mathbf{m})$$

- BIC-like approximation of ICL:

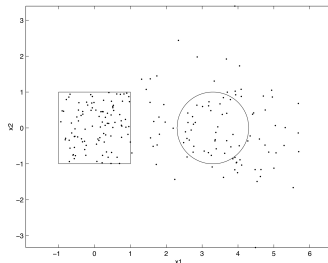
$$\ln p(\mathbf{x}, \mathbf{z} | \mathbf{m}) = \ln p(\mathbf{x}, \mathbf{z}; \hat{\boldsymbol{\theta}}_{\mathbf{x}, \mathbf{z}} | \mathbf{m}) - \frac{\nu}{2} \ln n + O_p(1)$$

In case of the right model \mathbf{m} : $\hat{\boldsymbol{\theta}}_{\mathbf{x}, \mathbf{z}} \xrightarrow{a.s.} \boldsymbol{\theta}^*$ and $\hat{\boldsymbol{\theta}}_{\mathbf{x}} \xrightarrow{a.s.} \boldsymbol{\theta}^*$. Thus, for n large enough, $\hat{\boldsymbol{\theta}}_{\mathbf{x}, \mathbf{z}} \approx \hat{\boldsymbol{\theta}}_{\mathbf{x}}$. Then, we take $\hat{\mathbf{z}} = \text{MAP}(\hat{\boldsymbol{\theta}}_{\mathbf{x}})$ (or also $\hat{\mathbf{z}} = \mathbf{t}(\hat{\boldsymbol{\theta}}_{\mathbf{x}})$). It gives

$$\begin{aligned} \text{ICL}_{\text{bic}} &= \ln p(\mathbf{x}, \hat{\mathbf{z}}; \hat{\boldsymbol{\theta}}_{\mathbf{x}}) - \frac{\nu}{2} \ln n \\ &= \text{BIC} - \xi(\hat{\boldsymbol{\theta}}_{\mathbf{x}}; \hat{\mathbf{z}}) \\ &= \text{CL} - \frac{\nu}{2} \ln n \end{aligned}$$

ICL criterion: robustness to model misspecification

- A bivariate mixture of a uniform and a Gaussian cluster:
 - non-Gaussian component: $\pi_1 = 0.5$, $p_1(\mathbf{x}_1) = 0.25 \mathbf{1}_{[-1,1]}(x^1) \mathbf{1}_{[-1,1]}(x^2)$
 - Gaussian component: $\pi_2 = 0.5$, $\boldsymbol{\mu}_2 = (3.3, 0)'$, $\boldsymbol{\Sigma}_2 = \mathbf{I}$
- 50 simulated data sets of size $n = 200$



K	1	2	3	4	5
BIC	.	60	.	32	8
ICLbic	.	100	.	.	.

ICL criterion: consistency?

- **Assumption:** true model with two groups and parameter θ_2^*
- **Theoretical result:**
 - Preliminaries: $\delta_n = n(\theta_2^* - \theta_2^{*p})' J(\theta_2^*)(\theta_2^* - \theta_2^{*p})$, $J(\theta_2^*)$ the Fisher matrix for a data unit calculated with the true parameter θ_2 and θ_2^{*p} its projected value on the parameter subspace associated to the one component case, $\mu_n = E[\chi_{\Delta\nu}^2(\delta_n)] = \Delta\nu + \delta_n$, $\sigma_n^2 = \text{Var}[\chi_{\Delta\nu}^2(\delta_n)] = 2(\Delta\nu + \delta_n)$
 - Asymptotically: by Chebishev inequality, with $\mu_n - \Delta\nu \ln n - 2n \ln 2 > 0$

$$p(\text{choose wrong model}) = p(\text{ICLbic}_2 < \text{ICLbic}_1) \leq \frac{\sigma_n^2}{(\mu_n - \Delta\nu \ln n - 2n \ln 2)^2}$$

Thus it goes towards 0 for well-separated groups

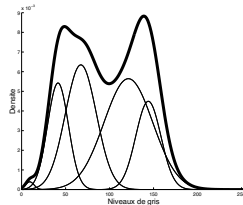
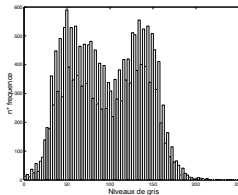
- **Experimental result:** 100 samples from a univariate Gaussian mixture

$$\pi_1 = \pi_2, \quad \mu_1 = 0, \quad \mu_2 = \Delta\mu, \quad \sigma_1^2 = \sigma_2^2 = 1$$

$\Delta\mu$	2.9		3.0		3.1		3.2		3.3	
n	BIC	ICL	BIC	ICL	BIC	ICL	BIC	ICL	BIC	ICL
100	94	23	96	31	97	44	95	45	97	60
400	100	9	100	21	100	48	100	70	100	85
700	100	8	100	15	100	39	100	72	100	96
1 000	100	6	100	16	100	56	100	75	100	91

Large n : BIC behaviour (1/2)

- The mixture density is wrong (as all models)
- Mixtures allow to estimate any distribution by increasing the number of components (high flexibility)



