Variable Clustering and Mixed Data.
The ClustOfVar package

Marie Chavent

En collaboration avec : Robin Genuer, Vanessa Kuentz, Amaury Labenne, Benoît Liquet, Jérôme Saracco

University of Bordeaux, France
Inria Bordeaux Sud-Ouest, CQFD Team
Irstea, UR ADBX, cestas, France
The University of Queensland, Australia

Outline

1. The package PCAmixdata

2. The package ClustOfVar

Outline

1. The package PCAmixdata
2. The package ClustOfVar

Multivariate data analysis

Standard methods (among others):

- PCA (Principal Component Analysis) for numerical data, MCA (Multiple Correspondence Analysis) for categorical data.
- MFA (Multiple Factor Analysis) or STATIS for multiple-table data but the data should be of the same nature (numerical or categorical) in a given group.

Standard R packages (among others):

- **ade4** (Dray and Dufour, 2007).
- **FactoMineR** (Lê, Josse, Husson et al., 2008).
- **ExPosition** (Beaton, Chin Fatt and Abdi, 2014).

Multivariate data analysis of mixed data type

PCA of a mixture of numerical and categorical data

- PCAMIX (Kiers, 1991)
- AFDM (Pagès, 2004).

  ↪ Function `AFDM` in the R package **FactoMineR**
- Hill & Smith (1976).

  ↪ Function `dudi.mix` in the R package **ade4**
- Others...?

The R package PCAmixdata

- GSVD (Generalized Singular Value Decomposition) implementation of the methods.
- Function `PCAmix`
  ↪ Same name but different from PCAMIX (Kiers, 1991).
  ↪ Includes PCA and MCA as special cases.
- Function `PCArot` for rotation in PCAMIX.
  ↪ paper in ADAC, 2012
- Function `MFAmix` for MFA with mixed data type within the groups of variables
  ↪ PhD of Amaury Labenne (Irstea).

A mixed data type example

The wine data set of dimension 21 × 31:

- ↪ 21 wines of Val de Loire
- ↪ 2 categorical variables (label of origin ans soil) and 29 numerical sensory descriptors.

```
library(PCAmixdata)
data(wine)
head(wine[, c(1, 2, 14:16)])

##             Label      Soil Flower Spice Plante
## 2EL        Saumur      Env1  2.320 1.840  2.000
## 1CHA       Saumur      Env1  2.440 1.739  2.000
## 1FON Bourgueuil      Env1  2.192 2.250  1.750
## 1VAU       Chinon      Env2  2.083 2.167  2.304
## 1DAM       Saumur Reference  2.231 2.148  1.762
## 2BOU Bourgueuil Reference  2.240 2.148  1.750
```

A mixed data type example

Two data sets:

$\hookrightarrow$ a numerical data matrix $\mathbf{X}_1$ of dimension 21 × 3.

$\hookrightarrow$ a categorical data matrix $\mathbf{X}_2$ of dimension 21 × 2.

```
X1 <- wine[, 14:16]
head(X1)

##      Flower Spice Plante
## 2EL   2.320 1.840  2.000
## 1CHA  2.440 1.739  2.000
## 1FON  2.192 2.250  1.750
## 1VAU  2.083 2.167  2.304
## 1DAM  2.231 2.148  1.762
## 2BOU  2.240 2.148  1.750
```

```
X2 <- wine[, 1:2]
head(X2)

##           Label      Soil
## 2EL      Saumur      Env1
## 1CHA     Saumur      Env1
## 1FON Bourgueuil      Env1
## 1VAU     Chinon      Env2
## 1DAM     Saumur Reference
## 2BOU Bourgueuil Reference
```

Data preprocessing

1. A single numerical data matrix:

```
library(FactoMineR)
head(cbind(X1, tab.disjonctif(X2)))

##       Flower Spice Plante Saumur Bourgueuil Chinon Reference Env1 Env2 Env4
## 2EL    2.320 1.840  2.000      1          0      0         0    1    0    0
## 1CHA   2.440 1.739  2.000      1          0      0         0    1    0    0
## 1FON   2.192 2.250  1.750      0          1      0         0    1    0    0
## 1VAU   2.083 2.167  2.304      0          0      1         0    0    1    0
## 1DAM   2.231 2.148  1.762      1          0      0         1    0    0    0
## 2BOU   2.240 2.148  1.750      0          1      0         1    0    0    0
```

2. The first three columns are standardized and the indicator matrix is centered.

The PCAmix function

PCA of a mixture of numerical and categorical data:

- ↪ Factor scores for rows in **F**.
- ↪ Factor scores for numerical columns in $\mathbf{A}_1$.
- ↪ Factor scores for categories in $\mathbf{A}_2$.

```
obj <- PCAmix(X.quanti = X1, X.quali = X2, ndim = 2)
```

```
F <- obj$scores
head(F)
```

```
##         dim1    dim2
## 2EL  -1.010 -1.1148
## 1CHA -1.559 -1.4747
## 1FON -1.436  1.0832
## 1VAU  1.710 -2.3895
## 1DAM -1.044  0.6605
## 2BOU -1.822  1.3077
```

```
A1 <- obj$quanti.cor
head(A1)
```

```
##            dim1    dim2
## Flower -0.7563 -0.1608
## Spice   0.5239  0.6743
## Plante  0.8083 -0.3351
```

```
A2 <- obj$categ.coord
head(A2)
```

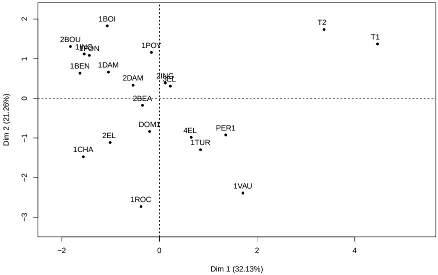
```
##                     dim1     dim2
## Label=Bourgueuil -0.7668  0.81280
## Label=Chinon      0.1222 -1.17544
## Label=Saumur      0.3738 -0.01591
## Soil=Env1        -0.4816 -0.05779
## Soil=Env2         0.5217 -1.27636
## Soil=Env4         2.4441  1.19147
```

Factor scores for wines

```
head(F)
```

```
##          dim1    dim2
## 2EL  -1.010 -1.1148
## 1CHA -1.559 -1.4747
## 1FON -1.436  1.0832
## 1VAU  1.710 -2.3895
## 1DAM -1.044  0.6605
## 2BOU -1.822  1.3077
```

```
# Component map with factor scores of the wines (rows)
plot(obj, choice = "ind")
```

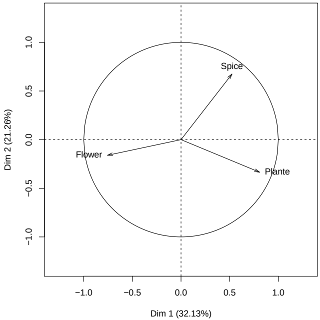


Factor scores (loadings) for numerical variables

```
head(A1)
```

```
##            dim1    dim2
## Flower -0.7563 -0.1608
## Spice   0.5239  0.6743
## Plante  0.8083 -0.3351
```

```
# Component map with factor scores of the numerical columns
plot(obj, choice = "cor")
```

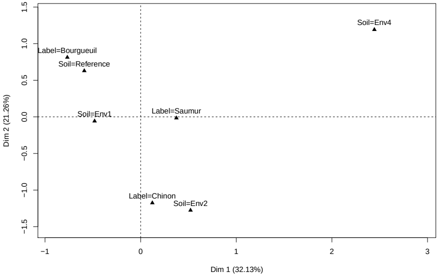


↪ The property that loadings are correlations is **TRUE**

Factor scores for the categories

```
head(A2)


##                       dim1     dim2
## Label=Bourgueuil -0.7668  0.81280
## Label=Chinon      0.1222 -1.17544
## Label=Saumur      0.3738 -0.01591
## Soil=Env1        -0.4816 -0.05779
## Soil=Env2         0.5217 -1.27636
## Soil=Env4         2.4441  1.19147
```

```
# Component map with factor scores of the categories
plot(obj, choice = "categ")
```



↪ Barycentric property is **TRUE**

Contributions of the numerical and categorical variables

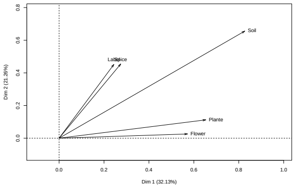
↪ Squared correlation for **numerical variables**.

↪ Correlation ratio for **categorical variables**.

```
# contributions of the variables
head(obj$sload)
```

```
##           dim1    dim2
## Flower 0.5720 0.02587
## Spice  0.2745 0.45464
## Plante 0.6533 0.11228
## Label  0.2440 0.45206
## Soil   0.8268 0.65609
```

```
plot(obj, choice = "var")
```



The PCAmix algorithm

An simple algorithm in three main steps

1. Preprocessing step.
2. GSVD (Generalized Singular Value Decomposition) step.
3. Scores processing step.

Some notations:

- Let $\mathbf{X}_1$ be a $n \times p_1$ **numerical** data matrix.
- Let $\mathbf{X}_2$ be a $n \times p_2$ **categorical** data matrix.
- Let m be the total number of categories.

