Package ‘miclust’

September 22, 2021

Type Package
Title Multiple Imputation in Cluster Analysis
Version 1.2.7
Description Implementation of a framework for cluster analysis with selection of the
final number of clusters and an optional variable selection procedure. The package
is designed to integrate the results of multiple imputed datasets while accounting
for the uncertainty that the imputations introduce in the final results. In addition,
the package can also be used for a cluster analysis of the complete cases of a single
dataset. The package also includes specific methods to summarize and plot the results.
The methods are described in Basagana et al. (2013) <doi:10.1093/aje/kws289>.

Depends R (>= 4.1)
License GPL-3
Encoding UTF-8
LazyData true
Suggests knitr, rmarkdown,
Imports doBy, combinat, flexclust, graphics, irr, matrixStats, stats,
       utils
Author Jose Barrera-Gomez [aut, cre] (<https://orcid.org/0000-0002-2688-6036>),
       Xavier Basagana [aut] (<https://orcid.org/0000-0002-8457-1489>)
Maintainer Jose Barrera-Gomez <jose.barrera@isglobal.org>
NeedsCompilation no
Repository CRAN
Date/Publication 2021-09-22 12:00:02 UTC

R topics documented:

miclust-package .................................................. 2
gedata ............................................................... 2
getvariablesfrequency ......................................... 4
miclust ............................................................ 5
minhanes ......................................................... 8
Description

Cluster analysis with selection of the final number of clusters and an optional variable selection procedure. The package is designed to integrate the results of multiply imputed data sets while accounting for the uncertainty that the imputations introduce in the final results. See ‘Procedure’ below for further details on how the tool works.

Procedure

The tool consists of a two-step procedure. In the first step, the user provides the data to be analyzed. They can be a single data.frame or a list of data.frames including the raw data and the imputed data sets. In the latter case, getdata needs to be used first to get data prepared. In the second step, the miclust performs k-means clustering with selection of the final number of clusters and an optional (backward or forward) variable selection procedure. Specific summary and plot methods are provided to summarize and visualize the impact of the imputations on the results.

Authors

Jose Barrera-Gomez (maintainer, <jose.barrera@isglobal.org>) and Xavier Basagana.

References

The methodology used in the package is described in

getdata

Description

Creates a midata object.

Usage

gedata(data)
**Arguments**

- **data** a list or data.frame object. If it is a data frame, it is assumed to contain just the raw data, with or without missing data. If it is a list of data frames, it is assumed that the first element contains the raw data and the remaining ones correspond to multiple imputed data sets. Since all variables are considered in the clustering procedure, no identifier variables must be present in the data. In addition, all variables need to be treated as numeric (i.e. categorical variables must be coded with numeric values). See Details below.

**Details**

All variables in data frames in impdata are standardized by getdata, so categorical variables need to be coded with numeric values. Standardization is performed by centering all variables at the mean and then dividing by the standard deviation (or the difference between the maximum and the minimum values for binary variables). Such a standardization is applied only to the imputed data sets. The standardization of the raw data is internally applied by the miclust if needed (which is the case of analyzing just the raw data, i.e. complete cases analysis).

**Value**

An object of classes c("list", "midata") including the following items:

- **rawdata** a data frame containing the raw data.
- **impdata** if data is an object of class list, impdata is a list containing the standardized imputed data sets.

**See Also**

- miclust.

**Examples**

```r
### data minhanes:
data(minhanes)
class(minhanes)

### number of imputed datasets:
length(minhanes) - 1

### raw data with missing values:
summary(minhanes[[1]])

### first imputed data set:
minhanes[[2]]
summary(minhanes[[2]])

### data preparation for a complete case cluster analysis:
data1 <- getdata(minhanes[[1]])
class(data1)
names(data1)
```
getvariablesfrequency

Calculates the ranked selection frequency of the variables.

Description

Creates a ranked selection frequency for all the variables that have been selected at least once along the analyzed imputed data sets. getvariablesfrequency can be useful for customizing the plot of these frequencies as it is shown in Examples below.

Usage

getvariablesfrequency(x, k = NULL)

Arguments

x an object of class miclust obtained with the function miclust.

k the number of clusters. The default value is the optimal number of clusters obtained by the function miclust.

Value

A list including the following items:

percfreq vector of the selection frequencies (percentage of times) of the variables in decreasing order.

varnames names of the variables.

See Also

miclust.

Examples

### see examples in miclust.
miclust

Cluster analysis in multiple imputed data sets with optional variable selection.

Description

Performs cluster analysis in multiple imputed data sets with optional variable selection. Results can be summarized and visualized with the summary and plot methods.