Missing data: illustration with the cancer data set (1/2)

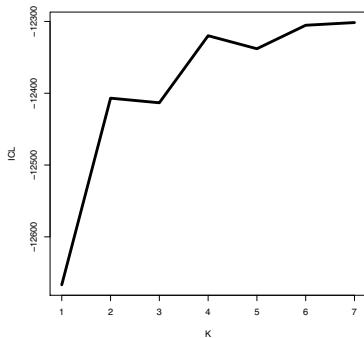
- **Strategy “mice⁵ + mixture”**: mixture on the dataset completed by mice

```
> data.imp=mice(data)
> data.comp.mice=complete(data.imp)
```

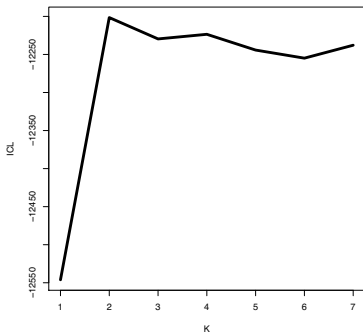
- **Strategy “full mixture”**: mixture on the observed (no completed) dataset

⁵<http://cran.r-project.org/web/packages/mice/mice.pdf>

Missing data: illustration with the cancer data set (2/2)



mice + mixture
 $\hat{K} = 7$



full mixture
 $\hat{K} = 2$

... may lose some cluster information when imputation before clustering

Need to formalize
○○○○○○○

Formalizing estimation
○○○○○○○○○○○○○○
○○○○○○○○○○○○○○○
○○○○○○○○○○○○○

Formalizing selection
○○○○○○○○○○○○○
○○○○○○○○●○○○

More advanced formalizing
○○○○○○○○○○○

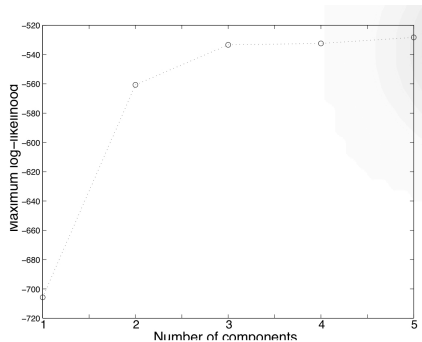
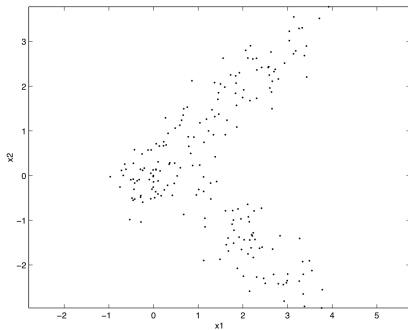
Experiments
○○○○○○○○○○○
○○○○○○○○○

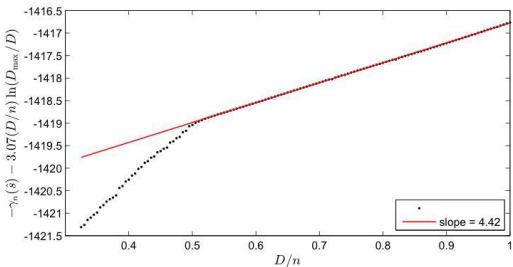
To go further
○○○

CAUTION

Impossible to use BIC/ICL for kernel/spectral clustering (data set has changed...)

Reformulate K -means: elbow as a slope heuristics (2/3)





Need to formalize
○○○○○○○

Formalizing estimation
○○○○○○○○○○○○○○○○
○○○○○○○○○○○○○○○○
○○○○○○○○○○○○○○○○

Formalizing selection
○○○○○○○○○○○○○○○
○○○○○○○○○○○○○○○

More advanced formalizing
●○○○○○○○○○○○

Experiments
○○○○○○○○○○○
○○○○○○○○○○○

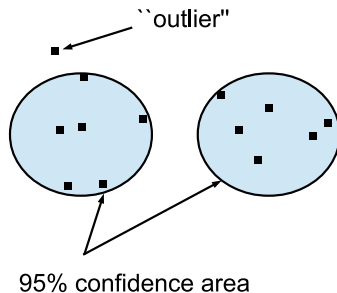
To go further
○○○

Outline

- 1 Need to formalize
- 2 Formalizing estimation
- 3 Formalizing selection
- 4 More advanced formalizing**
- 5 Experiments
- 6 To go further

Outliers: Two possibilities

- “After”: exclude data outside the confidence area of clusters

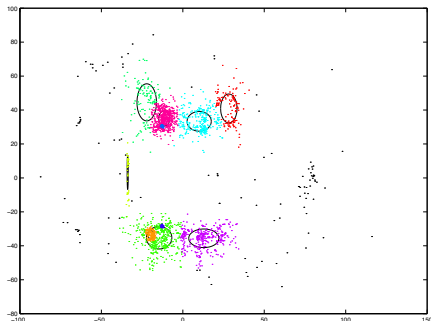
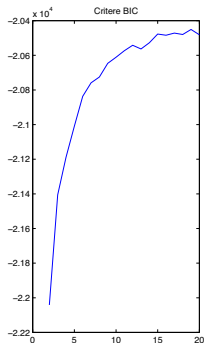
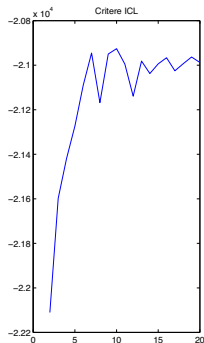