The PCAmix algorithm

Preprocessing step

1. Build a **numerical data matrix** $\mathbf{Z} = (\mathbf{Z}_1|\mathbf{Z}_2)$ of dimension $n \times (p_1 + m)$ with:
   - ↪ $\mathbf{Z}_1$ the standardized version of $\mathbf{X}_1$.
   - ↪ $\mathbf{Z}_2$ the centered indicator matrix of $\mathbf{X}_2$.
2. Build the diagonal matrix $\mathbf{N}$ of the weights of the rows.
   - ↪ The n rows are weighted by $\frac{1}{n}$.
3. Build the diagonal matrix $\mathbf{M}$ of the weights of the columns.
   - ↪ The p_1 first columns are weighted by 1.
   - ↪ The m last columns are weighted by $\frac{n}{n_s}$, with n_s the number of observations with category s.

↪ The total variance is $p_1 + m - p_2$.

The PCAmix algorithm

GSVD step

The GSVD (Generalized Value Decomposition) of $\mathbf{Z}$ with the diagonal metrics of the weights $\mathbf{N}$ and $\mathbf{M}$ gives the decompositon

$$\mathbf{Z} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^t \tag{1}$$

where

- $\mathbf{\Lambda} = \text{diag}(\sqrt{\lambda_1}, \ldots, \sqrt{\lambda_r})$ is the $r \times r$ diagonal matrix of the singular values of $\mathbf{ZMZ}^t\mathbf{N}$ and $\mathbf{Z}^t\mathbf{NZM}$, and r denotes the rank of $\mathbf{Z}$;
- $\mathbf{U}$ is the $n \times r$ matrix of the first r eigenvectors of $\mathbf{ZMZ}^t\mathbf{N}$ such that $\mathbf{U}^t\mathbf{NU} = \mathbb{I}_r$;
- $\mathbf{V}$ is the $p \times r$ matrix of the first r eigenvectors of $\mathbf{Z}^t\mathbf{NZM}$ such that $\mathbf{V}^t\mathbf{MV} = \mathbb{I}_r$.

The PCAmix algorithm

Scores processing step

1. The set of factor scores for rows is computed as:

$$\mathbf{F} = \mathbf{U}\mathbf{\Lambda}.$$

2. The set of factor scores for colums is computed as:

$$\mathbf{A} = \mathbf{M}\mathbf{V}\mathbf{\Lambda}.$$

3. $\mathbf{A}$ is splitted as follows: $\mathbf{A} = \begin{pmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \end{pmatrix} \begin{matrix} \}\, p_1 \\ \}\, m \end{matrix}$ with

   $\mathbf{A}_1$: scores of the p_1 numerical variables

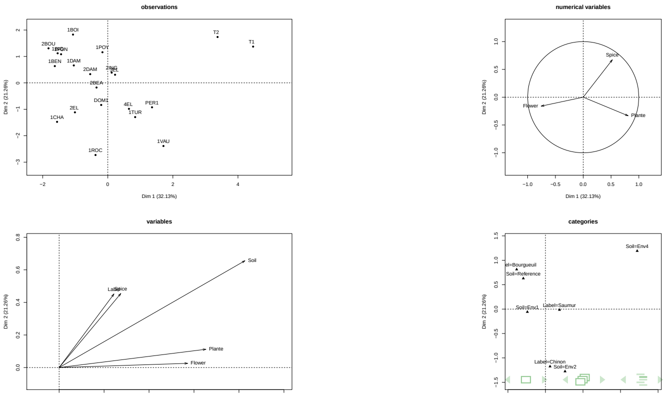
   $\mathbf{A}_2$: scores of the m categories

$\hookrightarrow$ Different from standard PCA where $\mathbf{A} = \mathbf{V}\mathbf{\Lambda}$.

Graphical output of PCAmix

Graphical output

Display the pattern of similarity of the observations and of the variables or categories as points in maps.



