



HAL
open science

About Two Disinherited Sides of Statistics: Data Units and Computational Saving

Christophe Biernacki, Alexandre Lourme, Maxime Brunin, Alain A. Celisse

► **To cite this version:**

Christophe Biernacki, Alexandre Lourme, Maxime Brunin, Alain A. Celisse. About Two Disinherited Sides of Statistics: Data Units and Computational Saving. Statlearn 2017, Apr 2017, Lyon, France. pp.1-56. hal-01665905

HAL Id: hal-01665905

<https://inria.hal.science/hal-01665905v1>

Submitted on 18 Dec 2017

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.

About Two Disinherited Sides of Statistics: Data Units and Computational Saving

C. Biernacki

8th ed. of the STATEARN workshop "Challenging problems in Statistical Learning"
April 6-7, 2017, Lyon (France)



Synopsis of the talk

$$\widehat{\text{target}} = \mathbf{f}(\underbrace{\text{data, model}}_{\text{Part I}}, \underbrace{\text{algo}}_{\text{Part II}})$$

○○○○
○○○
○○○
○○○○○

○○○
○○○○
○○○○○

Part I

Unifying Data Units and Models in Statistics

Focus on (Co)-Clustering

Joint work with A. Lourme
(Bordeaux University)

○○○○
○○○
○○○
○○○○○○○○
○○○○
○○○○○

Quizz!

$$y = \beta x^2 + e$$

- Is it a **linear** regression on co-variates (x^2)?
- Is it a **quadratic** regression on co-variates x ?

Both!

○○○○
○○○
○○○
○○○○○○○○
○○○○
○○○○○
○○○○○

Take home message

Units are entirely interrelated with models

This part:

- Be aware that interpretation of (“classical”) models is **unit dependent**
- Models should even be revisited as a **couple units × “classical” models**
- Opportunity for **cheap/wide/meaningful** enlarging of “classical” model families
- Focus on **model-based (co-)clustering** but larger potential impact



General (model-based) statistical framework

■ Data:

- Whole data set composed by n objects, described by d variables

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

- Each \mathbf{x}_i value is provided with a **unit id**
- We note “**id**” since units are often user defined (a kind of canonical units)

■ Model:

- A pdf¹ family, indexed by $\mathbf{m} \in \mathbb{M}^2$

$$p_{\mathbf{m}} = \{ \cdot \in \mathbb{X} \mapsto p(\cdot; \boldsymbol{\theta}) : \boldsymbol{\theta} \in \Theta_{\mathbf{m}} \}$$

- With $p(\cdot; \boldsymbol{\theta})$ a (parametric) pdf and $\Theta_{\mathbf{m}}$ a space where evolves this parameter

■ Target:

$$\widehat{\text{target}} = \mathbf{f}(\mathbf{x}, p_{\mathbf{m}})$$

Unit **id** is hidden everywhere and could have consequences on the target estimation!

¹probability density function

²Often, the index \mathbf{m} is confounded with the distribution family itself as a shortcut

○○○○
○○○
○○○○○

○○○
○○○○
○○○○○

Changing the data units

- Principle of **data units transformation \mathbf{u}** :

$$\mathbf{u} : \begin{array}{l} \mathbb{X} = \mathbb{X}^{\mathbf{id}} \\ \mathbf{x} = \mathbf{x}^{\mathbf{id}} = \mathbf{id}(\mathbf{x}) \end{array} \begin{array}{l} \longrightarrow \\ \longmapsto \end{array} \begin{array}{l} \mathbb{X}^{\mathbf{u}} \\ \mathbf{x}^{\mathbf{u}} = \mathbf{u}(\mathbf{x}) \end{array}$$

- \mathbf{u} is a **bijective** mapping to preserve the whole data set information quantity
- We denote by \mathbf{u}^{-1} the reciprocal of \mathbf{u} , so $\mathbf{u}^{-1} \circ \mathbf{u} = \mathbf{id}$
- Thus, \mathbf{id} is only a particular unit \mathbf{u}
- Often a **meaningful** restriction³ on \mathbf{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 \mathbf{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

³Possibility to relax this restriction, including for instance linear transformations involved in PCA (principal component analysis). But the variable definition is no longer respected.



Revisiting units as a modelling component

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

$$p_{\mathbf{m}} = \{\cdot \in \mathbb{X} \mapsto p(\cdot; \boldsymbol{\theta}) : \boldsymbol{\theta} \in \Theta_{\mathbf{m}}\} = \{\cdot \in \mathbb{X}^{\text{id}} \mapsto p(\cdot; \boldsymbol{\theta}) : \boldsymbol{\theta} \in \Theta_{\mathbf{m}}\} = p_{\mathbf{m}}^{\text{id}}$$

- 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

$$p_{\mathbf{m}}^{\mathbf{u}} = \{\cdot \in \mathbb{X}^{\mathbf{u}} \mapsto p(\cdot; \boldsymbol{\theta}) : \boldsymbol{\theta} \in \Theta_{\mathbf{m}}\}.$$

A model should be systematically defined by a couple **(u,m)**, denoted by $p_{\mathbf{m}}^{\mathbf{u}}$



Interpretation and identifiability of p_m^u

- Standard probability theory (again): there exists a measure $u^{-1}(m)$ s.t.⁴

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

- 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

⁴This set is usually restricted to a single element

○○○○
○○○
○○○
○○○○○○○○
○○○○
○○○○○

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: $\mathbf{m} = \mathbf{u}^{-1}(\mathbf{m})$



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: $p_{u_1^{-1}(m_1)}^{\text{id}}$ and $p_{u_2^{-1}(m_2)}^{\text{id}}$
- Example for abs. continuous x and differentiable u , the **density transform** in **id** is:

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

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



Focus on the clustering target

A current challenge is to enlarge model collection. . . and units could contribute to it!