- “During”: model outliers as a particular cluster in the mixture

$$p(\mathbf{x}_1; \theta) = \sum_{k=1}^K \pi_k p(\mathbf{x}_1; \alpha_k) + \pi_0 p(\mathbf{x}_1; \alpha_0)$$

Outliers: “during” example with acoustic emission control

- **Data:** $n = 2\,061$ event locations in a rectangle of \mathbb{R}^2 representing the vessel
- **Model:** Diagonal Gaussian mixture + uniform (noise)
- **Groups:** sound locations = vessel defects



Units: changing the data units

- Principle of **data units transformation** **u**:

$$\begin{aligned} \mathbf{u} : \quad \mathcal{X} = \mathcal{X}^{\text{id}} &\longrightarrow \mathcal{X}^{\mathbf{u}} \\ \mathbf{x} = \mathbf{x}^{\text{id}} = \text{id}(\mathbf{x}) &\longmapsto \mathbf{x}^{\mathbf{u}} = \mathbf{u}(\mathbf{x}) \end{aligned}$$

- **u** is a **bijective** mapping to preserve the whole data set information quantity
- We denote by \mathbf{u}^{-1} the reciprocal of **u**, so $\mathbf{u}^{-1} \circ \mathbf{u} = \text{id}$
- Thus, **id** is only a particular unit **u**
- Often a **meaningful** restriction⁷ on **u**: it proceeds lines by lines and rows by rows

$$\mathbf{u}(\mathbf{x}) = (\mathbf{u}(\mathbf{x}_1), \dots, \mathbf{u}(\mathbf{x}_n)) \quad \text{with} \quad \mathbf{u}(\mathbf{x}_i) = (\mathbf{u}_1(x_{i1}), \dots, \mathbf{u}_d(x_{id}))$$

- Advantage to respect the variable definition, transforming only its unit
- $\mathbf{u}(\mathbf{x}_i)$ means that **u** applied to the data set \mathbf{x}_i , restricted to the single individual i
- \mathbf{u}_j corresponds to the specific (bijective) transformation unit associated to variable j

⁷Possibility to relax this restriction, including for instance linear transformations involved in PCA (principal component analysis). But the variable definition is no longer respected.

Units: revisiting units as a modelling component

- Explicitly exhibiting the “canonical” unit **id** in the model

$$p_m = \{\cdot \in \mathcal{X} \mapsto p(\cdot; \theta) : \theta \in \Theta_m\} = \{\cdot \in \mathcal{X}^{\text{id}} \mapsto p(\cdot; \theta) : \theta \in \Theta_m\} = p_m^{\text{id}}$$

- Thus the variable space and the probability measure are **embedded**
- As the **standard probability theory**: a couple (variable space, probability measure)!
- Changing **id** into **u**, while preserving **m**, is expected to produce a new modelling

$$p_m^u = \{\cdot \in \mathcal{X}^u \mapsto p(\cdot; \theta) : \theta \in \Theta_m\}.$$

A model should be systematically defined by a couple (u, m) , denoted by p_m^u

Units: interpretation and identifiability of p_m^u

- Standard probability theory (again): there exists a measure $u^{-1}(m)$ s.t.⁸

$$u^{-1}(m) \in \{m' \in \mathbb{M} : p_{m'}^{\text{id}} = p_m^u\}$$

- There exists **two alternative interpretations** of strictly the same model:
 - p_m^u : data measured with **unit u** arise from **measure m**;
 - $p_{u^{-1}(m)}^{\text{id}}$: data measured with **unit id** arise from **measure $u^{-1}(m)$**
- Two points of view:

Statistician

The model p_m^u is not identifiable over the couple (m, u)

Practitioner

Freedom to choose the interpretation which is the most meaningful for him

⁸This set is usually restricted to a single element

Units: opportunity for designing new models

Great opportunity to **build** easily numerous new **meaningful models** p_m^u !