Properties for component maps interpretation

- Each eigenvalue λ_α is the variance of the αth column of $\mathbf{F}$.
- Each score $a_{j\alpha}$ of a numerical variable j is its correlation with the αth column of $\mathbf{F}$.
- Each score $a_{s\alpha}$ of a category s is the mean value of the scores of the observations having this category.
- the contribution $c_{j\alpha}$ of a variable j to the component α is:

$$\begin{cases} c_{j\alpha} = a_{j\alpha}^2 & \text{if variable } j \text{ is numerical,} \\ c_{j\alpha} = \sum_{s \in I_j} \frac{n}{n_s} a_{s\alpha}^2 & \text{if variable } j \text{ is categorical.} \end{cases}$$

$\hookrightarrow$ $c_{j\alpha}$ is a squared correlation if j is numerical.
$\hookrightarrow$ $c_{j\alpha}$ is a correlation ratio if j is categorical.

Numerical output of PCAmix

Numerical output

A set of new orthogonal numerical variables called principal components.

```
# The original data
head(cbind(X1, X2))


##      Flower Spice Plante      Label      Soil
## 2EL   2.320 1.840  2.000     Saumur      Env1
## 1CHA  2.440 1.739  2.000     Saumur      Env1
## 1FON  2.192 2.250  1.750 Bourgueuil      Env1
## 1VAU  2.083 2.167  2.304     Chinon      Env2
## 1DAM  2.231 2.148  1.762     Saumur Reference
## 2BOU  2.240 2.148  1.750 Bourgueuil Reference
```

```
# The two first principal components
head(F)


##        dim1    dim2
## 2EL  -1.010 -1.1148
## 1CHA -1.559 -1.4747
## 1FON -1.436  1.0832
## 1VAU  1.710 -2.3895
## 1DAM -1.044  0.6605
## 2BOU -1.822  1.3077
```

Principal component intepretation

Property of the principal components

The principal components (columns of **F**) are **non correlated** linear combination of the columns of **Z** (new synthetic variables) with:

- maximum dispersion,
- maximum link to the original variables.

↪ Maximum dispersion:

$$\begin{aligned}\lambda_\alpha &= \|\mathbf{f}_\alpha\|_{\mathbf{N}} \\ &= \mathrm{Var}(\mathbf{f}_\alpha)\end{aligned}$$

↪ Maximum link:

$$\begin{aligned}\lambda_\alpha &= \|\mathbf{a}_\alpha\|_{\mathbf{M}^{-1}} \\ &= \sum_{j=1}^{p_1} r^2(\mathbf{x}_j, \mathbf{f}_\alpha) + \sum_{j=p_1+1}^{p_2} \eta^2(\mathbf{f}_\alpha|\mathbf{x}_j)\end{aligned}$$

r^2 and η^2 are resp. squared correlation and correlation ratio.

Principal components prediction

Each principal component $\mathbf{f}_\alpha$ writes as a linear combination of the columns of $\mathbf{X} = (\mathbf{X}_1|\mathbf{G})$ where $\mathbf{X}_1$ is the numerical data matrix and $\mathbf{G}$ is the indicator matrix of the categorical matrix $\mathbf{X}_2$:

$$\mathbf{f}_\alpha = \beta_0 + \sum_{j=1}^{p_1+m} \beta_j \mathbf{x}_j$$

with:

$$\beta_0 = -\sum_{k=1}^{p_1} v_{k\alpha} \frac{\bar{\mathbf{x}}_k}{s_k} - \sum_{k=p_1+1}^{p_1+m} v_{k\alpha},$$

$$\beta_j = v_{j\alpha} \frac{1}{s_j}, \text{ for } j = 1, \ldots, p_1$$

$$\beta_j = v_{j\alpha} \frac{n}{n_j}, \text{ for } j = p_1+1, \ldots, p_1+m$$

Principal components prediction

```
# The original data
head(cbind(X1, tab.disjonctif(X2)))


##      Flower Spice Plante Saumur Bourgueuil Chinon Reference Env1 Env2 Env4
## 2EL   2.320 1.840  2.000      1          0      0         0    1    0    0
## 1CHA  2.440 1.739  2.000      1          0      0         0    1    0    0
## 1FON  2.192 2.250  1.750      0          1      0         0    1    0    0
## 1VAU  2.083 2.167  2.304      0          0      1         0    0    1    0
## 1DAM  2.231 2.148  1.762      1          0      0         1    0    0    0
## 2BOU  2.240 2.148  1.750      0          1      0         1    0    0    0
```

```
# Coefficients for the first PC
obj$coef$dim1


##                     [,1]
## const            -2.8090
## Flower           -3.2064
## Spice             1.6228
## Plante            3.1596
## Label=Bourgueuil -0.4783
## Label=Chinon      0.0762
## Label=Saumur      0.2332
## Soil=Env1        -0.3004
## Soil=Env2         0.3254
## Soil=Env4         1.5244
## Soil=Reference   -0.3676
```

```
# The first principal component
F[1:6, 1, drop = FALSE]


##        dim1
## 2EL  -1.010
## 1CHA -1.559
## 1FON -1.436
## 1VAU  1.710
## 1DAM -1.044
## 2BOU -1.822
```