- **Model:** mixture model \mathbf{m} of parameter $\boldsymbol{\theta} = \{\pi_k, \boldsymbol{\alpha}_k\}_{k=1}^g$

$$p_{\mathbf{m}}(\mathbf{x}; \boldsymbol{\theta}) = \sum_{k=1}^g \pi_k p(\mathbf{x}; \boldsymbol{\alpha}_k)$$

- g is the number of clusters
- Clusters correspond to a hidden partition $\mathbf{z} = (z_1, \dots, z_n)$, where $z_i \in \{1, \dots, g\}$
- $\pi_k = p(Z = k)$ and $p(\mathbf{x}; \boldsymbol{\alpha}_k) = p(\mathbf{X} = \mathbf{x} | Z = k)$
- **Target:** estimate \mathbf{z} (and often g)
 - Estimate $\hat{\boldsymbol{\theta}}_{\mathbf{m}}$ by maximum likelihood (typically)
 - Estimate \mathbf{z} by the MAP principle $\hat{z}_i = \arg \max_{k \in \{1, \dots, g\}} p(Z_i = k | \mathbf{X}_i = \mathbf{x}_i; \hat{\boldsymbol{\theta}}_{\mathbf{m}})$
 - Estimate g by BIC or ICL criteria typically (maximum likelihood based criteria)



Outline

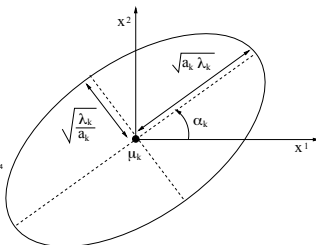
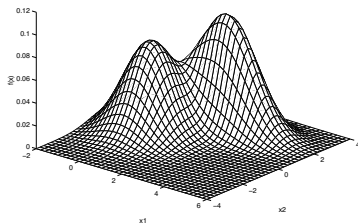
- 1 Introduction
- 2 Units in model-based clustering
 - Scale units and parsimonious Gaussians
 - Non scale units and Gaussians
 - Units and Poissons
- 3 Units in model-based co-clustering
 - Model for different kinds of data
 - Units and Bernoulli
 - Units and multinomial



14 spectral models on Σ_k

- $\mathbf{X} = \mathbb{R}^d$
- d -variate Gaussian model \mathbf{m} : $p_{\mathbf{m}}(\cdot; \alpha_k) = \mathcal{N}_d(\boldsymbol{\mu}_k, \Sigma_k)$
- [Celeux & Govaert, 1995]⁵ propose the following eigen decomposition

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



⁵Celeux, G., and Govaert, G.. Gaussian parsimonious clustering models. Pattern Recognition, 28(5), 781–793 (1995).



Scale unit invariance

- Consider scale unit transformation $\mathbf{u}(\mathbf{x}) = \mathbf{D}\mathbf{x}$, with diagonal $\mathbf{D} \in \mathbb{R}^{d \times d}$
- Very **current transformation**: standard units (mm, cm), standardized units
- [Biernacki & Lourme, 2014] listed models where invariance holds (8 among 14)
 - The general model is invariant:

$$[\lambda_k \mathbf{S}_k \mathbf{\Lambda}_k \mathbf{S}'_k] = \mathbf{u}^{-1}([\lambda_k \mathbf{S}_k \mathbf{\Lambda}_k \mathbf{S}'_k])$$

- An example of not invariant model:

$$[\lambda_k \mathbf{S} \mathbf{\Lambda}_k \mathbf{S}'] \neq \mathbf{u}^{-1}([\lambda_k \mathbf{S} \mathbf{\Lambda}_k \mathbf{S}'])$$

- Do not forget to compare all models $\mathbf{m}' = \mathbf{u}^{-1}(\mathbf{m})$ in **unit id** for BIC / ICL validity
- Use the **Rmixmod** package



Illustration on the Old Faithful geyser data set

- All models are with free proportions (π_k)
- All ICL values are expressed with the initial unit $\mathbf{id} = \min \times \min$
- We observe the [effect of unit on the ICL ranking](#) for some models
- [Cheap](#) opportunity to [enlarge](#) the model family!

