

Package ‘ccda’

February 19, 2015

Type Package

Title Combined Cluster and Discriminant Analysis

Version 1.1

Date 04-12-2014

Author Solt Kovacs, Jozsef Kovacs, Peter Tanos

Maintainer Solt Kovacs <ccda@caesar.elte.hu>

Description This package implements the combined cluster and discriminant analysis method for finding homogeneous groups of data with known origin as described in Kovacs et. al (2014): Classification into homogeneous groups using combined cluster and discriminant analysis (CCDA). Environmental Modelling & Software. DOI: <http://dx.doi.org/10.1016/j.envsoft.2014.01.010>

License GPL-2

Depends MASS

NeedsCompilation no

Repository CRAN

Date/Publication 2014-12-08 22:23:08

R topics documented:

ccda.main	2
percentage	3
plotccda.cluster	4
plotccda.q95	4
plotccda.results	5

Index	6
--------------	----------

ccda.main

*Combined Cluster and Discriminant Analysis***Description**

Classification into homogeneous groups using combined cluster and discriminant analysis (CCDA).

Usage

```
ccda.main(dataset, names_vector, nr, nameslist,
prior = "proportions", return.RCDP=FALSE)
```

Arguments

dataset	Contains only the dataset as a matrix (without labels).
names_vector	Contains labels (names of sample origins) for each individual observation.
nr	Number of randomly coded datasets (RCD) investigated.
nameslist	Contains the names of sample origins as a list.
prior	A specified method that can be either "proportions" (in the case of different group sizes) or "equal" (in the case of equal group sizes). If unspecified, "proportions" is used as the default.
return.RCDP	A logical value indicating whether the method should return the percentages for the randomly coded datasets as a matrix. Not returned, unless set to "TRUE".

Details

ccda.main determines the basic grouping (Step I). For this it uses hierarchical clustering with Ward's method for the averages of the measured variables. Step II, the core cycle then runs for every one of the obtained groupings. For a suggestion on the number of randomly coded datasets investigated (nr), see Appendix in Kovacs et al., 2014. It should be noted that nr has a linear influence on the amount of time needed for computing.

Step III, the evaluation of the results is left to the user based on the output of ccda.main. Based on these outputs, the function plot.ccda.result helps the decision regarding further division.

The subgroups component of the output contains the grouping with the highest corresponding difference value. The iterative further investigation of these subgroups is required in order to obtain homogeneous groups as a final result. One should stop when the highest difference value is reached when every sampling location belongs to the same group.

Value

nameslist	Returns the input nameslist.
q95	The 95 % quantiles of the ratios of correctly classified cases by LDA for the randomly coded datasets.
ratio	Ratios of correctly classified cases by LDA for each coded dataset.

difference	Ratio-q95.
sub_groups	Suggestion for subdivision according to the maximal difference value.
RCDP	Percentages for the randomly coded datasets as a matrix.

References

Jozsef Kovacs, Solt Kovacs, Norbert Magyar, Peter Tanos, Istvan Gabor Hatvani, Angela Anda (2014): Classification into homogeneous groups using combined cluster and discriminant analysis (CCDA). Environmental Modelling & Software. DOI: <http://dx.doi.org/10.1016/j.envsoft.2014.01.010>

See Also

[percentage](#), [plotccda.results](#), [plotccda.q95](#), [plotccda.cluster](#)

Examples

```
ccda.main(iris[,1:4] , iris[,5], 500, c("setosa","versicolor","virginica"),
"proportions",return.RCDP=FALSE)
```

percentage	<i>Calculation of the ratio of correctly classified cases by linear discriminant analysis</i>
------------	---

Description

Extracts the ratio of correctly classified cases from the output of lda.

Usage

```
percentage(dataset, starting_vector, prior)
```

Arguments

dataset	Contains only the dataset as a matrix (without labels).
starting_vector	A vector specifying the class for each observation.
prior	A specified method that can be either "proportions" (in the case of different group sizes) or "equal" (in the case of equal group sizes).

Value

perctg The ratio of correctly classified cases by lda for the input grouping.

See Also

[ccda.main](#)

plotccda.cluster *Plot of the basic grouping*

Description

The function `plotccda.cluster` draws the dendrogram for the basic grouping using hierarchical clustering for the averages with Ward's method (as used in `ccda.main`).

Usage

```
plotccda.cluster(x)
```

Arguments

`x` The output list of `ccda.main`.

See Also

[ccda.main](#), [plotccda.results](#), [plotccda.q95](#)

Examples

```
result<-ccda.main(iris[,1:4] , iris[,5], 500, c("setosa","versicolor",  
"virginica"),"proportions",return.RCDP=FALSE)  
plotccda.cluster(result)
```

plotccda.q95 *CCDA density drawer*

Description

The function `plotccda.q95` draws the simulated density for the randomly coded datasets.

Usage

```
plotccda.q95(x,pl="max")
```

Arguments

`x` The output list of `ccda.main` which has to include the RCDP output! (Set `return.RCDP=TRUE` while running `ccda.main`).

`pl` "max" if the grouping with the highest difference value is considered or the number of the grouping for which the plot is made.

See Also

[ccda.main](#), [plotccda.results](#), [plotccda.cluster](#)

Examples

```
result<-ccda.main(iris[,1:4] , iris[,5], 500, c("setosa","versicolor",
"virginica"), "proportions", return.RCDP=TRUE)
plotccda.q95(result)
plotccda.q95(result, pl=2)
```

plotccda.results *Plot of the results of ccda.main*

Description

Plots the summarized results of CCDA for all possible groupings based on the output of ccda.main.

Usage

```
plotccda.results(x)
```

Arguments

x The output list of ccda.main.

See Also

[ccda.main](#), [plotccda.cluster](#), [plotccda.q95](#)

Examples

```
result<-ccda.main(iris[,1:4] , iris[,5], 500, c("setosa","versicolor",
"virginica"),"proportions", return.RCDP=FALSE)
plotccda.results(result)
```

Index

`ccda.main`, [2](#), [3–5](#)

`percentage`, [3](#), [3](#)

`plotccda.cluster`, [3](#), [4](#), [5](#)

`plotccda.q95`, [3](#), [4](#), [4](#), [5](#)

`plotccda.results`, [3–5](#), [5](#)