Usage

miclust(
  data, 
  method = "kmeans", 
  search = c("none", "backward", "forward"), 
  ks = 2:3, 
  maxvars = NULL, 
  usedimp = NULL, 
  distance = c("manhattan", "euclidean"), 
  centpos = c("means", "medians"), 
  initcl = c("hc", "rand"), 
  verbose = TRUE, 
  seed = NULL 
)

Arguments

data          object of class midata obtained with the function getdata.
method        clustering method. Currently, only "kmeans" is accepted.
search        search algorithm for the selection variable procedure: "backward", "forward" or "none". If "none" (default), no variable selection is performed.
ks            the values of the explored number of clusters. Default is exploring 2 and 3 clusters.
maxvars       if method = "forward", the maximum number of variables to be selected.
usedimp       numeric. Which imputed data sets must be included in the cluster analysis. If NULL (default), all available imputed data sets are included. If usedimp is numeric (or a numeric vector), its values indicate which imputed data sets are included.
distance      two metrics are allowed to compute distances: "manhattan" (default) and "euclidean".
centpos       position computation of the cluster centroid. If "means" (default) the position of the centroid is computed by the mean. If "medians", by the median.
initcl        starting values for the clustering algorithm. If "rand", they are randomly selected; if "hc", they are computed via hierarchical clustering. See Details below.
verbose       a logical value indicating output status messages. Default is TRUE.
seed          a number. Seed for reproducibility of results. Default is NULL (no seed).
Details

The optimal number of clusters and the final set of variables are selected according to CritCF. CritCF is defined as

\[ CritCF = \left( \frac{2m}{2m + 1}, \frac{1}{1 + \frac{W}{B}} \right)^{\left(1 + \log_2(k + 1)ight) \left(1 + \log_2(m + 1)\right)}, \]

where \( m \) is the number of variables, \( k \) is the number of clusters, and \( W \) and \( B \) are the within- and between-cluster inertias. Higher values of CritCF are preferred (Breaban, 2011). See References below for further details about the clustering algorithm.

For computational reasons, option "rand" is suggested instead of "hc" for high dimensional data.

Value

A list with class "miclust" including the following items:

- **clustering** a list of lists containing the results of the clustering algorithm for each analyzed data set and for each analyzed number of clusters. Includes information about selected variables and the cluster vector.
- **completescasesperc** if data contains a single data frame, percentage of complete cases in data.
- **data** input data.
- **ks** the values of the explored number of clusters.
- **usedimp** indicator of the imputed data sets used.
- **kfin** optimal number of clusters.
- **critcf** if data contains a single data frame, critcf contains the optimal (maximum) value of CritCF (see Details) and the number of selected variables in the reduction procedure for each explored number of clusters. If data is a list, critcf contains the optimal value of CritCF for each imputed data set and for each explored value of the number of clusters.
- **numberofselectedvars** number of selected variables.
- **selectedkdistribution** if data is a list, frequency of selection of each analyzed number of clusters.
- **method** input method.
- **search** input search.
- **maxvars** input maxvars.
- **distance** input distance.
- **centpos** input centpos.
- **selmetriccent** an object of class kccaFamily needed by the specific summary method.
- **initcl** input initcl.

References

See Also

`getdata` for data preparation before using `miclust`.

Examples

```r
### data preparation:
minhanes1 <- getdata(data = minhanes)

### Example 1:
### Multiple imputation clustering process with backward variable selection
### using only the imputations 1 to 10 for the clustering process and exploring
### 2 vs. 3 clusters:
minhanes1clust <- miclust(data = minhanes1, search = "backward", ks = 2:3,
    usedimp = 1:10, seed = 4321)

minhanes1clust
minhanes1clust$kfin ### optimal number of clusters

### graphical summary:
plot(minhanes1clust)

### selection frequency of the variables for the optimal number of clusters:
y <- getvariablesfrequency(minhanes1clust)
y
plot(y$percfreq, type = "h", main = "", xlab = "Variable",
ylab = "Percentage of times selected", xlim = 0.5 + c(0, length(y$varnames)),
    lwd = 15, col = "blue", xaxt = "n")
axis(1, at = 1:length(y$varnames), labels = y$varnames)

### default summary for the optimal number of clusters:
summary(minhanes1clust)

### summary forcing 3 clusters:
summary(minhanes1clust, k = 3)

### Example 2:
### Same analysis but without variable selection
###
minhanes2clust <- miclust(data = minhanes1, ks = 2:3, usedimp = 1:10, seed = 4321)
minhanes2clust
plot(minhanes2clust)
summary(minhanes2clust)
```
#### Example 3:
#### Complete case clustering process with backward variable selection

```r
nhanes0 <- getdata(data = minhanes[[1]])
nhanes2clust <- miclust(data = nhanes0, search = "backward", ks = 2:3, seed = 4321)
nhanes2clust
summary(nhanes2clust)
```

### nothing to plot for a single data set analysis
# plot(nhanes2clust)

#### Example 4:
#### Complete case clustering process without variable selection

```r
nhanes3clust <- miclust(data = nhanes0, ks = 2:3, seed = 4321)
nhanes3clust
summary(nhanes3clust)
```

---

**minhanes**

*Multiple imputation for nhanes data.*

---

**Description**

A list with 101 data sets. The first data set contains nhanes data from mice package. The remaining data sets were obtained by applying the multiple imputation function mice from package mice.

