

Package ‘SPARTAAS’

October 22, 2021

Type Package

Title Statistical Methods for Archaeology

Version 1.1.0

Maintainer Arthur Coulon <arthur-coulon@outlook.fr>

Description Statistical pattern recognition and dating using archaeological artefacts assemblages.

Package of statistical tools for archaeology.

hclustcompro(perioclust): Bellanger Lise, Coulon Arthur, Husi Philippe (2020, ISBN:978-3-030-60103-4).

mapclust: Bellanger Lise, Coulon Arthur, Husi Philippe (2021) <[doi:10.1016/j.jas.2021.105431](https://doi.org/10.1016/j.jas.2021.105431)>.

seriograph: Desachy Bruno (2004) <[doi:10.3406/pica.2004.2396](https://doi.org/10.3406/pica.2004.2396)>.

License GPL (>= 2)

Encoding UTF-8

LazyData true

Depends R (>= 3.5.0)

Imports FactoMineR,

grDevices,dplyr,tidyr,ggplot2,plotly,stringr,colorspace,crayon,
shiny,shinydashboard,shinyjs,shinyjqui,fpc,ggdendro,htmltools,
rstudioapi,htmlwidgets,shinythemes,markdown,explor,shinyWidgets,scatterD3,
sp,ks,rgdal,foreign,maptools,grid,cluster,leaflet,ape,mapview

Suggests knitr, rmarkdown

RoxygenNote 7.1.1

NeedsCompilation no

Author Lise Bellanger [aut],
Philippe Husi [aut],
Arthur Coulon [aut, cre],
Bruno Desachy [ctb],
Benjamin Martineau [ctb]

Repository CRAN

Date/Publication 2021-10-22 14:30:02 UTC

R topics documented:

| | |
|------------------------------------------|-----------|
| SPARTAAS-package | 2 |
| adjacency | 3 |
| arrondi | 4 |
| CAdist | 5 |
| datacancer | 6 |
| dataceram | 7 |
| datangkor | 8 |
| datarcheo | 9 |
| hclust | 10 |
| hclustcompro | 11 |
| hclustcompro_app | 13 |
| hclustcompro_detail_resampling | 14 |
| hclustcompro_select_alpha | 17 |
| hclustcompro_subdivide | 20 |
| mapclust | 21 |
| mapclust_app | 25 |
| mapclust_cut_tree | 26 |
| overlap | 28 |
| seriograph | 29 |
| serio_app | 32 |
| timerange | 33 |
| Index | 35 |

| | |
|------------------|-----------------|
| SPARTAAS-package | <i>SPARTAAS</i> |
|------------------|-----------------|

Description

Statistical **P**attern **R**ecognition and da**T**ing using Archeological Artefacts assemblage**S**: hclustcompro(perioclust): compromised hierarchical agglomerative clustering method.

Details

Statistical pattern recognition and dating using archaeological artefacts assemblages. Package of statistical tools for archaeology. hclustcompro(perioclust): Bellanger Lise, Coulon Arthur, Husi Philippe (2020, ISBN:978-3-030-60103-4). mapclust: Bellanger Lise, Coulon Arthur, Husi Philippe (2021) <doi:10.1016/j.jas.2021.105431>. seriograph: Desachy Bruno (2004) <doi:10.3406/pica.2004.2396>.

Author(s)

NA

Maintainer: Arthur Coulon <arthur-coulon@outlook.fr>

References

Bellanger L., Coulon A., Husi P., 2020 – Perioclust: a new Hierarchical agglomerative clustering method including temporal or spatial ordering constraints. Springer Series, Studies in Classification, Data Analysis, and Knowledge Organization. <doi: 10.1007/978-3-030-60104-1>

Bellanger L., Husi P., Laghzali Y. (2015). Spatial statistic analysis of dating using pottery: an aid to the characterization of cultural areas in West Central France. In : Traviglia A. ed., Across Space and Time, Proceedings of the 41th International Conference on Computer Applications and Quantitative Methods in Archaeology (CAA-2013), Perth (Australie), Amsterdam University Press : 276-282.

adjacency

Dissimilarity matrix base on connectivity information.

Description

From the data of a network, we build a contiguity matrix. Based on this matrix, we generate a dissimilarity matrix. The matrix contains only 0 or 1, 1 if there is no relationship and 0 if there is a relationship. The network object is a two-column data frame. The first column contains the elements of the network and the second column contains a list of all other elements related to it. The list is a character string consisting of the names of the elements separated by commas (see example).

Usage

adjacency(network)

Arguments

network data frame with 2 columns. The first one contains all the elements (nodes) and the second one a string with all the elements related to it (links).

Value

D Dissimilarity matrix base on adjacency.