- Just **combine** a standard model family $\{\mathbf{m}\}$ with a standard unit family $\{\mathbf{u}\}$
- New family can be huge! **Combinatorial problems** can occur. . .
- **Some model stability** can exist in some (specific) cases: $\mathbf{m} = \mathbf{u}^{-1}(\mathbf{m})$

Units: model selection

As any model, possible to choose between $p_{m_1}^{u_1}$ and $p_{m_2}^{u_2}$

However, caution when using likelihood-based model selection criteria (as BIC)

- **Prohibited** to compare m_1 in unit u_1 and m_2 in unit u_2
- But **allowed** after transforming in **identical unit id**
- Thus compare their equivalent expression: $p_{u_1^{-1}(m_1)}^{id}$ and $p_{u_2^{-1}(m_2)}^{id}$
- Example for abs. continuous x and differentiable u , the **density transform** in **id** is:

$$p_{u^{-1}(m)}^{id} = \{ \cdot \in \mathcal{X}^{id} \mapsto p(u(\cdot); \theta) \times |J^u(\cdot)| : \theta \in \Theta_m \}$$

with $J^u(\cdot)$ the **Jacobian** associated to the transformation u

Units: prostate cancer data (1/2)

- **Individuals:** 506 patients with prostatic cancer grouped on clinical criteria into two Stages 3 and 4 of the disease
- **Variables:** $d = 12$ pre-trial variates were measured on each patient, composed by
 - **Eight continuous** variables (age, weight, systolic blood pressure, diastolic blood pressure, serum haemoglobin, size of primary tumour “SZ”, index of tumour stage and histologic grade, serum prostatic acid phosphatase “AP”)
 - **Two ordinal** variables (performance rating, cardiovascular disease history)
 - **Two categorical** variables with various numbers of levels (electrocardiogram code, bone metastases)
- Some **missing data:** 62 missing values ($\approx 1\%$)
- Two historical units for performing the clustering task:
 - **Raw units id:** [McParland & Gormley, 2015]⁹
 - **Transformed data u:** since SZ and AP are skewed, [Jorgensen & Hunt, 1996]¹⁰ propose

$$\mathbf{u}_{SZ} = \sqrt{\cdot} \text{ and } \mathbf{u}_{AP} = \ln(\cdot)$$

⁹McParland, D. and Gormley, I. C. (2015). Model based clustering for mixed data: clustmd. arXiv preprint arXiv:1511.01720.

¹⁰Jorgensen, M. and Hunt, L. (1996). Mixture model clustering of data sets with categorical and continuous variables. In Proceedings of the Conference ISIS, volume 96, pages 375–384.

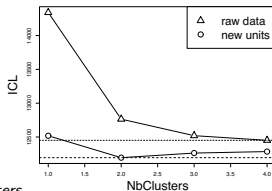
Units: prostate cancer data (2/2)

- **Model m**: full mixed data $\mathbf{x} = (\mathbf{x}^{cont}, \mathbf{x}^{cat}, \mathbf{x}^{ordi}, \mathbf{x}^{int}, \mathbf{x}^{rank})$ (missing data are allowed also) are simply modeled by **inter conditional independence**

$$p(\mathbf{x}; \alpha_k) = p(\mathbf{x}^{cont}; \alpha_k^{cont}) \times p(\mathbf{x}^{cat}; \alpha_k^{cat}) \times p(\mathbf{x}^{ordi}; \alpha_k^{ordi}) \times \dots$$

In addition, for symmetry between types, **intra conditional independence** for each

- **Results**:
 - New units \mathbf{u}_{SZ} and \mathbf{u}_{AP} are selected by ICL
 - New units allow to select **two groups** and provides a **lower error rate**



clusters	
1	2
287	5
52	162

clusters	
1	2
270	22
23	191

Table: Raw units: **11%** misclassified

Table: New units: **9%** misclassified

Variable selection for Gaussians¹¹

Definition

$$p(\mathbf{x}_1; \boldsymbol{\theta}) = \underbrace{\left\{ \sum_{k=1}^K \pi_k p(\mathbf{x}_1^S; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right\}}_{\text{clustering variables}} \times \underbrace{\left\{ p(\mathbf{x}_1^U; \mathbf{a} + \mathbf{x}_1^R \mathbf{b}, \mathbf{C}) \right\}}_{\text{redundant variables}} \times \underbrace{\left\{ p(\mathbf{x}_1^W; \mathbf{u}, \mathbf{V}) \right\}}_{\text{independent variables}}$$

where

- all parts are Gaussians
- S : set of variables useful for clustering
- U : set of redundant clustering variables, expressed with $R \subseteq S$
- W : set of variables independent of clustering

Trick

Variable selection is recasted as a particular model selected by BIC

¹¹Raftery and Dean (2006), Maugis *et al.* (09a), Maugis *et al.* (09b)

Need to formalize
○○○○○○○

Formalizing estimation
○○○○○○○○○○○○○○○○
○○○○○○○○○○○○○○○○
○○○○○○○○○○

Formalizing selection
○○○○○○○○○○○○○
○○○○○○○○○○○○○

More advanced formalizing
○○○○○○○○○○○

Experiments
●○○○○○○○○○
○○○○○○○○○

To go further
○○○

Outline

- 1 Need to formalize
- 2 Formalizing estimation
- 3 Formalizing selection
- 4 More advanced formalizing
- 5 Experiments**
- 6 To go further

(R)MixtComp package

MixtComp

 [Visit notebooks repo](#)

MixtComp takes mixture model analysis one step further and deals with mixed, missing or uncertain data which are common in today's data sets. Mixed data concerned by MixtComp include continuous, categorical, integer and functional (as time series) ones. All of them can be combined, offering many possibilities as multivariate time series, *etc.*

A related publication for this software is still ongoing but you can find early information [here](#). [Here](#) is the related package on the CRAN.