family	$\mathbf{id} = (\min, \min)$		$\mathbf{u}^{\text{scale}_1} = (\text{sec}, \min)$		$\mathbf{u}^{\text{scale}_2} = (\text{stand}, \text{stand})$	
	\mathbf{m}	ICL ^{id}	\mathbf{m}	ICL ^{id}	\mathbf{m}	ICL ^{id}
All mod.	$[\lambda_k \mathbf{S} \mathbf{\Lambda}_k \mathbf{S}'_k]$	1 160.3	$[\lambda_k \mathbf{S} \mathbf{\Lambda}_k \mathbf{S}'_k]$	1 158.7	$[\lambda_k \mathbf{S}_k \mathbf{\Lambda}_k \mathbf{S}'_k]$	1 160.3
General mod.	$[\lambda_k \mathbf{S}_k \mathbf{\Lambda}_k \mathbf{S}'_k]$	1 161.4	$[\lambda_k \mathbf{S}_k \mathbf{\Lambda}_k \mathbf{S}'_k]$	1 161.4	$[\lambda_k \mathbf{S}_k \mathbf{\Lambda}_k \mathbf{S}'_k]$	1 161.4



Outline

1 Introduction

2 Units in model-based clustering

- Scale units and parsimonious Gaussians
- **Non scale units and Gaussians**
- Units and Poissons

3 Units in model-based co-clustering

- Model for different kinds of data
- Units and Bernoulli
- Units and multinomial



Prostate cancer data of [Biar & Green, 1980]⁸

- **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]⁶
 - **Transformed data \mathbf{u} :** since SZ and AP are skewed, [Jorgensen & Hunt, 1996]⁷ propose

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

⁶McParland, D. and Gormley, I. C. (2015). Model based clustering for mixed data: clustmd. arXiv preprint arXiv:1511.01720.

⁷Jorgensen, 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.

⁸Byar DP, Green SB (1980): Bulletin Cancer, Paris 67:477-488



Clustering with the MixtComp software [Biernacki et al., 2016]⁹

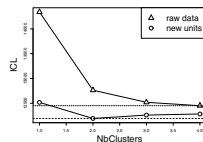
- Model m in Mixtcomp: full mixed data $\mathbf{x} = (\mathbf{x}^{cont}, \mathbf{x}^{cat}, \mathbf{x}^{ordi}, \mathbf{x}^{int}, \mathbf{x}^{rank})$ (missing data are allowed also) are simply modeled by inter conditional independence

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

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

- Results:

- New units u_{SZ} and 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

Table : MixtComp model on raw units: 11% misclassified

clusters	
1	2
270	22
23	191

Table : MixtComp model on new units: 9% misclassified

⁹MixtComp is a clustering software developed by Biernacki C., Iovleff I. and Kubicki V. and freely available on the MASSICCC web platform <https://massiccc.lille.inria.fr/>



Outline

- 1 Introduction
- 2 Units in model-based clustering
 - Scale units and parsimonious Gaussians
 - Non scale units and Gaussians
 - Units and Poissons
- 3 Units in model-based co-clustering
 - Model for different kinds of data
 - Units and Bernoulli
 - Units and multinomial



Which units for count data?

- Count data: $x \in \mathbb{N}$
- Standard model \mathbf{m} is Poisson: $p(\cdot; \alpha_k) = \mathcal{P}(\lambda_k)$
- d -variate case $\mathbf{x} = (x^1, \dots, x^d) \in \mathbb{N}^d$ and conditional independence by variable
- Two standard unit transformations (by variable $j \in \{1, \dots, d\}$):
 - Shifted observations: $\mathbf{u}(x^j) = x^j - a_j$ with $a_j \in \mathbb{N}$
 - Scaled observations: $\mathbf{u}(x^j) = b_j x^j$ with $b_j \in \mathbb{N}^*$

Shifted example

- **id:** total number of educational years
- $\mathbf{u}_{\text{shift}}(\cdot) = (\cdot) - 8$: university number of educational years^a

^aEight is the number of years spent by english pupils in a secondary school.

Scaled example

- **id:** total number of educational years
- $\mathbf{u}_{\text{scaled}}(\cdot) = 2 \times (\cdot)$: total number of educational semesters



Medical data

- R dataset `rwm1984COUNT` of [Rao *et al.*, 2007, p.221]¹⁰ and studied in [Hilbe, 2014]¹¹
- $n = 3874$ patients that spent time into German hospitals during year 1984
- Patients are described through eleven mixed variables
- **m**: a MixtComp model combining Gaussian, Poisson and multinomial distributions

	<i>variables</i>	<i>type</i>	<i>model</i>
1	number of visits to doctor during year	count	Poisson
2	number of days in hospital	count	Poisson
3	educational level	categorical	multinomial
4	age	count	Poisson
5	outwork	binary	Bernoulli
6	gender	binary	Bernoulli
7	matrimonial status	binary	Bernoulli
8	kids	binary	Bernoulli
9	household yearly income	continuous	Gaussian
10	years of education	count	Poisson
11	self employed	binary	Bernoulli

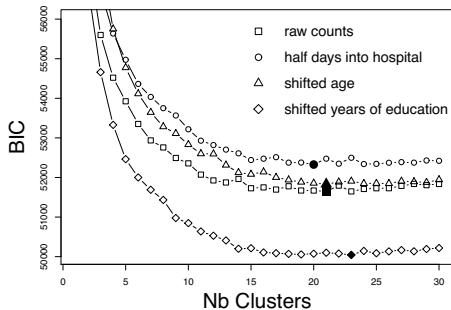
¹⁰Rao, C. R., Miller, J. P., and Rao, D. C. (2007). Handbook of statistics: epidemiology and medical statistics, volume 27. Elsevier.