Author(s)

A. COULON

L. BELLANGER

P. HUSI

Examples

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##-- or do help(data=index) for the standard data sets.  
library(SPARTAAS)  
data(datangkor)  
  
## network stratigraphic data (Network)  
network <- datangkor$stratigraphy  
  
dissimilarity <- adjacency(network)  
dissimilarity
```

| | |
|---------|---------------------------------|
| arrondi | <i>Return the rounded value</i> |
|---------|---------------------------------|

Description

Always returns the upper value when the next digit is 5.

Usage

```
arrondi(x, acc)
```

Arguments

| | |
|-----|---------------------------------|
| x | The number to round |
| acc | The accuracy (number of digits) |

Details

Rounding to a negative number of digits means rounding to a power of ten, so for example `arrondi(x, digits = -2)` rounds to the nearest hundred.

Value

| | |
|-----|-----------------------------------|
| res | value or vector of values rounded |
|-----|-----------------------------------|

Author(s)

A. COULON
L. BELLANGER
P. HUSI

Examples

```

library(SPARTAAS)

x1 <- c(15,25,35,45,55)
round(x1,-1)
arrondi(x1,-1)

x2 <- c(-15,-25,-35,-45,-55)
round(x2,-1)
arrondi(x2, -1)

x3 <- 1.125
round(x3,2)
arrondi(x3, 2)

x4 <- seq(-0.55,0.55,0.1)
data.frame(
  val = x4,
  round = round(x4,1),
  arrondi = arrondi(x4, 1),
  equal = (arrondi(x4, 1) == round(x4,1))
)

```

CAdist

Distance matrix based on correspondence analysis results

Description

Run an Correspondences analysis on a contingency table then return the matrix distance of the coordinates (you can choose the number of axes to use to build the distance matrix with the nCP parameter).

Usage

```
CAdist(df, nPC = NULL, graph = TRUE)
```

Arguments

| | |
|-------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| df | Data.frame, matrix or table with the data for the correspondence analysis |
| nPC | Number of principal components to be retained for the construction of the distance matrix. Must be between 1 and the minimum of ncol - 1 and nrow - 1. Could also be "max". |
| graph | Logical parameter for plot the Correspondences Analysis (axis1, axis2) |

Value

| | |
|---|---------------------|
| D | The distance matrix |
|---|---------------------|

Author(s)

A. COULON
L. BELLANGER
P. HUSI

Examples

```
##---- Should be DIRECTLY executable !! ----  
##-- ==> Define data, use random,  
##--or do help(data=index) for the standard data sets.  
library(SPARTAAS)  
data(datangkor)  
  
## contingency table  
cont <- datangkor$contingency  
  
distance <- CAdist(cont, nPC = "max")  
distance  
  
## run without printing the plot  
distance <- CAdist(cont, nPC = "max", graph=FALSE)
```

datacancer

Data set of cancerology.

Description

Longitude, latitude, number of thyroid cancers. The data concern two departments of France (Loire Atlantique and Vendee) between 1998 and 2012. For reasons of confidentiality they are simulated data.

Usage

```
data("datacancer")
```

Format

List of two objects with 359 observations on the following 3 variables.

\$coord (data.frame):

longitude a numeric vector: geographical coordinate

latitude a numeric vector: geographical coordinate

\$var (vector):

var a numeric vector: number of thyroid cancers (simulated)

Author(s)

M. Karakachoff (IR CHU - l'institut du Thorax INSERM UMR 1087 - CNRS UMR 6291) Nantes, France

F. Molinie (resp. Loire-Atlantique-Vendee cancer registry - [registre-des-cancers](#)) France

Examples

```
library(SPARTAAS)
data(datacancer)
str(datacancer)
head(datacancer$coord)
str(datacancer$var)
```

dataceram

Data set of archeology

Description

This important dataset come from the Collective Research Project (CRP) on medieval and modern ceramics in the Middle Loire Basin. This project is a long-term one, since it began in 1996 and has already been the subject of two books on the subject (Husi dir. 2003 and 2013) as well as an online logicist publication (Husi dir. 2022).

Usage

```
data("dataceram")
```

Format

List of three objects with 226 observations.

\$contingency (data.frame) Contingency table of the quantities of 183 types of pottery shards in the 226 sets.

\$timerange (data.frame) The first column corresponds to the identifier (sets), the second and the third to the lower and upper bound of temporal range of the observations.

\$geographic_area Vector of the geographical area of each observation.

Author(s)

Husi P. (dir.) – La céramique médiévale et moderne du bassin de la Loire moyenne, chronotypologie et transformation des aires culturelles dans la longue durée (6e —19e s.), XXe Supplément à la Revue Archéologique du Centre de la France, FERACF, Tours, 2022.

Examples

```
library(SPARTAAS)
data(dataceram)
str(dataceram)
str(dataceram$contingency)
head(dataceram$timerange)
head(dataceram$geographic_area)
```

datangkor

Data set of archeology

Description

The archaeological data come from excavations carried out at Angkor Thom (Cambodia), the capital of the Khmer Empire between the 9th and 15th centuries (Gaucher, 2004). The dataset consists of the pottery assemblages (quantities of different types of pottery shards contained in the sets - `..$contingency`) and the stratigraphy of the sets from 3 disconnected archaeological sites (`..$stratigraphy`).

Usage