**Usage**

minhanes

**Format**

A list of 101 data.frames each of them with 25 observations of the following 4 variables:

- **age** age group (1 = 20-39, 2 = 40-59, 3 = 60+). Treated as numerical.
- **bmi** body mass index (kg/m²)
plot.miclust

- **hyp** hypertensive (1 = no, 2 = yes). Treated as numerical.
- **chl** total serum cholesterol (mg/dL)

**Source**

https://CRAN.R-project.org/package=mice

**Examples**

```r
data(minhanes)
### raw data:
minhanes[[1]]
summary(minhanes[[1]])

### number of imputed data sets:
length(minhanes) - 1

### first imputed data set:
minhanes[[2]]
summary(minhanes[[2]])
```

---

plot.miclust  
*Shows a graphical representation of the results.*

**Description**

Creates a graphical representation of the results of `miclust`.

**Usage**

```r
## S3 method for class 'miclust'
plot(x, k = NULL, ...)
```

**Arguments**

- `x` object of class miclust obtained with the function `miclust`.
- `k` number of clusters. The default value is the optimal number of clusters obtained by `miclust`.
- `...` further arguments for the plot function.

**Value**

a plot to visualize the clustering results.

**See Also**

`miclust`, `summary.miclust`. 
**print.miclust**  
*Prints the results.*

**Description**

Creates a summary print of the results of `miclust`.

**Usage**

```r
## S3 method for class 'miclust'
print(x, ...)
```

**Arguments**

- `x`: object of class `miclust` obtained with the function `miclust`.
- `...`: further arguments for the print method.

**Value**

prints a description of the clustering main results.

---

**print.summary.miclust**  
*Prints the summary of results.*

**Description**

Prints the summary of the results of `summary.miclust`.

**Usage**

```r
## S3 method for class 'summary.miclust'
print(x, digits = 2, ...)
```

**Arguments**

- `x`: object of class `summary.miclust` obtained with the method `summary.miclust`.
- `digits`: digits for the print method. Default is 2.
- `...`: further arguments for the print method.

**Value**

a print of the summary of the results generated by `summary.miclust`.

**See Also**

`miclust`, `summary.miclust`. 
summary.miclust

Summarizes the results.

Description
Performs a within-cluster descriptive analysis of the variables after the clustering process performed by the function `miclust`.

Usage
```r
## S3 method for class 'miclust'
summary(object, k = NULL, quantilevars = NULL, ...)
```

Arguments
- `object`: object of class `miclust` obtained with the function `miclust`.
- `k`: number of clusters. The default value is the optimal number of clusters obtained by `miclust`.
- `quantilevars`: numeric. If a variable selection procedure was used, the cut-off percentile in order to decide the number of selected variables in the variable reduction procedure by decreasing order of presence along the imputations results. The default value is `quantilevars = 0.5`, i.e., the number of selected variables is the median number of selected variables along the imputations.
- `...`: further arguments for the plot function.

Value
An object with classes `c("list", "summary.miclust")` including the following items:

- `allocationprobabilities`: if imputations were analyzed, descriptive summary of the probability of cluster assignment.
- `classmatrix`: if imputations were analyzed, the individual probabilities of cluster assignment.
- `cluster`: if imputations were analyzed, the final individual cluster assignment.
- `clustersize`: if imputations were analyzed, size of the imputed cluster and between-imputations summary of the cluster size.
- `clustervector`: if a single data set (raw data set) has been clustered, a vector containing the individuals cluster assignments.
- `clustervectors`: if imputed data sets have been clustered, the individual cluster assignment in each imputation.
- `completecasesperc`: if a single data set (raw data set) has been clustered, the percentage of complete cases in the data set.
- `k`: number of clusters.
- `kappas`: if imputations were analyzed, the Cohen’s kappa values after comparing the cluster vector in the first imputation with the cluster vector in each of the remaining imputations.
**kappadistribution** a summary of kappas.

**m** number of imputations used in the descriptive analysis which is the total number of imputations provided.

**quantilevars** if variable selection was performed, the input value of `quantilevars`.

**search** search algorithm for the selection variable procedure.

**selectedvariables** if variable selection was performed, the selected variables obtained considering `quantilevars`.

**selectedvarspresence** if imputations were analyzed and variable selection was performed, the presence of the selected variables along imputations.

**summarybycluster** within-cluster descriptive analysis of the selected variables.

**usedimp** indicator of imputations used in the clustering procedure.

**See Also**

`miclust`, `plot.miclust`.

**Examples**

```r
### see examples in miclust.
```
Index

* datasets
  minhanes, 8

getdata. 2, 5, 7
getvariablesfrequency, 4

miclust, 2–4, 5, 9–12
miclust-package, 2
minhanes, 8

plot.miclust, 9, 12
print.miclust, 10
print.summary.miclust, 10

summary.miclust, 9, 10, 11