Principal components prediction

```
# Scores on the learning set
test <- c(4, 17, 19, 21)
obj2 <- PCAmix(X.quanti = X1[-test, ], X.quali = X2[-test, ], ndim = 2)
head(obj2$scores)


##          dim1    dim2
## 2EL  -0.4964 -1.6273
## 1CHA -0.9921 -2.1075
## 1FON -1.3538  0.9510
## 1DAM -0.8317  0.6308
## 2BOU -1.6982  1.4204
## 1BOI -1.0393  2.0484


# Scores on the test set
predict(obj2, X.quanti = X1[test, ], X.quali = X2[test, ])


##          dim1    dim2
## 1VAU  2.0684 -2.1522
## 2BEA -0.2293 -0.1568
## 2ING  0.4489  0.3720
## T2    3.9652  1.8734
```

Outline

1. The package PCAmixdata
2. The package ClustOfVar

Clustering of mixed data type

Clustering of observations

- Lump together very similar observations:
  - ↪ Separates observations into clusters that can be scored as a single observation.
  - ↪ Data reduction.
- For numerical data:
  - ↪ Standard methods (among others): functions `kmeans` and `hsclut` in the R package **stat**.
  - ↪ Standard specific R packages (among others): **cluster**, **fastcluster**.
- For categorical or mixed data type: standard methods on principal components of PCAmix for instance.

Clustering of mixed data type

Clustering of variables

- Lumps together strongly related variables:
  - ↪ Separates variables into clusters that can be scored as a single variable.
  - ↪ Dimension reduction and redundancy removal.
- For numerical data:
  - ↪ Specific methods:
    - VARCLUS (SAS)
    - Likelihood Linkage Analysis (Lerman, 1987)
    - CLV (Vigneau and Qannari, 2003)
    - Diametrical clustering (Dhillon et al., 2003)
  - ↪ Specific R package: **ClustVarLV** (Vigneau & Chen, 2014), **ClustOfVar** (2012).
- For categorical or mixed data type: **ClustOfVar**.

The R package ClustOfVar

- Homogeneity criterion based on squared correlations and/or correlation ratios.
- Function hclustvar
  ↪ hierarchical clustering algorithm.
- Function kmeansvar
  ↪ k-means type partitionning algorithm.
- Function stability
  ↪ bootstrap approach to evaluate the stability of the partitions to determine suitable numbers of clusters.

The R package ClustOfVar

- Each cluster C_k is summarized by a numerical synthetic variable:

$$\mathbf{y}_k = \arg\max_{\mathbf{u}\in\mathbb{R}^n} \left\{ \sum_{\substack{j \in C_k \\ j \text{ numerical}}} r^2(\mathbf{x}_j, \mathbf{u}) + \sum_{\substack{j \in C_k \\ j \text{ categorical}}} \eta^2(\mathbf{x}_j, \mathbf{u}) \right\}$$

↪ $\mathbf{y}_k$ is the first principal component of PCAmix applied to the cluster C_k.

↪ Dimension reduction by replacing p variables (numerical and/or categorical) by $K < p$ numerical synthetic variables.

↪ Function predict to predict cluster scores on new observations.

The homogeneity criteria

Homogeneity of a cluster

The homogeneity of a cluster C_k of variables is:

$$H(C_k) = \sum_{\substack{j \in C_k \\ j \text{ numerical}}} r^2(\mathbf{x}_j, \mathbf{y}_k) + \sum_{\substack{j \in C_k \\ j \text{ categorical}}} \eta^2(\mathbf{x}_j, \mathbf{y}_k)$$

where $\mathbf{y}_k$ is the first principal component of PCAmix applied to the cluster.

$\hookrightarrow$ $H(C_k) = \lambda_1^k$

where λ_1^k is the first eigenvalue of PCAmix applied to C_k.

Hierarchical clustering of variables

The hierarchical algorithm

1. Starts with the partition in p clusters with one variable in each cluster.
2. Successively aggregate the two clusters with the smallest dissimilarity d:
   $d(A,B) = H(A) + H(B) - H(A \cup B) = \lambda^1_A + \lambda^1_B - \lambda^1_{A \cup B}$
3. Stop when the partition in one cluster is obtained

$\hookrightarrow$ The function hclustvar built the hierarchy.