¹¹Hilbe, J. M. (2014). Modeling count data. Cambridge University Press.



Several units for count data

- **Four unit systems** are sequentially considered differing over the count data
 - $u_1 = \text{id}$: original unit
 - u_2 : the time spent into hospital is counted in half days instead of days
 - u_3 : the minimum of the age series is deduced from all ages leading to shifted ages
 - u_4 : the min. of years of edu. is deduced from the series leading to shifted years of edu.
- BIC selects 23 clusters obtained under **shifted years** of education





Specific transformation for RNA-seq data

- A sample of RNA-seq gene expressions arising from the rat count table of <http://bowtie-bio.sourceforge.net/recount/>
- 30000 genes described by 22 **counting** descriptors
- Remove genes with low expression (classical): 6173 genes finally
- Two different processes for dealing with data:
 - **Standard** [Rau *et al.*, 2015]¹²: $\mathbf{u} = \mathbf{id}$ and \mathbf{m} is Poisson mixture
 - **"RNA-seq unit"** [Gallopin *et al.*, 2015]¹³:

$$\mathbf{u}(\cdot) = \ln(\text{scaled normalization}(\cdot))$$

is a transformation being motivated by genetic considerations and \mathbf{m} is Gaussian mixture

- Experiment with 30 clusters (as in [Gallopin *et al.*, 2015])

<i>model</i>	<i>data</i>	<i>BIC</i>
Poisson	raw unit	2 615 654
Gaussian	transformed	909 190

¹²Rau, A., Maugis-Rabusseau, C., Martin-Magniette, M.-L. and Celeux, G. (2015). Co-expression analysis of high-throughput transcriptome sequencing data with Poisson mixture models. *Bioinformatics*, 31 (9), 1420-1427.

¹³Gallopin, M., Rau, A., Celeux, G., and Jaffrézic, F. (2015). Transformation des données et comparaison de modèles pour la classification des données rna-seq. In 47èmes Journées de Statistique de la SFdS.



Outline

1 Introduction

2 Units in model-based clustering

- Scale units and parsimonious Gaussians
- Non scale units and Gaussians
- Units and Poissons

3 Units in model-based co-clustering

- Model for different kinds of data
- Units and Bernoulli
- Units and multinomial



Co-clustering framework

- It corresponds to the following **specific mixture model** **m** [Govaert and Nadif, 2014]¹⁴:

$$p(\mathbf{x}; \theta) = \sum_{(\mathbf{z}, \mathbf{w})} \prod_{i,j} \pi_{z_i} \rho_{w_j} p(x_i^j; \alpha_{z_i w_j})$$

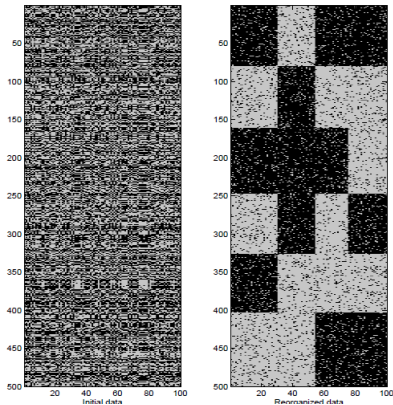
- z**: partition in g_r rows
- w**: partition in g_c columns
- z** \perp **w** and $x_i^j | (z_i, w_j) \perp x_{i'}^{j'} | (z_{i'}, w_{j'})$
- Distribution $p(\cdot; \alpha_{z_i w_j})$ depends on the kind of data
 - Binary** data: $x_i^j \in \{0, 1\}$, $p(\cdot; \alpha_{kl}) = \mathcal{B}(\alpha_{kl})$
 - Categorical** data with m levels:
 - $\mathbf{x}_i^j = \{x_i^{jh}\} \in \{0, 1\}^m$ with $\sum_{h=1}^m x_i^{jh} = 1$ and $p(\cdot; \alpha_{kl}) = \mathcal{M}(\alpha_{kl})$ with $\alpha_{kl} = \{\alpha_k^{jh}\}$
 - Count** data: $x_i^j \in \mathbb{N}$, $p(\cdot; \alpha_{kl}) = \mathcal{P}(\mu_k \nu_l \gamma_{kl})$
 - Continuous** data: $x_i^j \in \mathbb{R}$, $p(\cdot; \alpha_{kl}) = \mathcal{N}(\mu_{kl}, \sigma_{kl}^2)$
- BlockCluster** [Bhatia et al., 2015]¹⁵ is an R package for co-clustering

¹⁴G. Govaert and M. Nadif (2014). Co-clustering: models, algorithms and applications. ISTE, Wiley. ISBN 978-1-84821-473-6.

¹⁵P. Bhatia, S. Iovleff, G. Govaert (2015). Blockcluster: An R Package for Model Based Co-Clustering. *Journal of Statistical Software*, in press.