Two ways to be used

- On the CRAN:
<https://cran.r-project.org/web/packages/RMixtComp/index.html>
- As a note book (nothing to install):
<https://team.inria.fr/modal/software/notebooks/>

Example 1: prostate cancer data¹²

- **Individuals:** 506 patients with prostatic cancer grouped on clinical criteria into two Stages 3 and 4 of the disease
- **Variables:** $d = 12$ pre-trial variates were measured on each patient, composed by **eight continuous** variables (age, weight, systolic blood pressure, diastolic blood pressure, serum haemoglobin, size of primary tumour, index of tumour stage and histologic grade, serum prostatic acid phosphatase) and **four categorical** variables with various numbers of levels (performance rating, cardiovascular disease history, electrocardiogram code, bone metastases)
- Some **missing data:** 62 missing values ($\approx 1\%$)

We forget the classes (Stages of the disease) for performing **clustering**

Questions

- How many clusters?
- Which partition?

¹²Byar DP, Green SB (1980): Bulletin Cancer, Paris 67:477-488

RMixtComp basic example

Data

Load the CSV data file as dataframe.

```
[1]: data <- read.table("mixtcomp-example.csv", sep = ";", header = TRUE)  
head(data)
```

A data.frame: 6 × 12

Age	Wt	PF	HX	SBP	DBP	EKG	HG	SZ	SG	AP	BM
<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<dbl>	<chr>
75	76	1	1	15	9	5	138	1.4142	8	1.0986	1
76	?	?	?	?	?	?	?	5.3852	9	2.4849	?
54	116	1	1	13	7	4	146	6.4807	?	1.9459	1
69	102	1	2	14	8	5	134	1.7321	9	1.0986	1
66	?	?	?	?	?	?	?	1	9	2.3979	?
75	94	2	2	14	7	2	176	2	8	2.1972	1

Clustering with RMixtComp

Launch the RMixtComp package.

```
[2]: library(RMixtComp)
```

Loading required package: RMixtCompUtilities

Define the distribution used for each variable.

```
[3]: model <- list(Age = "Gaussian", Wt = "Gaussian", PF = "Multinomial",  
  HX = "Multinomial", SBP = "Gaussian", DBP = "Gaussian",  
  EKG = "Multinomial", HG = "Gaussian", SZ = "Gaussian",  
  SG = "Gaussian", AP = "Gaussian", BM = "Multinomial")
```

Define the SEM algorithm's parameters

```
[4]: algo <- list(nbBurnInIter = 50,  
  nbIter = 100,  
  nbGibbsBurnInIter = 50,  
  nbGibbsIter = 100,  
  nInitPerClass = floor(nrow(data)/2),  
  nSemTry = 5,  
  confidenceLevel = 0.95,  
  ratioStableCriterion = 0.99,  
  nStableCriterion = 10)
```

Choose the desired number of classes and the number of runs for each given number of classes.

```
[5]: nClass <- 1:8  
  nRun <- 3
```

```
[8]: res <- mixtCompLearn(data, model, algo, nClass = nClass, criterion = "ICL", nRun = nRun, nCore = 1)  
  
===== Run MixtComp in learn mode with 3 run(s) per number of classes and 1  
core(s)  
Data: 506 individuals and 12 variables.  
-- K = 1  
Start run 1 on thread number 83  
Run 1 DONE on thread number 83  
Start run 2 on thread number 83  
Run 2 DONE on thread number 83  
Start run 3 on thread number 83  
Run 3 DONE on thread number 83  
Run completed in 0.222s  
-- K = 2  
Start run 1 on thread number 83  
Run 1 DONE on thread number 83  
Start run 2 on thread number 83  
Run 2 DONE on thread number 83  
Start run 3 on thread number 83  
Run 3 DONE on thread number 83  
Run completed in 0.52s  
-- K = 3  
Start run 1 on thread number 83  
Run 1 DONE on thread number 83  
Start run 2 on thread number 83  
Run 2 DONE on thread number 83  
Start run 3 on thread number 83  
Run 3 DONE on thread number 83  
Run completed in 0.65s  
  
...  
-- K = 8  
Start run 1 on thread number 83  
Run 1 DONE on thread number 83  
Start run 2 on thread number 83  
Run 2 DONE on thread number 83  
Start run 3 on thread number 83  
Run 3 DONE on thread number 83  
Run completed in 1.161s  
Total runtime: 6.864s  
Best model according to ICL: 2 clusters.
```

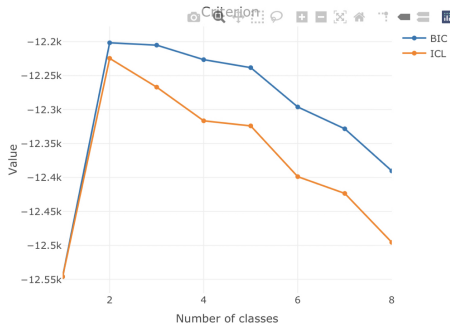
Output's Analysis

...