$\hookrightarrow$ The function cutreevar cuts the hierarchy and extract a partition.

The wine data example

```
library(ClustOfVar)
data(wine)
```

27 numerical variables and 2 categorical variables.

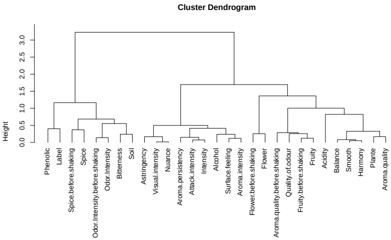
```
# 27 numerical variables
X1 <- wine[, 3:29]
head(X1[, 7:8])

##       Nuance Surface.feeling
## 2EL    4.000           3.269
## 1CHA   3.000           2.808
## 1FON   3.393           3.000
## 1VAU   2.786           2.538
## 1DAM   4.036           3.385
## 2BOU   4.259           3.407
```

```
# 2 categorical variables
X2 <- wine[, 1:2]
head(X2)

##           Label      Soil
## 2EL      Saumur      Env1
## 1CHA     Saumur      Env1
## 1FON Bourgueuil      Env1
## 1VAU     Chinon      Env2
## 1DAM     Saumur Reference
## 2BOU Bourgueuil Reference
```

Partition in 6 clusters

```
# Construction of the hierarchy
tree <- hclustvar(X.quanti = X1, X.quali = X2)
```

```
# Graphical representation
plot(tree)
```



```
# Partition in 6 clusters
part <- cutreevar(tree, 6)
# summary(part)
part$var$cluster1
```

```
##                               squared loading
## Odor.Intensity.before.shaking          0.7618
## Spice.before.shaking                   0.6160
## Odor.Intensity                         0.6663
## Spice                                  0.5358
## Bitterness                             0.6621
## Soil                                   0.7769
```

↪ Squared correlation (resp. correlation ratio) between Bitterness (resp. Soil) and the synthetic variable of cluster1 is 0.662 (resp. 0.776)

Numerical output

```
print(part)

##
## Call:
## cutreevar(obj = tree, k = 6)
##
##
##
##  name       description
##  "$var"     "list of variables in each cluster"
##  "$sim"     "similarity matrix in each cluster"
##  "$cluster" "cluster memberships"
##  "$wss"     "within-cluster sum of squares"
##  "$E"       "gain in cohesion (in %)"
##  "$size"    "size of each cluster"
##  "$scores"  "score of each cluster"
```

↪ The value $coef exists but is not indicated in print()...

Cluster scores

```
   #Synthetic variables of the clusters
   head(part$scores)


##      cluster1 cluster2 cluster3 cluster4 cluster5 cluster6
## 2EL   -1.2944  0.13892  -1.8840   0.3923   0.5870  -1.0870
## 1CHA  -2.1928 -2.64044  -2.4608   2.9837  -0.6879   0.1903
## 1FON  -0.8512 -1.55726   0.3459   1.6638   1.0280   2.4568
## 1VAU  -1.0207 -3.75874   1.1070   5.5924  -5.7755   0.2965
## 1DAM   1.0583  3.08852  -0.8204  -2.7681   2.2735  -0.8496
## 2BOU  -0.5377  0.02068  -0.2884  -2.2150   1.6799   1.3836


   #Coefficient of synthetic variable of cluster1
   part$coef$cluster1


##                                   [,1]
## const                        -23.75846
## Odor.Intensity.before.shaking   1.54125
## Spice.before.shaking            1.67396
## Odor.Intensity                  2.08113
## Spice                           1.81323
## Bitterness                      2.23125
## Env1                           -0.27887
## Env2                           -0.09136
## Env4                            1.31648
## Reference                      -0.03201
```