Binary illustration



○○○○
○○○
○○○
○○○○○○○○
●○○○
○○○○○

Outline

1 Introduction

2 Units in model-based clustering

- Scale units and parsimonious Gaussians
- Non scale units and Gaussians
- Units and Poissons

3 Units in model-based co-clustering

- Model for different kinds of data
- **Units and Bernoulli**
- Units and multinomial



SPAM E-mail Database¹⁷

- $n = 4601$ e-mails composed by 1813 “spams” and 2788 “good e-mails”
- $d = 48 + 6 = 54$ continuous descriptors¹⁶
 - 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**

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

Two different units considered for variable $j \in \{1, \dots, 54\}$

- id_j : see the previous coding
- $\mathbf{u}_j(\cdot) = 1 - (\cdot)$: reverse the coding

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

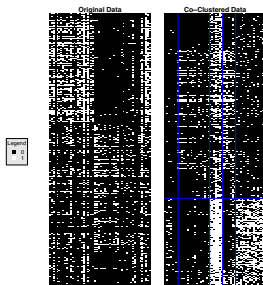
¹⁶There are 3 other continuous descriptors we do not use

¹⁷<https://archive.ics.uci.edu/ml/machine-learning-databases/spambase/>

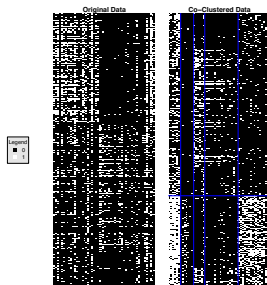


Select the whole coding $\mathbf{u} = (\mathbf{u}_1, \dots, \mathbf{u}_d)$

- Fix $g_l = 2$ (two individual classes) and $g_r = 5$ (five variable classes)
- Use co-clustering in a **clustering aim**: just interested in indiv. classes (spams?)
- Use a “naive” algorithm to find the **best \mathbf{u}** by ICL (2^{54} possibilities)



initial unit id
ICL=92682.54
error rate=0.1984



best unit \mathbf{u}
ICL=92524.57
error rate=0.2008

○○○○
○○○
○○○
○○○○○

○○○
○○○●
○○○○○

Result analysis of the e-mail database

- Just one variable ($j = 19$: “you”) has a reversed coding in \mathbf{u}
- Thus variable “you” has **not the same coding as other variables** in its column class
- Poor ICL increase with \mathbf{u}

Conclusion for the e-mail database

- Here initial units \mathbf{id} have a particular **meaning for the user**: do not change!
- In case of unit change, it becomes **essentially technic** (as Manly unit is)



Outline

- 1 Introduction
- 2 Units in model-based clustering
 - Scale units and parsimonious Gaussians
 - Non scale units and Gaussians
 - Units and Poissons
- 3 Units in model-based co-clustering
 - Model for different kinds of data
 - Units and Bernoulli
 - Units and multinomial



Congressional Voting Records Data Set¹⁹

- Votes for each of the $n = 435$ U.S. House of Representatives Congressmen
 - Two classes: 267 democrats, 168 republicans
 - $d = 16$ votes with $m = 3$ modalities [Schlimmer, 1987]¹⁸:
 - “yea”: voted for, paired for, and announced for
 - “nay”: voted against, paired against, and announced against
 - “?”: voted present, voted present to avoid conflict of interest, and did not vote or otherwise make a position known
-
- | | |
|--------------------------------------|--|
| 1. handicapped-infants | 9. mx-missile |
| 2. water-project-cost-sharing | 10. immigration |
| 3. adoption-of-the-budget-resolution | 11. synfuels-corporation-cutback |
| 4. physician-fee-freeze | 12. education-spending |
| 5. el-salvador-aid | 13. superfund-right-to-sue |
| 6. religious-groups-in-schools | 14. crime |
| 7. anti-satellite-test-ban | 15. duty-free-exports |
| 8. aid-to-nicaraguan-contras | 16. export-administration-act-south-africa |

¹⁸Schlimmer, J. C. (1987). Concept acquisition through representational adjustment. Doctoral dissertation, Department of Information and Computer Science, University of California, Irvine, CA.

¹⁹<http://archive.ics.uci.edu/ml/datasets/Congressional+Voting+Records>

○○○○
○○○
○○○○○

○○○
○○○○
○○●○○

Allowed user meaningful recodings

- “yea” and “nea” are arbitrarily coded (**question dependent**), not “?”
- Example:
 3. **adoption**-of-the-budget-resolution = “yes” \Leftrightarrow 3. **rejection**-of-the-budget-resolution = “no”
- However, “?” is **not question dependent**

Thus, two different units considered for variable $j \in \{1, \dots, 16\}$

- id_j :

$$x_i^j = \begin{cases} (1, 0, 0) & \text{if voted “yea” to vote } j \text{ by congressman } i \\ (0, 1, 0) & \text{if voted “nay” to vote } j \text{ by congressman } i \\ (0, 0, 1) & \text{if voted “?” to vote } j \text{ by congressman } i \end{cases}$$