Criterion

Draw the criterion value (BIC and ICL) for each model that was built. The higher the value (close to 0) the better the model.

```
[9]: plotCrit(res, pkg = "plotly")
```



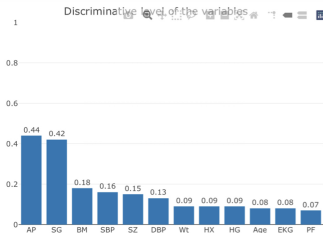
Choose the number of classes to study in the following.

```
[10]: K <- 3
      resK <- extractMixtCompObject(res, K)
```

Variables

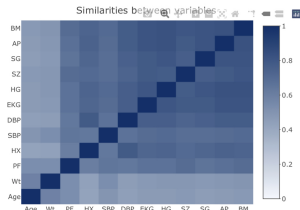
Draw the discriminating level of each variable. A high value (close to one) means that the variable is highly discriminating. A low value (close to zero) means that the variable is poorly discriminating.

```
[11]: plotDiscrimVar(resK, pkg = "plotly")
```



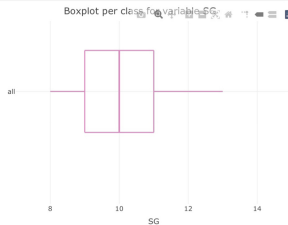
Draw the similarity between every pair of variable. A high value (close to one) means that the two variables provide the same information for the clustering task (i.e. similar partitions). A low value (close to zero) means that the two variables provide some different information for the clustering task (i.e. different partitions).

```
[12]: heatmapVar(resK, pkg = "plotly")
```



Select a variable to draw its distribution.

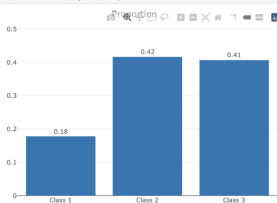
```
[13]: variable <- "SG"
plotDataBoxplot(resK, variable, grl = TRUE, pkg = "plotly")
```



Classes

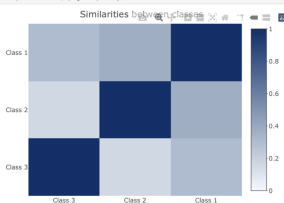
Draw the proportion of individuals in each class.

```
[14]: plotProportion(resK, pkg = "plotly")
```



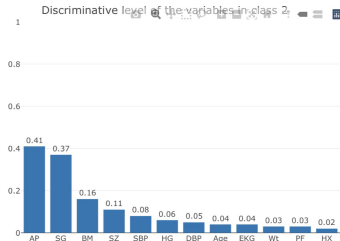
Draw the similarity level between each pair of classes. A high value (close to one) means that the 2 classes are strongly different (i.e. low overlapping). A low value (close to zero) means that the 2 classes are similar for the clustering task (i.e. high overlapping).

```
[15]: heatmapClass(resK, pkg = "plotly")
```



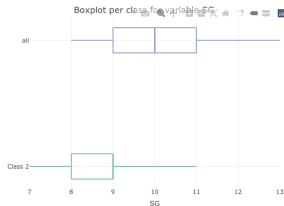
Draw the discriminating level of each variable for the selected class.

```
[16]: class <- 2  
plotDiscrimVar(resK, class = class, pkg = "plotly")
```



Select a variable to draw its distribution for the selected class.

```
[17]: variable <- "SG"  
plotDataBoxplot(resK, variable, class = class, grl = TRUE, pkg = "plotly")
```

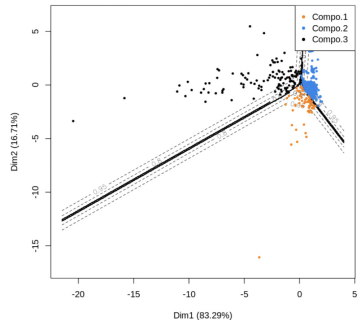


Probabilities

Draw the probability of assignment to a class for each individual. Individuals have been reordered in decreasing assignment probability.

```
[18]: heatmapTikSorted(resK, pkg = "plotly")
```





Example 2: Canadian weather data¹³

- This dataset is extracted from the `fda` package (Ramsay, Wickham, Graves, and Hooker 2018). It contains daily **temperature** and **precipitation** at 35 different locations in Canada averaged over 1960 to 1994.
- Data are contained in a list of 3 elements: `tempav`, a 365×35 matrix where one column corresponds to the daily temperature, `precav` a 365×35 matrix where one column corresponds to the daily precipitation and `time`, a vector containing the index of the day.

¹³Ramsay JO, Silverman BW (2005). Functional Data Analysis. Springer Series in Statistics, 2nd edition. Springer. ISBN 038740080X.