```
data("datangkor")
```

Format

List of two objects with 17 observations.

\$contingency (data.frame) Contingency table of the quantities of 12 types of pottery shards in the 17 sets.

\$stratigraphy (data.frame) Saves the stratigraphic network. First column corresponds to the nodes (sets) and the second to the edges by listing the nodes connected to it.

Author(s)

Gaucher, J. (2004). Angkor Thom, une utopie réalisée ? Structuration de l'espace et modèle indien d'urbanisme dans le Cambodge ancien. *Arts Asiatiques*, Volume 59, pp. 58-86.

Examples

```
library(SPARTAAS)
data(datangkor)
str(datangkor)
str(datangkor$contingency)
str(datangkor$stratigraphy)
```

datarcheo

Data set of archeology

Description

Latitude, longitude, Absolute difference between two dating and the name of the archeological site. Data concern archaeological contexts datation in West Central France based on a large collection of mediaval pottery finds. Two original statistical models are developed to estimate context dates using pottery. The absolute difference is calculated for each context. Data are based on a collective research about medieval ceramic directed by P. Husi ("[La céramique médiévale dans la vallée de la Loire moyenne](#)") and from archeo-statistical modelling project CeraR (L. Bellanger & Ph. Husi).

Usage

```
data("datarcheo")
```

Format

List of three objects with 240 observations on the following 4 variables.

\$coord (data.frame):

longitude a numeric vector: geographical coordinate

latitude a numeric vector: geographical coordinate **\$var (vector):**

regionalized_var a numeric vector: difference between two dating model **\$label (vector):**

noms a character vector(factor): name of archeological site

Author(s)

P. Husi IR CNRS, UMR CITERES-LAT, CNRS/Tours University, France :

Examples

```
library(SPARTAAS)
data(datarcheo)
str(datarcheo)
head(datarcheo$coord)
str(datarcheo$var)
levels(datarcheo$label)
```

`hclust`*Hierarchical Clustering*

Description

Overload of [hclust](#) for dealing with two dissimilarities matrices. Hierarchical cluster analysis on a set of dissimilarities and methods for analyzing it.

Usage

```
hclust(d, method = "complete", members = NULL, d2 = NULL, alpha = NULL)
```

Arguments

| | |
|----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>d</code> | a dissimilarity structure as produced by <code>dist</code> . |
| <code>method</code> | the agglomeration method to be used. This should be (an unambiguous abbreviation of) one of "ward.D", "ward.D2", "single", "complete", "average" (= UPGMA), "mcquitty" (= WPGMA), "median" (= WPGMC) or "centroid" (= UPGMC). |
| <code>members</code> | NULL or a vector with length size of <code>d</code> . See the 'Details' section. |
| <code>d2</code> | a second dissimilarity structure as produced by <code>dist</code> . |
| <code>alpha</code> | The mixing parameter in order to generate the <code>D_alpha</code> matrix on which the classical <code>hclust</code> method is applied. Formula: $D_alpha = alpha * d + (1-alpha) * d2$. |

Details

Data fusion (alpha optimal value parameter see [hclustcompro_select_alpha](#). It is necessary to define the appropriate proportion for each data source. This is the first sensitive point of the method that the user must consider. A tool is provided to guide his decision.

Value

[hclust](#)

Author(s)

The `hclust` function is based on Fortran code contributed to STATLIB by F. Murtagh.

A. COULON

L. BELLANGER

P. HUSI

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.

## The function is currently defined as
function (d, method = "complete", members = NULL, d2 = NULL,
         alpha = NULL)
{
  if (!is.null(d2)) {
    if (!length(d) == length(d2)) {
      stop("d and d2 have not the same size.")
    }
    if (is.null(alpha)) {
      sa <- hclustcompro_select_alpha(d, d2, method = method,
                                     resampling = FALSE)
      alpha <- sa$alpha[1]
    }
    alpha <- as.numeric(alpha)
    if (!(alpha > 0 & alpha < 1)) {
      warning("Alpha must be between 0 and 1.")
      sa <- hclustcompro_select_alpha(d, d2, method = method,
                                     resampling = FALSE)
      alpha <- sa$alpha[1]
    }
    d <- dist(alpha * d + (1 - alpha) * d2)
  }
  stats::hclust(d, method, members)
}

```

hclustcompro

hclustcompro

Description

Compromised Hierarchical bottom-up clustering method. The method use two sources of informations. The merging of the two data sources is done by a parameter (alpha) which allows to weight each source. Formula: $D_{\alpha} = \alpha * D1 + (1-\alpha) * D2$

Usage

```
hclustcompro(
  D1,
  D2,
  alpha="EstimateAlphaForMe",
  k=NULL,
  title="notitle",
  method="ward.D2",