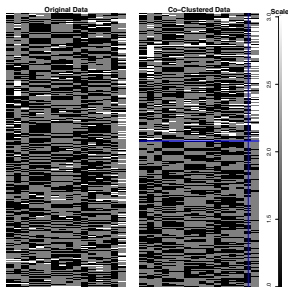
- $\mathbf{u} = (\mathbf{u}_1, \dots, \mathbf{u}_d)$: reverse the coding **only for “yea” and “nea”**

$$\mathbf{u}_j(x_i^j) = \begin{cases} (0, 1, 0) & \text{if voted “yea” to vote } j \text{ by congressman } i \\ (1, 0, 0) & \text{if voted “nay” to vote } j \text{ by congressman } i \\ (0, 0, 1) & \text{if voted “?” to vote } j \text{ by congressman } i \end{cases}$$

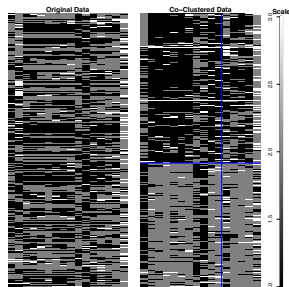


Select the whole coding $\mathbf{u} = (\mathbf{u}_1, \dots, \mathbf{u}_d)$

- Fix $g_l = 2$ (two individual classes) and $g_r = 2$ (two variable classes)
- Use co-clustering in a **clustering aim**: just interested in political party
- Use a comprehensive algorithm to find the **best \mathbf{u} by ICL** ($2^{16} = 65536$ cases)



initial unit id
ICL=5916.13
error rate=0.2850



best unit \mathbf{u}
ICL=5458.156
error rate=0.1034



Result analysis of the Congressional Voting Records Data Set

- Five variables has a reversed coding in \mathbf{u} :
 - 3. adoption-of-the-budget-resolution
 - 7. anti-satellite-test-ban
 - 9. aid-to-nicaraguan-contras
 - 10. mx-missile
 - 16. duty-free-exports
- Thus be aware to change the meaning of them when having a look at the figure!
- Significant **ICL and error rate improvements** with \mathbf{u}

Conclusion for the Congressional Voting Records

- Here initial units \mathbf{id} where arbitrary fixed: make sense to change!
- In addition, good improvement. . .

Part II

Computation Time/Accuracy Trade-off

Focus on Linear Regression

Joint work with M. Brunin & A. Céliste
(Lille University & CNRS & Inria)

An unexpected behaviour...

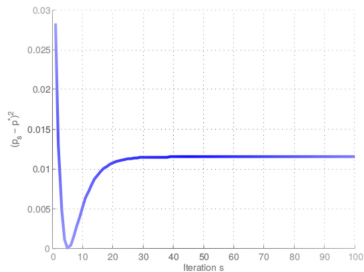
Standard idea

The larger is the iteration number, the better is the resulting estimate

Not so certain...

An **early** stopping rule could **reduce computation** time while **increasing accuracy**

Ex.: two Gaussian univariate mixture, just proportions unknown (convex), use EM



Take home message

Early stopping of some estimation algorithms could be statistically efficient while preserving computational time

This part:

- Identify **bias/variance** influence throughout the algorithm iterations
- Define an **early stopping rule** reaching the bias/variance trade-off
- Focus on linear regression but expected to be (much) **more general**

Outline

- 1 Introduction
- 2 Understanding the algorithm dynamic
- 3 First attempts for a stopping rule
- 4 Numerical simulations

Linear regression

- Usual linear regression model:

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\theta}^* + \boldsymbol{\epsilon},$$

with $\mathbf{X} \in \mathcal{M}_{n,d}(\mathbb{R})$, $\text{rg}(\mathbf{X}) = d$ ($n > d$), $\boldsymbol{\theta}^* \in \mathbb{R}^d$, $\boldsymbol{\epsilon} \sim \mathcal{N}_n(\mathbf{0}, \sigma^2 \mathbf{I}_n)$

- Usual Ordinary Least Squares (OLS) parameter estimate:

$$\hat{\boldsymbol{\theta}} = \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^d} \underbrace{\frac{1}{2n} \|\mathbf{Y} - \mathbf{X}\boldsymbol{\theta}\|_{2,n}^2}_{g(\boldsymbol{\theta})} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$$

- Usual OLS prediction estimate:

$$\hat{\mathbf{Y}} = \mathbf{X}\hat{\boldsymbol{\theta}}$$

- Usual oracle predictive accuracies of $\hat{\boldsymbol{\theta}}$ ($\mathbf{Y}^* = \mathbf{X}\boldsymbol{\theta}^*$, MSE=Mean Squared Error):

$$\Delta(\hat{\mathbf{Y}}) = \frac{1}{n} \left\| \hat{\mathbf{Y}} - \mathbf{Y}^* \right\|_{2,n}^2 \quad \text{or} \quad \text{MSE}(\hat{\mathbf{Y}}) = \mathbb{E} \left[\Delta(\hat{\mathbf{Y}}) \right]$$

Alternative estimate of the OLS

Find an estimator that performs better in terms of predictive accuracy than OLS $\hat{\theta}$

Use a **gradient descent algorithm** to minimise $g(\theta)$ (with fixed step α):

$$\forall k \geq 0, \quad \hat{\theta}^{(k+1)} = \hat{\theta}^{(k)} - \alpha \nabla g(\hat{\theta}^{(k)})$$