↪ Alternative to PCA for dimension reduction

Cluster scores prediction

```
# Cluster scores on the learning set
test <- c(4, 17, 19, 21)
tree2 <- hclustvar(X.quanti = X1[-test, ], X.quali = X2[-test, ])
part2 <- cutreevar(tree2, 6)
head(part2$scores)


##      cluster1 cluster2 cluster3 cluster4 cluster5 cluster6
## 2EL    0.5972  -0.8764  -3.0627  -1.1851  -0.2311  -1.4191
## 1CHA  -0.5541  -3.9711  -2.8356  -2.6847   4.8873  -2.7473
## 1FON  -2.6951  -2.5548  -0.4834   0.7742   3.2199  -0.2333
## 1DAM   2.0437   3.6906  -0.3632   0.2548  -2.5659   3.4588
## 2BOU  -2.3038  -0.1847   0.8875   0.4046  -1.9035   1.9645
## 1BOI  -0.6219   2.7477  -0.9022   1.1135  -0.9947   2.7016


# Cluster scores on the test set
predict(part2, X.quanti = X1[test, ], X.quali = X2[test, ])


##      cluster1 cluster2 cluster3 cluster4 cluster5 cluster6
## 1VAU -2.27020   -2.099  -0.7265   0.8095   7.6496  -8.7524
## 2BEA  0.06991    3.166   0.5602  -0.9222  -0.5749   0.9119
## 2ING -4.29945   -3.885  -2.7616  -0.8187   9.3715  -8.9123
## T2    3.11760   -2.130   3.3774   4.4639  -0.9823  -3.9453
```

K-means type clustering

Initialization step

Either:

↪ A partition in K clusters is given in input.

↪ A random partition is performed:

1. Random selection of K variables as initial centers
2. Allocation of each variable to the cluster with the closest initial center

↪ Definition of a similarity measure between two variables of any type (numerical and/or categorical)

↪ Squared canonical correlation (see paper in JSS, 2012)

↪ The function mixedvarsim.

K-means type clustering

Representation and allocation steps

Repeat:

1. Construct the synthetic variable $\mathbf{y}_k$ of each cluster C_k by applying PCAmix.
2. Assign each variable to the closest cluster.

Stops if no more changes in the partition (or a maximum number of iterations is reached).

↪ The closest cluster is that whose synthetic variable is the closest in term of squared correlation (for numerical variable) or correlation ratio (for categorical variable).

Illustration on Gene expression data (SMPGD 2013)

The context

Dimension reduction for high-dimensional supervised classification: sample size n is moderate with $n << p$.

```
# Patients treated by radical prostatectomy
load("ProstateData.RData")

# n=79 patients and p=7684 genes
cont[1:5, 1:4]


##       X1007_s_at X1255_g_at  X1294_at X200002_at
## PG13      0.8434    -0.3383 -0.148563      5.306
## PG15      0.3103    -0.2636 -0.008254      4.240
## PG37      0.5766    -0.2427 -0.097211      4.841
## PG41      0.9507    -0.3089 -0.176770      5.052
## PG46      0.1489    -0.2356 -0.042659      4.397
```

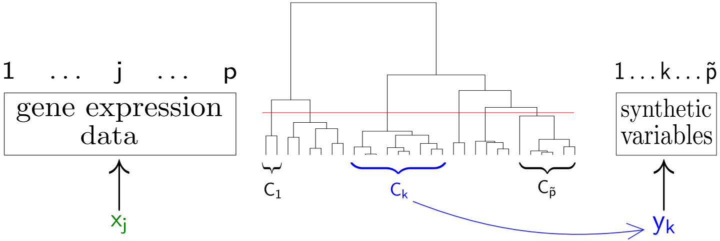
```
# Categorical dependant variable
table(type)


## type
##  0  1
## 37 42


# 1=recurrent and 0=non-recurrent
```

The methodology

An approach in two steps

1. Non supervised dimension reduction of the predictors by summarizing them in new few synthetic variables:
   - ↪ by PCA (e.g. with FactoMineR, PCAmixdata).
   - ↪ by clustering of variables with ClustOfVar to eliminates the redundancy.
2. Construction of a classifier with the synthetic variables as predictors:
   - ↪ LDA (Linear Discriminant Analysis) or random forest.
   - ↪ Selection of synthetic variables : stepwise with Wilks test for LDA or the package VSURF for random forests.

Non supervised dimension reduction with ClustOfVar



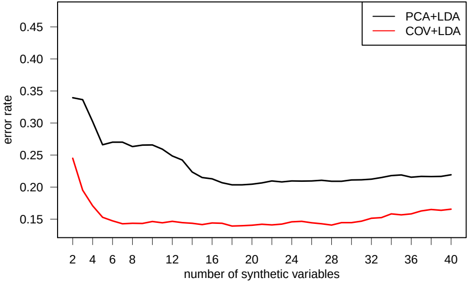
Comparison with PCA for supervised classification

```
library(PCAmixdata)
pca <- PCAmix(X.quanti = cont, ndim = 13, graph = FALSE)
Y1 <- pca$scores  #synthetic variables of PCA
```

```
library(ClustOfVar)
km <- kmeansvar(X.quanti = cont, init = 13, nstart = 1)
Y2 <- km$scores  #synthetic variables of COV
```

```
library(MASS)
# LDA on the synthetic variables of PCA with leave one out cross validation
pred1 <- lda(Y1, type, CV = TRUE)$class
sum(pred1 != type)/79  #Error rate


## [1] 0.2405


# LDA on the synthetic variables of PCA with leave one out cross validation
pred2 <- lda(Y2, type, CV = TRUE)$class
sum(pred2 != type)/79  #Error rate


## [1] 0.1646
```