```

```
    suppl_plot=TRUE
  )
```

Arguments

| | |
|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| D1 | First dissimilarity matrix (square matrix) or distance matrix. Could be a contingency table (see CADist). A factorial correspondences analysis is carried out and the distances are used (Chi-square Metric). |
| D2 | Second dissimilarity matrix (square matrix), same size than D1, or distance matrix. |
| alpha | The mixing parameter in order to generate the D_alpha matrix. Formula: $D_alpha = alpha * D1 + (1-alpha) * D2$ |
| k | The number of cluster you want. |
| title | The title to display on the dendrogram plot. |
| method | The agglomeration method to be used. This should be (an unambiguous abbreviation of) one of "ward.D", "ward.D2", "single", "complete", "average" (= UPGMA), "mcquitty" (= WPGMA), "median" (= WPGMC) or "centroid" (= UPGMC). |
| suppl_plot | Logical define whether to display additional plots (WSS and averages sil plot). |

Details

CAH

Data fusion (alpha optimal value parameter see [hclustcompro_select_alpha](#)). It is necessary to define the appropriate proportion for each data source. This is the first sensitive point of the method that the user must consider. A tool is provided to guide his decision.

Cut dendrogram

The division into classes, and subclasses, is the second crucial point. It must be done on the basis of knowledge of the study area and some decision support tools such as the cluster silhouette or the calculation of intra cluster variability (WSS: Within Sum of Square). You can use [hclustcompro_subdivide](#) in order to sub-divide a cluster into sub-cluster.

Value

The function returns a list (class: hclustcompro_cl).

| | |
|---------|---------------------------------------------------------------------------------------------------------------------|
| D1 | First dissimilarity matrix (square matrix) |
| D2 | Second dissimilarity matrix (square matrix) |
| D_alpha | The matrix use in the CAH result of the mixing of the two matrix (D1 and D2) |
| alpha | Alpha |
| tree | An object of class hclust which describes the tree produced by the clustering process (see hclust) |
| cluster | The vector of cluster of the selected partition |
| cutree | Plot of the cut dendrogram |
| call | How you call the function |
| cont | Original contingency data (if D1 is a contingency table) |

Author(s)

The hclust function is based on Fortran code contributed to STATLIB by F. Murtagh.

A. COULON

L. BELLANGER

P. HUSI

Examples

```
library(SPARTAAS)
data(datangkor)

#network stratigraphic data (Network)
network <- datangkor$stratigraphy

#contingency table
cont <- datangkor$contingency

#obtain the dissimilarities matrices
distance <- CAdist(cont, nPC = 11)
constraint <- adjacency(network)

#You can also run hclustcompro with the dist matrix directly
hclustcompro(D1 = distance, D2 = constraint, alpha = 0.7, k = 4)
```

hclustcompro_app *Launch the shiny application.*

Description

see [hclustcompro](#), [hclustcompro_select_alpha](#), [seriograph](#). You can also check the wiki on the application.

Usage

```
hclustcompro_app()
```

Value

No return value

Author(s)

A. COULON

L. BELLANGER

P. HUSI

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.

library(SPARTAAS)
if(interactive()){
  hclustcompro_app()
}
```

hclustcompro_detail_resampling

Resampling process in detail (one curve by set of clone).

Description

Base on a re-sampling process, we generate clone and we check for which alpha the clone and the original object are separated on the dendrogram (see below). The function show each set of clone curve.

Usage

```
hclustcompro_detail_resampling(D1, D2 = NULL, acc = 2, method = "ward.D2", iter = 5)
```

Arguments

| | |
|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| D1 | First dissimilarity matrix or contingency table (square matrix). You can replace D1 by a hclustcompro object (Don't use D2 in this case). |
| D2 | Second dissimilarity matrix or network data (square matrix) same size than D1. If D1 is a hclustcompro object D2 is set to NULL. |
| acc | Number of digits after the comma for the alpha value. |
| method | The agglomeration method to be used. This should be (an unambiguous abbreviation of) one of "ward.D", "ward.D2", "single", "complete", "average" (= UPGMA), "mcquitty" (= WPGMA), "median" (= WPGMC) or "centroid" (= UPGMC). |
| iter | The number of clones checked for each observation. |

Details**Definition of the criterion:**

A criterion for choosing alpha IN [0;1] must be determined by balancing the weights between the two information sources in the final classification. To obtain alpha, we define the following criterion:

$$CorCrit_{alpha} = |Cor(dist_{cophenetic}, D1) - Cor(dist_{cophenetic}, D2)|$$

equation(1)

The CorCrit_alpha criterium in (1) represents the difference in absolute value between two cophenetic correlation (Cophenetic correlation is defined as the correlation between two distances matrices. It is calculated by considering the half distances matrices as vectors. It measures of how faithfully a dendrogram preserves the pairwise distances between the original unmodeled data points). The first correlation is associated with the comparison between D1 and ultrametric distances from the HAC with alpha fixed; while the second compares D2 and ultrametric distances from the HAC with alpha fixed. Then, in order to compromise between the information provided by D1 and D2, we decided to estimate alpha with $\hat{\alpha}$ such that:

$$\hat{\alpha} = \min CorCrit_{alpha}$$

equation(2)

Resampling strategy:

To do this, a set of "clones" is created for each observation i . A clone c of observation i is a copy of observation i for which the adjacency relationships to others have been modified. The clone has none conection exept with j . A set is generated by varying j for all observations except i . A HAC is then carried out using the combination defined in (1) with $D1(c)$ a $(n+1) \times (n+1)$ matrix where the observations i and c are identical and $D2(c)$ a $(n+1) \times (n+1)$ matrix where the clone c of i has different neighbourhood relationships from those of i . We can create a set of $n-1$ clones for each element i in n , so $n(n-1)$ clones in total.

Intuitively, by varying alpha between 0 and 1, we will be able to identify when the clone and the initial observation will be separated on the dendrogram. This moment will correspond to the value of alpha above which the weight given to information on the connection between observations contained in D2 has too much impact on the results compared to that of D1.

Let $CorCrit_{alpha}(c)$ defines the same criterion as in (1) in which D1 and D2 are replaced respectively by $D1(c)$ and $D2(c)$. The estimated alpha is the average of estimated values for each clone. For each clone (c):

$$\hat{\alpha}(c) = \min CorCrit_{alpha}(c)$$

equation(3)

$\hat{\alpha}^*$ is the average of the $\hat{\alpha}(c)$. In the same spirit as confidence intervals based on bootstrap percentiles (Efron & Tibshirani, 1993), a percentile confidence interval based on replication is also be obtained using the empirical percentiles of the distribution of $\hat{\alpha}(c)$.

$$\hat{\alpha}^* = (1/n(n-1)) * \sum \hat{\alpha}(c)$$

equation(4)

$cIN[1; n(n-1)]$.

Value

plot The interactive plot: CorCrit_alpha criterium for each resampling dataset

Author(s)

A. COULON
L. BELLANGER
P. HUSI

Examples