RMixtComp Canadian Weather example

Unsupervised clustering with functional data

```
In [1]: library(RMxtComp)
```

Loading required package: RMixtCompUtilities

Discover the data

Load the Canadian Weather dataset included in the package.

```
In [2]: data(CanadianWeather)
```

This dataset contains daily temperature and precipitation at 35 different locations in Canada averaged over 1960 to 1994. Once imported you have access to a `CanadianWeather` object containing 5 elements:

- `temperav`, an average of temperatures for each day of the year;

```
In [3]: head(CanadianWeather$temperav)
```

St. Johns	Halifax	Sydney	Yarmouth	Charlotttvl	Fredericton	Schefferville	Arvida	Baguetteville	Quebec	...	Vancouver	Victoria
A matrix: 6 × 2 of type dbl												
-3.6	-4.4	-3.8	-1.4	-5.8	-7.9	-22.5	-14.1	-14.6	-10.8	...	2.3	
-3.1	-4.2	-3.5	-1.6	-5.6	-7.5	-23.0	-14.4	-14.7	-11.4	...	2.1	
-3.4	-5.3	-4.6	-2.5	-7.3	-9.3	-23.0	-15.0	-15.5	-12.5	...	1.9	
-4.4	-5.4	-5.0	-2.3	-7.0	-8.7	-21.8	-14.3	-14.3	-11.2	...	2.0	
-2.9	-5.6	-4.1	-2.4	-6.7	-9.1	-23.5	-16.2	-15.8	-12.0	...	1.6	
-4.5	-7.1	-6.1	-3.7	-8.9	-10.9	-24.4	-16.3	-16.9	-13.2	...	1.4	

- `precipav`, an average of precipitation for each day of the year;

```
In [4]: head(CanadianWeather$precipav)
```

St. Johns	Halifax	Sydney	Yarmouth	Charlotttvl	Fredericton	Schefferville	Arvida	Baguetteville	Quebec	...	Vancouver	Victoria
A matrix: 6 × 35 of type dbl												
5.2	6.0	5.3	5.6	4.6	4.0	1.1	2.6	3.0	4.1	...	5.5	
5.8	5.3	5.2	3.7	4.4	3.2	1.3	1.2	1.8	2.3	...	6.6	
3.9	2.6	2.1	2.8	2.3	3.3	1.2	2.1	1.3	2.6	...	6.8	
4.3	5.3	5.0	5.3	4.8	3.3	1.3	2.3	2.5	4.3	...	5.1	
6.2	6.0	7.3	3.8	5.1	2.7	1.0	1.7	2.1	2.3	...	3.8	
3.4	2.1	2.2	2.4	1.5	0.8	1.3	2.0	1.6	1.5	...	2.5	

- `coordinates`, the coordinates of each location;

```
In [5]: head(CanadianWeather$coordinates)
```

	N.latitude	W.longitude
A matrix: 6 × 2 of type dbl		
St. Johns	47.34	52.43
Halifax	44.39	63.36
Sydney	46.09	60.11
Yarmouth	43.50	66.07
Charlotttvl	42.48	80.25
Fredericton	45.58	66.39

- `region`, the climate zone of each location;

```
In [6]: head(CanadianWeather$region)
```

St. Johns	'Atlantic'
Halifax	'Atlantic'
Sydney	'Atlantic'
Yarmouth	'Atlantic'
Charlotttvl	'Atlantic'
Fredericton	'Atlantic'

- and `time`, a time index for days over a year, from 1 to 365.

```
In [7]: head(CanadianWeather$time)
```

```
1. 1
2. 2
3. 3
4. 4
5. 5
6. 6
```


Clustering with RMixtComp

Define the distribution used for each variable and the associated hyperparameters. The functional model requires 2 hyperparameters: `nSub` is the number of subregressions into which the function will be decomposed; `nCoeff` is the number of polynomial coefficients of each subregression (2 = line).

```
In [12]: nSub <- 4
nCoeff <- 2
func <- list(type = "Func_CS", paramStr = paste0("nSub: ", nSub, ", nCoeff: ", nCoeff))
model <- list(tempav = func, precav = func)
```

Define the SEM algorithm's parameters

```
In [13]: algo <- createAlgo()
```

Choose the desired number of classes and the number of runs for each given number of classes.

```
In [14]: nClass <- 2:4
nRun <- 3
```

```
In [15]: res <- mixtCompLearn(data, model, algo, nClass = nClass, criterion = "ICL", nRun = nRun, nCore = 2)
```

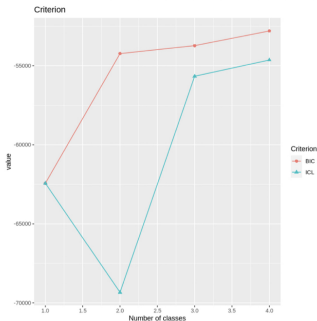
```
===== Run Hierarchical MixtComp in learn mode with 3 run(s) per number of classes and 2 core(s)
Data: 35 individuals and 3 variables.
-- K = 1
Run time: 32.274s
-- K = 2
Number of splits to perform: 1
Split a cluster in two
Compute criterion
Run time: 32.655s
-- K = 3
Number of splits to perform: 2
Split a cluster in two
Split a cluster in two
Compute criterion
Run time: 96.874s
-- K = 4
Number of splits to perform: 2
Split a cluster in two
Split a cluster in two
Compute criterion
Run time: 86s
Total runtime: 247.804s
Best model according to ICL: 4 clusters.
```

Output's Analysis

Criterion

This chart represents the criterion value (choose between BIC and ICL) for each model that was built. The higher the value (close to 0) the better the model.