Comparison with PCA for supervised classification



↪ 10-CV error rate estimation

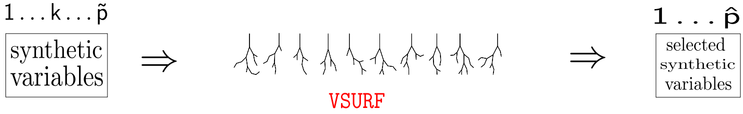
↪ Use of the functions predict of the packages PCAmixdata and ClustOfVar to construct scores on test sets.

Some R code

```
test <- sample(1:79, 20)

# Dimension reduction and models construction on the learning set
pca <- PCAmix(X.quanti = cont[-test, ], ndim = 13, graph = FALSE)
Y1 <- pca$scores  #synthetic variables of PCA
m1 <- lda(Y1, type[-test])

km <- kmeansvar(X.quanti = cont[-test, ], init = 13, nstart = 1)
Y2 <- km$scores  #synthetic variables of COV
m2 <- lda(Y2, type[-test])

# Prediction of the scores on the test set
Y1test <- predict(pca, X.quanti = cont[test, ])
Y2test <- predict(km, X.quanti = cont[test, ])

# Prediction of the dependant variable on the test set
pred1 <- predict(m1, Y1test)$class
sum(pred1 != type[test])/20  #Error rate on the test set with PCA


## [1] 0.2


pred2 <- predict(m2, Y2test)$class
sum(pred2 != type[test])/20  #Error rate on the test set with COV


## [1] 0.15
```

Synthetic variable selection with VSURF

The R package VSURF (Genuer et al. 2010)



$1 \dots k \dots \tilde{p}$

synthetic variables

⇒

VSURF

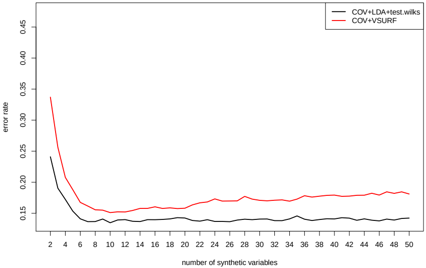
⇒

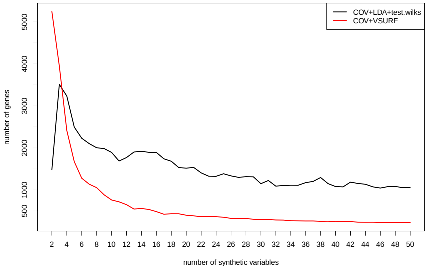
$1 \dots \hat{p}$

selected synthetic variables

- ↪ Random forests: aggregation of a collection of randomized tree-based predictors
- ↪ `VSURF`: data-driven procedure to automatically select the most important variables

Some results





COV+VSURF output for 13 clusters of variables:

↪ error rate of 16%,

↪ 4 synthetic variables selected among 13

↪ 516 genes (93.5% of genes discarded).

Some references

- Beaton, D., Chin Fatt, C. R., Abdi, H. (2014). An ExPosition of multivariate analysis with the singular value decomposition in R. *Computational Statistics & Data Analysis*, 72, 176-189.
- Chavent, M., Kuentz, V., Liquet B., Saracco, J. (2012), ClustOfVar: An R Package for the Clustering of Variables. *Journal of Statistical Software* 50, 1-16.
- Chavent, M., Kuentz, V., Saracco, J. (2012), Orthogonal Rotation in PCAMIX. *Advances in Classification and Data Analysis* 6, 131-146.
- Dray, S., Dufour, A., 2007. The ade4 package: implementing the duality diagram for ecologists. *Journal of Statistical Software* 22 (4), 120.
- Genuer, R., Poggi, J.-M. and Tuleau-Malot. C., 2010 Variable Selection using Random Forests. *Pattern Recognition Letters* 31, 2225-2236.
- Lê, S., Josse, J., Husson, F., et al. (2008). Factominer: an R package for multivariate analysis. *Journal of Statistical Software* 25 (1), 118.