```
#####
#      For view the equation      #
#####

plot(
  c(.6,.6,.6,.6),
  c(.9,.5,-.3,-.7),
  xlim = c(.6,1.4),
  ylim = c(-1.1,1),
  axes = FALSE,
  main = "Equations:",
  xlab = "",
  ylab = "",
  pch = 1
)
text(.65, .9, "( 1 )")
text(.65, .5, "( 2 )")
text(.65,-.3, "( 3 )")
text(.65,-.7, "( 4 )")

text(1, .9,
  expression(CorCrit[alpha] == abs(Cor(dist[cophenetic],dist[ceramic]) - Cor(dist[cophenetic],
  dist[stratigraphic]))
))
text(1, .5, expression(hat(alpha) == min(CorCrit[alpha], alpha)))

text(1,-.3, expression(hat(alpha)^(c) == min(CorCrit[alpha]^(c), alpha)))
text(1,-.7, expression(hat(alpha)^(c) == frac(1,n(n-1)) * sum(hat(alpha)^(c),c==1,n(n-1))))

#####

##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
library(SPARTAAS)
data(datangkor)

#network stratigraphic data (Network)
network <- datangkor$stratigraphy
```

```
#contingency table
cont <- datangkor$contingency

hclustcompro_detail_resampling(D1 = CAdist(cont, nPC="max"), D2 = adjacency(network))
```

hclustcompro_select_alpha

Estimation of the optimal value(s) for the alpha parameter.

Description

The following criterion "balances" the weight of D1 and D2 in the final clustering. The alpha value is only a point estimate but the confidence interval gives a range of possible values.

Based on a resampling process, we generate clones and recalculate the criteria according to alpha (see below).

Usage