In [16]: `plotCrit(res)`

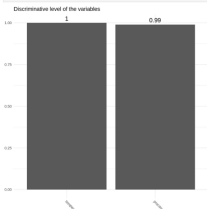


Choose the number of classes to study in the following.

In [17]: `K <- 3`
`resK <- extractMixtCompObject(res, K)`

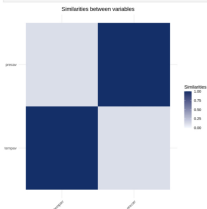
This chart represents the discriminating level of each variable. A high value (close to one) means that the variable is highly discriminating. A low value (close to zero) means that the variable is poorly discriminating.

```
plotDiscrimVar(resK)
```



This graph displays the similarity between every pair of variable. A high value (close to one) means that the two variables provide the same information for the clustering task (i.e. similar partitions). A low value (close to zero) means that the two variables provide some different information for the clustering task (i.e. different partitions).

```
heatmapVar(resK)
```

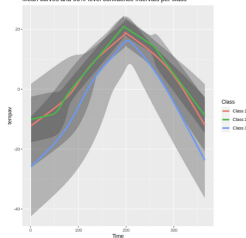


Draw the distribution of the two variables.

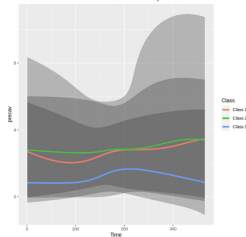
In [20]:

```
plotDataCI(resK, "tempav")
plotDataCI(resK, "precav")
```

Mean curves and 95%-level confidence intervals per class



Mean curves and 95%-level confidence intervals per class



Need to formalize
○○○○○○○

Formalizing estimation
○○○○○○○○○○○○○○
○○○○○○○○○○○○○○○
○○○○○○○○○○○○○

Formalizing selection
○○○○○○○○○○○○○
○○○○○○○○○○○○○

More advanced formalizing
○○○○○○○○○○○

Experiments
○○○○○○○○○○○
○○○○○○○○○●

To go further
○○○

Graphic visualization on a map with $K = 4$

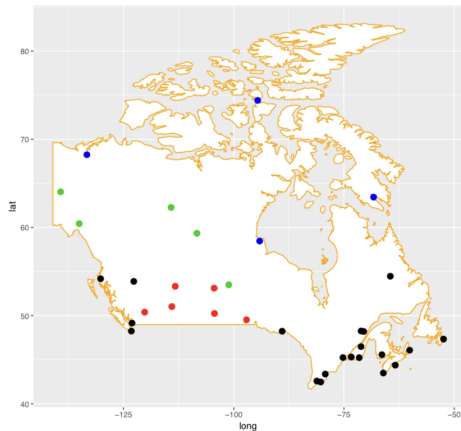


Figure 6: Geographic positions of the stations. The colour indicates the class: black = 1, red = 2, green = 3 and blue = 4.

Need to formalize
○○○○○○○

Formalizing estimation
○○○○○○○○○○○○○○○○
○○○○○○○○○○○○○○○○
○○○○○○○○○○

Formalizing selection
○○○○○○○○○○○○○
○○○○○○○○○○○○○

More advanced formalizing
○○○○○○○○○○○

Experiments
○○○○○○○○○○○
○○○○○○○○○

To go further
●○○

Outline

1 Need to formalize

2 Formalizing estimation

3 Formalizing selection

4 More advanced formalizing

5 Experiments

6 To go further

Need to formalize
○○○○○○○

Formalizing estimation
○○○○○○○○○○○○○○
○○○○○○○○○○○○○○○
○○○○○○○○○○○

Formalizing selection
○○○○○○○○○○○○○
○○○○○○○○○○○

More advanced formalizing
○○○○○○○○○○○

Experiments
○○○○○○○○○○○
○○○○○○○○○

To go further
○●○

Some remaining questions

- More on dependent data (like times series)
- High-dimensional data
- Missing not at random data (MNAR)
- ...

Need to formalize
○○○○○○○

Formalizing estimation
○○○○○○○○○○○○○○○
○○○○○○○○○○○○○○○
○○○○○○○○○○○○○○○
○○○○○○○○○○○○○

Formalizing selection
○○○○○○○○○○○○○
○○○○○○○○○○○○○
○○○○○○○○○○○○○

More advanced formalizing
○○○○○○○○○○○

Experiments
○○○○○○○○○○○
○○○○○○○○○○○
○○○○○○○○○○○

To go further
○○●

Next lesson

Clustering : une vision unifiée pour une utilisation éclairée
Traitement de la grande dimension & co-clustering