- **New** parameter estimate (this one obtained at **iteration k**):

$$\hat{\theta}^{(k)} = \left(\mathbf{I}_d - \left(\mathbf{I}_d - \frac{\alpha}{n} \mathbf{X}'\mathbf{X} \right)^k \right) \hat{\theta} + \left(\mathbf{I}_d - \frac{\alpha}{n} \mathbf{X}'\mathbf{X} \right)^k \theta^{(0)} \quad \left(\xrightarrow{k \rightarrow \infty} \hat{\theta} \right)$$

- **New** predictive estimate (this one obtained at **iteration k**):

$$\hat{\mathbf{Y}}^{(k)} = \mathbf{X} \hat{\theta}^{(k)} \quad \left(\xrightarrow{k \rightarrow \infty} \hat{\mathbf{Y}} \right)$$

Expected predictive gain of the new estimate

Stopping at $k < \infty$ can be better than the OLS ($k = \infty$)!

- Result on **MSE**:

$$\bar{k} = \operatorname{argmin}_{k \in \mathbb{N}} \left\{ \operatorname{MSE} \left(\hat{\mathbf{Y}}^{(k)} \right) \right\} \Rightarrow \operatorname{MSE} \left(\hat{\mathbf{Y}}^{(\bar{k})} \right) < \operatorname{MSE} \left(\hat{\mathbf{Y}} \right)$$

- Result on Δ (holds with high probability):

$$k^* = \operatorname{argmin}_{k \in \mathbb{N}} \left\{ \Delta \left(\hat{\mathbf{Y}}^{(k)} \right) \right\} \Rightarrow \Delta \left(\hat{\mathbf{Y}}^{(k^*)} \right) < \Delta \left(\hat{\mathbf{Y}} \right)$$

How to estimate the optimal iteration \bar{k} or k^* ?

Scope of the current study

This is a **toy** study

- Since the OLS is available in closed-form, its computational time is the best

But a **prospective** study

- Allows to **mimic** algorithm dependent estimates (numerous: closed-form is rare!)
- Allows to **understand** some fundamental factors acting in the estimate accuracy
- Allows to **glimpse** expected difficulties for estimating optimal values of k

Thus, a step before a future **generic method** for computational/accuracy trade-off. . .

Outline

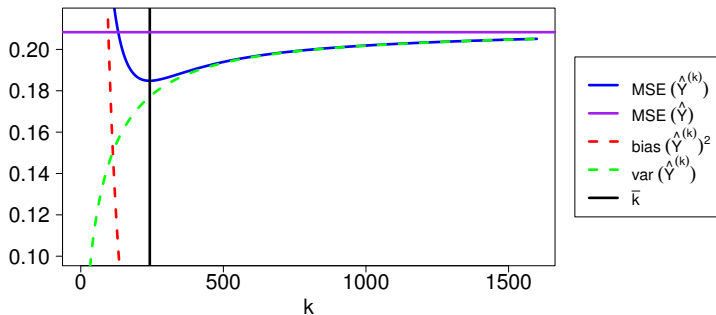
- 1 Introduction
- 2 Understanding the algorithm dynamic**
- 3 First attempts for a stopping rule
- 4 Numerical simulations

Trade-off bias variance for the MSE

$$\text{MSE}(\hat{\mathbf{Y}}^{(k)}) = \underbrace{\frac{1}{n} \left\| \mathbf{S}^k \mathbf{P}' (\mathbf{Y}^{(0)} - \mathbf{Y}^*) \right\|_{2,n}^2}_{\text{bias}(\hat{\mathbf{Y}}^{(k)})^2} + \underbrace{\frac{\sigma^2}{n} \text{Tr} \left((\mathbf{I}_n - \mathbf{S}^k)^2 \right)}_{\text{var}(\hat{\mathbf{Y}}^{(k)})}$$

where $\mathbf{K} = \frac{1}{n} \mathbf{X} \mathbf{X}' = \mathbf{P} \mathbf{\Lambda} \mathbf{P}'$; $\mathbf{S} = \mathbf{I}_n - \alpha \mathbf{\Lambda}$; $\alpha = 0.01 \in]0, \frac{1}{\hat{\lambda}_1} [$; $\hat{\lambda}_1 = \|\mathbf{K}\|_2$

For $d = 20$ $n = 30$



Something more on optimal values of k

There exists $M_1, M_2, M_3, M_4 > 0$ such as, with high probability, for large n ,

$$M_1 + M_2 \log(n) \leq k^* \leq M_3 + M_4 \log(n).$$

- Thus it suggests to perform “few” iterations for small samples sizes
- Somewhat consistent with the fact that the OLS ($k = \infty$) is a “large n ” estimate
- But even for large n values, k^* has not to be too high
- And if we perform too many iterations, we have the following variance effect:

$$\forall k \in \mathbb{N}, \quad \text{MSE} \left(\hat{\mathbf{Y}}^{(k)} \right) \geq \frac{\sigma^2}{4n} \sum_{j=1}^d \min \left\{ 1, \left(k\alpha\hat{\lambda}_j \right)^2 \right\}$$

Outline

- 1 Introduction
- 2 Understanding the algorithm dynamic
- 3 First attempts for a stopping rule**
- 4 Numerical simulations