```
hclustcompro_select_alpha(
  D1,
  D2,
  acc=2,
  resampling=TRUE,
  method="ward.D2",
  iter=5,
  suppl_plot=TRUE
)
```

Arguments

| | |
|------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| D1 | First dissimilarity matrix or contingency table (square matrix) |
| D2 | Second dissimilarity matrix or network data (square matrix) same size than D1 |
| acc | Number of digits after the comma for the alpha value |
| resampling | Logical for estimate the confidence interval with resampling strategy. If you have a lot of data you can save calculation time by setting this option to FALSE |
| method | The agglomeration method to be used. This should be (an unambiguous abbreviation of) one of "ward.D", "ward.D2", "single", "complete", "average" (= UPGMA), "mcquitty" (= WPGMA), "median" (= WPGMC) or "centroid" (= UPGMC) |
| iter | The number of clones checked for each observation. (200 observations iter=1: ~30 secs, 1000 observations iter=1: ~40 mins) |
| suppl_plot | Logical define whether to display additional plots. |

Details

Definition of the criterion:

A criterion for choosing alpha IN [0;1] must be determined by balancing the weights between the two information sources in the final classification. To obtain alpha, we define the following criterion:

$$CorCrit_{alpha} = |Cor(dist_{cophenetic}, D1) - Cor(dist_{cophenetic}, D2)|$$

Equation(1)

The CorCrit_alpha criterium in (1) represents the difference in absolute value between two cophenetic correlation (Cophenetic correlation is defined as the correlation between two distances matrices. It is calculated by considering the half distances matrices as vectors. It measures of how faithfully a dendrogram preserves the pairwise distances between the original unmodeled data points). The first correlation is associated with the comparison between D1 and ultrametric distances from the HAC with alpha fixed; while the second compares D2 and ultrametric distances from the HAC with alpha fixed. Then, in order to compromise between the information provided by D1 and D2, we decided to estimate alpha with $\hat{\alpha}$ such that:

$$\hat{\alpha} = \min CorCrit_{alpha}$$

Equation(2)

Resampling strategy:

To do this, a set of "clones" is created for each observation i . A clone c of observation i is a copy of observation i for which the distances from the second source of information have been modified. The change is made by copying the distances, for the second source, from another observation j . A HAC is then carried out using the combination defined in (1) with $D1(c)$ a $(n+1) \times (n+1)$ matrix where the observations i and c are identical and $D2(c)$ a $(n+1) \times (n+1)$ matrix where the clone c of i has different distances from those of i . A set of clones is generated by varying j for all observations except i . We can create a set of $n-1$ clones for each element i in n , so $n(n-1)$ clones in total.

Intuitively, by varying alpha between 0 and 1, we will be able to identify when the clone and the initial observation will be separated on the dendrogram. This moment will correspond to the value of alpha above which the weight given to information on the connection between observations contained in D2 has too much impact on the results compared to that of D1.

Let $CorCrit_{alpha}(c)$ defines the same criterion as in (1) in which D1 and D2 are replaced respectively by $D1(c)$ and $D2(c)$. The estimated alpha is the average of estimated values for each clone. For each clone (c):

$$\hat{\alpha}(c) = \min CorCrit_{alpha}(c)$$

Equation(3)

$\hat{\alpha}^*$ is the average of the $\hat{\alpha}(c)$. In the same spirit as confidence intervals based on bootstrap percentiles (Efron & Tibshirani, 1993), a percentile confidence interval based on replication is also be obtained using the empirical percentiles of the distribution of $\hat{\alpha}(c)$.

$$\hat{\alpha}^* = (1/n(n-1)) * \sum \hat{\alpha}(c)$$

$$\text{Equation(4)}$$

$$cIN[1; n(n - 1)].$$

Warnings:

It is possible to observe an alpha value outside the confidence interval. This problem can be solved, in some cases, by increasing the number of iterations or by changing the number of axes used for the construction of the matrix D1 following the correspondence analysis. If alpha nevertheless remains outside the interval, it means that the data is noisy and the resampling procedure is affected.

Value

The function returns a list (class: selectAlpha_obj).

alpha The estimate value of the parameter alpha (min CorCrit_alpha)

alpha.plot The CorCrit for all the possible alpha

If resampling = TRUE

sd The standard deviation

conf The confidence interval of alpha.

boxplot boxplot of alpha estimation with resampling

values All the potential alpha values obtained from clones

Author(s)

A. COULON

L. BELLANGER

P. HUSI

Examples

```
#####
#      For view the Equation      #
#####
```

```
plot(
  c(.6,.6,.6,.6),
  c(.9,.5,-.3,-.7),
  xlim = c(.6,1.4),
  ylim = c(-1.1,1),
  axes = FALSE,
  main = "Equations:",
  xlab = "",
  ylab = "",
  pch = 1
)
text(.65, .9, "( 1 )")
text(.65, .5, "( 2 )")
```

```

text(.65,-.3, "( 3 )")
text(.65,-.7, "( 4 )")

text(1, .9,
  expression(CorCrit[alpha] == abs(Cor(dist[cophenetic],dist[ceramic]) - Cor(dist[cophenetic],
    dist[stratigraphic]))
  )))
text(1, .5, expression(hat(alpha) == min(CorCrit[alpha], alpha)))

text(1,-.3, expression(hat(alpha)^(c) == min(CorCrit[alpha]^(c), alpha)))
text(1,-.7, expression(hat(alpha)^"*" == frac(1,n(n-1)) * sum(hat(alpha)^(c),c==1,n(n-1))))

#####

##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
library(SPARTAAS)
data(datangkor)

#network stratigraphic data (Network)
network <- datangkor$stratigraphy

#contingency table
cont <- datangkor$contingency

dissimilarity <- CADist(cont,nPC="max",graph=FALSE)
constraint <- adjacency(network)

hclustcompro_select_alpha(D1 = dissimilarity, D2 = constraint)
hclustcompro_select_alpha(D1 = dissimilarity, D2 = constraint, acc = 3, resampling = TRUE)

```

hclustcompro_subdivide

subdivide a cluster after runing hclustcompro.

Description

Allow the user to subdivide one cluster into sub clusters. This function work only with 'hclustcompro_cl' object return by hclustcompro function.

Usage

