mixedClust: an R package for mixed data classification, clustering and co-clustering
Margot Selosse, Julien Jacques, Christophe Biernacki

To cite this version:
Margot Selosse, Julien Jacques, Christophe Biernacki. mixedClust: an R package for mixed data classification, clustering and co-clustering. 25th Summer Session Working Group on Model-Based Clustering, Jul 2018, Ann Arbor, United States. <hal-01949171>

HAL Id: hal-01949171
https://hal.archives-ouvertes.fr/hal-01949171
Submitted on 9 Dec 2018

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.
mixedClust: an R package for mixed data classification, clustering and co-clustering

Package functionalities

The package provides model-based algorithm for clustering, co-clustering and classification with mixed-type data.

Principal functions are:
mixedClust # to perform clustering
mixedClassif # to perform classification, in a parsimonious way or not
predictions # use the result from mixedClassif for predictions

Notations

- $x$: $N$ rows and $J = J_1 + \ldots + J_d + \ldots + J_D$ columns
- $x$ composed of several matrices: $x^1, \ldots, x^D$
- $x^d$: $N \times J_d$ matrix
- $x^d$ is made of variables from one of 5 different types: Continuous, Nominal, Ordinal, Integer or Functional.
- In unsupervised methods: $G$ clusters in line, $H_1 \ldots H_D$ clusters in column

$$x = \begin{bmatrix} x^1 & \ldots & x^D \end{bmatrix}, x^d = (x^d_{ij})_{1 \leq i \leq N} \leq j \leq J_d.$$

Models (for $D = 2$)

Legend: Observed partitions - Latent partitions

- Clustering:
  $$p(x; \theta) = \sum_{z \in Z} p(z; \theta) \times p(x^1|z; \theta)p(x^2|z; \theta)$$
- Co-clustering:
  $$p(x; \theta) = \sum_{z \in Z, w \in W} p(z; \theta)p(w; \theta)p(z; \theta)p(x^1|z, w; \theta)p(x^2|z, w; \theta)$$
- Classification without parsimony:
  $$p(x; \theta) = p(z; \theta) \times p(x^1|z; \theta)p(x^2|z; \theta)$$
- Classification with parsimony (obtained by clustering the features):
  $$p(x; \theta) = \sum_{w \in W} p(z; \theta)p(w; \theta)p(z; \theta)p(x^1|z, w; \theta)p(x^2|z, w; \theta)$$

The parameters we want to estimate are:

$$\Theta = \{\Theta^d_w, \Phi^d_w, \Phi^d_{w,h}, \Phi^d_{h}, \Phi^d_{w,h}, 2 \leq d \leq D\}$$

$\Theta^d_w$: parameters of distribution of $g$th row-cluster and $h$th column-cluster of $x^d$. It will depend on the type of $x^d$.

$\Phi^d_w$: mixing proportion of $g$th row-cluster

$\Phi^d_{w,h}$: mixing proportion of $h$th column cluster for $x^d$

Inference

EM and BIC not tractable in co-clustering, due to the double missing structure. Consequently, we use:

- Stochastic EM algorithm, with a Gibbs sampler for the latent variables simulation
- ICL-BIC criterion for model selection

Results for classification on real dataset

Dataset

- Trauma-survey: 823 persons answered to 88 psychological questions about anxiety, depression, anger and possibly traumatizing life events. 307 of them were diagnosed with trauma, and 516 were declared not traumatized.
- $x^1$: Categorical data from 17 questions about traumatizing life events.
- $x^2$: Ordinal data from 71 questions about anger, depression and anxiety.
- $2/3$ of the dataset was used to train the model. The last $1/3$ was then used for prediction.

<table>
<thead>
<tr>
<th>Model</th>
<th>precision</th>
<th>recall</th>
<th>specificity</th>
</tr>
</thead>
<tbody>
<tr>
<td>not parsimonious</td>
<td>0.75</td>
<td>0.80</td>
<td>0.83</td>
</tr>
<tr>
<td>$H_1^1, H_2^1$</td>
<td>0.76</td>
<td>0.88</td>
<td>0.85</td>
</tr>
<tr>
<td>$H_1^2, H_2^2$</td>
<td>0.82</td>
<td>0.92</td>
<td>0.88</td>
</tr>
</tbody>
</table>

Table: Precision, recall and specificity for different kc.

On classification with parsimony:

- Better results are obtained on predictions when we introduce parsimony than when we don’t.
- Parsimony training result gives less parameters, which makes easier the interpretation.

Features clusters for parsimonious classification

Figure: Patients classification and features clusters. Categorical answers about life events on the left. Ordinal answers about Anger/Anger/Depression on the right.

R code

### Defining the dataset properties ###

distri = c("Multinomial", "Bos") # defining the distribution types
distriSEM = c(250) # total number of iterations
nbSEMburn = 200 # burn-in period
nbindmini = 10 # minimum number of elements in one block
init = "kmeans" # initialization type

### defining the number of clusters ###

kcr = 2 # Two classes : Traumatized/Not Traumatized
kc = c(2,5) # Introducing parsimony (put to 0 for no parsimony)

### running the classification function ###

classif = mixedClassif(data, y, train, distri, kr = kr, kc = kc, init, 
distri, nbSEM, nbindmini)

### printing predicted labels ###

prediction = predictions(classif, M.test)

References