Controlling bias/variance in Δ

Controlling Δ could be possible by (hopefully sharp) inequalities

Highlighting (squared) bias and variance in Δ : $\forall k \geq 0$

$$\Delta \left(\hat{\mathbf{Y}}^{(k)} \right) \leq \underbrace{\frac{2}{n} \left\| E \left[\hat{\mathbf{Y}}^{(k)} \right] - \mathbf{Y}^* \right\|_{2,n}^2}_{B_k^2} + \underbrace{\frac{2}{n} \left\| \hat{\mathbf{Y}}^{(k)} - E \left[\hat{\mathbf{Y}}^{(k)} \right] \right\|_{2,n}^2}_{V_k}$$

We have now to control also B_k^2 and $V_k \dots$

Controlling the squared bias B_k^2

If $\|\theta^*\|_{2,d} \leq 1$ and $\theta^{(0)} = 0, \forall k \in \mathbb{N}$

$$B_k^2 \leq 2\hat{\lambda}_1 e^{-2k\alpha\hat{\lambda}_d} := B_k^{2,\text{sup}}$$

This upper bound seems to be **sharp enough** to capture the **exponential** algorithm dynamic of the (squared) bias observed on the figures!

Controlling the variance V_k

$\exists C_1 > 0$, with probability at least $1 - e^{-y}$, $\forall k \in \{0 \dots k_{\max}\}$

$$V_k \leq \underbrace{2\mathbb{E}[V_k]}_{\text{main term}} + C_1 \frac{(y + \log(k_{\max} + 1))}{n}$$

and

$$2\mathbb{E}[V_k] \leq \frac{4\sigma^2}{n} \sum_{j=1}^d \min \left\{ 1, (k\alpha\hat{\lambda}_j)^2 \right\} := V_k^{\text{sup}}$$

- This upper bound seems to be **sharp enough** to capture the **quadratic then asymptote** algorithm dynamic of the variance observed on the figures!
- k_{\max} : **not dangerous** since it corresponds to the maximum iterations that the practitioner can perform in the real world and it is involved only through a logarithm scale

Stopping rule to estimate k^*

From previous results, we have with probability at least $1 - e^{-y}$, $\forall k \in \{0 \dots k_{\max}\}$,

$$\Delta(\hat{Y}^{(k)}) \leq B_k^{2,\text{sup}} + 2E[V_k] + C_1 \frac{(y + \log(k_{\max} + 1))}{n}.$$

From it, we propose the two following estimates for k^* :

$$\hat{k}_1 = \min \left\{ k \in \mathbb{N} : B_{k+1}^{2,\text{sup}} + 2\hat{E}[V_{k+1}] > B_k^{2,\text{sup}} + 2\hat{E}[V_k] \right\}$$

$$\hat{k}_2 = \min \left\{ k \in \mathbb{N} : B_{k+1}^{2,\text{sup}} + \hat{E}[V_{k+1}] > B_k^{2,\text{sup}} + \hat{E}[V_k] \right\}$$

where $\hat{E}[V_k] = \frac{2\hat{\sigma}^2}{n} \sum_{j=1}^d \left(1 - (1 - \alpha\hat{\lambda}_j)^k \right)^2$

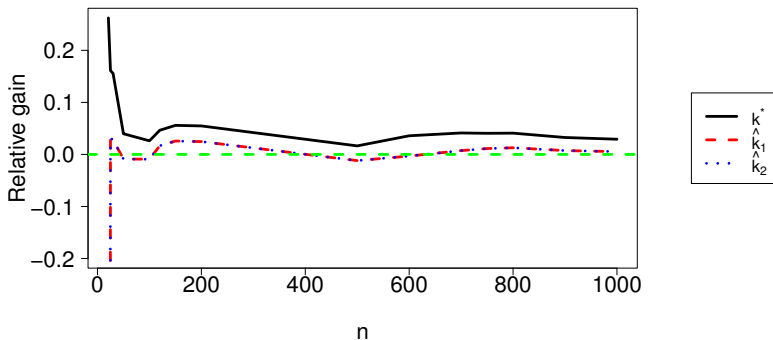
Note: not completely satisfactory since estimate $\hat{\sigma}^2$ is required. . .

Outline

- 1 Introduction
- 2 Understanding the algorithm dynamic
- 3 First attempts for a stopping rule
- 4 Numerical simulations**

Definition of the relative gain

$$\text{GainRel}(\hat{\mathbf{Y}}^{(k)}) = \frac{\text{MSE}(\hat{\mathbf{Y}}) - \text{MSE}(\hat{\mathbf{Y}}^{(k)})}{\text{MSE}(\hat{\mathbf{Y}})}.$$

Relative gain as a function of n for $d = 20$ For $d = 20$ 

- Estimates \hat{k}_1 and \hat{k}_2 with confounded behaviour
- Strong correlation with the behaviour of k^*
- Potential gain higher for small n but not too small for (quite) large n
- $n = 21$: unexpected problem for \hat{k}_1 and \hat{k}_2 ($\hat{\sigma}^2$?)
- $n \geq 22$: not completely satisfactory but not so bad for a first attempt. . .

Thank's!