```
hclustcompro_subdivide(hclustcompro_cl,cluster,nb_class)
```

Arguments

hclustcompro_cl
A hclustcompro_cl object

| | |
|----------|--------------------------------------------------------------------------------------|
| cluster | The number of the cluster. Numbered from left to right on the dendrogram (1, 2, ...) |
| nb_class | The number of sub-clusters you want |

Value

hclustcompro_cl
 A new hclustcompro_cl object updated see [hclustcompro](#)

Author(s)

A. COULON
 L. BELLANGER
 P. HUSI

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
library(SPARTAAS)
data(datangkor)

#network stratigraphic data (Network)
network <- datangkor$stratigraphy

#contingency table
cont <- datangkor$contingency

#obtain the dissimilarities matrices
distance <- CADist(cont, nPC = 11)
constraint <- adjacency(network)

#You can also run hclustcompro with the dist matrix directly
clustering <- hclustcompro(D1 = distance, D2 = constraint, alpha = 0.7, k = 7) #number of cluster 7
clustering <- hclustcompro_subdivide(clustering, cluster = 5, nb_class = 2)

#subdivide more than one cluster
clustering2 <- hclustcompro(D1 = distance, D2 = constraint, 0.7, k=7) #number of cluster 7
clustering2 <- hclustcompro_subdivide(clustering2, cluster = c(5,7), nb_class = c(2,2))
```

Description

Get started with MapClust

This function performs a divisive hierarchical clustering on a regionalized variable using spatial patches algorithm (Woillez and al. 2007; Woillez, Rivoirard and Petitgas 2009). It is a hierarchical top-down clustering with geographical constraint. It is possible to cut the tree by clicking on the dendrogram at the level you want. Results include description of the clusters and graphics. When cutting the dendrogram you can look at the two plots (WSSPlot and AveSilPlot) which shows the relative good quality of the partitions. The first one shows the Within Sum of Square (WSS) for each partition and you can use the Elbow approach in order to select one partition. The second plot show the average silhouette width. This index is between -1 and 1. The closer it is to 1 the more the partition is good. See the fonction help page of `silhouette{cluster}`

See [mapclust_cut_tree](#) if you want cut the dendrogram to another dlim or number of clusters without re run `mapclust()`.

Usage

```
mapclust(
  coord,
  var,
  label = "Nolabel",
  iter = 20,
  Plabel = TRUE,
  lonlat = TRUE,
  positive_var = FALSE,
  n = NULL
)
```

Arguments

| | |
|---------------------------|---------------------------------------------------------------------------------------------------|
| <code>coord</code> | The x-coordinates (longitude) and the y-coordinates (latitude) data.frame or matrix dimension 2. |
| <code>var</code> | The regionalized variable(s) of interest |
| <code>label</code> | (optional) The names of the samples or an id. Must be factor. |
| <code>iter</code> | The number of iterations. The number of different dlim you want to test. (must be higher than 10) |
| <code>Plabel</code> | Logical parameter for activate or not the print of labels on the dendrogram |
| <code>lonlat</code> | Logical parameter for activate or not the cartography in lonlat system with Leaflet (basemap) |
| <code>positive_var</code> | logical parameter defining whether your variable of interest is positive or not |
| <code>n</code> | Number of groups. If NULL you can select the number of groups by clicking on the dendrogram |

Details

Dlim is the select minimum distance from sample to patch centre: to identify patches (units are those of coordinates). The dlim is automatically initialized and the user does not need to use it. The minimal data are data frame or matrix with at least 3 columns.

Value

the function returns a list.

Plot:

| | |
|-------------------|--------------------------------------------------------|
| dendrogram | The global dendrogram (hclust object) |
| dendrogram_ggplot | The global dendrogram (ggplot2 object) |
| cuttree | The cut dendrogram |
| map | The map of the selected partition |
| AveSilPlot | The average silhouette width plot (for each partition) |
| WSSPlot | The Within Sum of Square plot (for each partition) |
| silhouette | The silhouette plot of the selected partition |

Value:

| | |
|----------------|------------------------------------------------------------------------------------------------|
| X | The x-coordinates data you used |
| Y | The y-coordinates data you used |
| var | The regionalized variable(s) data you used |
| label | The label vector you used |
| density | The estimate density based on var. Equal to var if you used a unidimensionnal density variable |
| cluster | The vector of cluster of the selected partition |
| Plabel | Logical parameter for activate or not the print of labels on the dendrogram |
| fullhist | The composition cluster for each observation |
| hist | The composition cluster without duplicates (matches to leaf of the dendrogram) |
| dlim | The vector of the different limit distances |
| cutdlim | The select dlim for the cut of the current partition |
| DiMatrix | The matrix of Weighted Euclidian distances |
| silhouetteData | The silhouette data of the selected partition |
| AveSilData | The average silhouette value for each partition |
| Moran | The Moran index for each groups for each partitions |
| lonlat | Logical parameter if your coordinates are in longitude latitude format or not. |

Author(s)

A. COULON L. BELLANGER P. HUSI

References

Bellanger L., Coulon A. and Husi P. (2021) Determination of cultural areas based on medieval pottery using an original divisive hierarchical clustering method with geographical constraint (Map-Clust), Journal of Archaeological Science, Volume 132 doi: [10.1016/j.jas.2021.105431](https://doi.org/10.1016/j.jas.2021.105431).

Bellanger L., Husi P., Laghzali Y. (2015). Spatial statistic analysis of dating using pottery: an aid to the characterization of cultural areas in West Central France. In : Traviglia A. ed., Across Space and Time, Proceedings of the 41th International Conference on Computer Applications and Quantitative Methods in Archaeology (CAA-2013), Perth (Australie), Amsterdam University Press : 276-282.

Woillez M., Poulard J.C., Rivoirard J., Petitgas P., Bez N. (2007). Indices for capturing spatial patterns and their evolution in time, with application to European hake (*Merluccius merluccius*) in the Bay of Biscay. ICES J. Mar. Sci. 64, 537-550.

Woillez M., Rivoirard J. and Petitgas P. (2009) Notes on survey-based spatial indicators for monitoring fish populations, Aquatic Living Resources, 22 :155-164.

Examples

```
#####
## loading data
library(SPARTAAS)
data(datarcheo)
data(datacancer)

#####
### Example: 1
## Function "mapclust"
# object <- mapclust( coord = ..., var = ..., label = ...)

classification <- mapclust(datarcheo$coord, datarcheo$var, datarcheo$label, n=4)

#Global dendrogram
classification$dendrogram
#Cut dendrogram
classification$cuttree
#silhouette of selected partition
classification$silhouette

#You can cut the dendrogram for another dlim
NewCut <- mapclust_cut_tree(classification, dlim=0.30)

#See evaluation using Silhouette width by running:
NewCut$silhouette
#If the plot is empty try to increase the height of the window (full screen)

#See summary of the data by running:
summary(NewCut$silhouetteData)

#####
```

```
## kmeans comparison
# prepare data (only geographical data)
datakmeans <- datarcheo$coord

#kmeans
number_cluster <- 4
cl <- kmeans(datakmeans, number_cluster)
plot(datakmeans, col = cl$cluster)
```

mapclust_app

Shiny application for ClustCutMap method

Description

This function performs a divisive hierarchical clustering on a regionalized variable using spatial patches algorithm (Woillez et al. 2007). It is a hierarchical top-down clustering with geographical constraint. It is possible to cut the tree by clicking on the dendrogram at the level you want. Results include description of the clusters and graphics. When cutting the dendrogram you can look at the two plots (WSSPlot and AveSilPlot) which shows the relative good quality of the partitions. The first one shows the Within Sum of Square (WSS) for each partition and you can use the Elbow approach in order to select one partition. The second plot show the average silhouette width. This index is between -1 and 1. The closer it is to 1 the more the partition is good. See the fonction help page of `silhouette{cluster}`

Usage

```
mapclust_app()
```

Details

Run the function without argument to open the application. You can import your own data in the Uploading tab. The calculation starts in the "ClustCutMap" tab. Select data, press run and choose the partition.

Author(s)

A. Coulon

Examples

```
#open application
#library(SPARTAAS)
#if(interactive()){
  #mapclust_app()
#}
```

| | |
|-------------------|-------------------------------------------------------------------------------------------------------------------------------|
| mapclust_cut_tree | <i>Function to cut the dendrogram for a new height (distance limit) or a new number of clusters and map the new partition</i> |
|-------------------|-------------------------------------------------------------------------------------------------------------------------------|

Description

The function returns the new map, one dendrogram with the cutting line, the silhouette width and the new vector cluster. You must run [mapclust](#) beforehand to have a `mapclust_cl` object that can be used by `mapclust_cut_tree`.

Usage

```
mapclust_cut_tree(classification, nb_grp = NA, dlim = NA)
```

Arguments

| | |
|-----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>classification</code> | The object return by <code>mapclust</code> Class: <code>mapclust_cl</code> |
| <code>nb_grp</code> | The number of groups you want on the partition. Must be integer. (don't use <code>dlim</code> in this case) |
| <code>dlim</code> | The value of <code>dlim</code> where you want to cut the dendrogram. You can enter whatever you want (numeric value) and the function select for you the nearest lower <code>dlim</code> with the same partition. (don't use <code>nb_grp</code> in this case) |

Value

the function returns a list.

Plot:

| | |
|--------------------------------|--------------------------------------------------------|
| <code>dendrogram</code> | The global dendrogram (hclust object) |
| <code>dendrogram_ggplot</code> | The global dendrogram (ggplot2 object) |
| <code>cuttree</code> | The cut dendrogram |
| <code>map</code> | The map of the selected partition |
| <code>AveSilPlot</code> | The average silhouette width plot (for each partition) |
| <code>WSSPlot</code> | The Within Sum of Square plot (for each partition) |
| <code>silhouette</code> | The silhouette plot of the selected partition |

Value:

| | |
|--------------------|-------------------------------------------|
| <code>X</code> | The x-coordinates data you used |
| <code>Y</code> | The y-coordinates data you used |
| <code>var</code> | The regionalized variable.s data you used |
| <code>label</code> | The label vector you used |

| | |
|----------------|------------------------------------------------------------------------------------------------|
| density | The estimate density based on var. Equal to var if you used a unidimensionnal density variable |
| cluster | The vector of cluster of the selected partition |
| Plabel | Logical parameter for activate or not the print of labels on the dendrogram |
| fullhist | The composition cluster for each observation |
| hist | The composition cluster without duplicates (matches to leaf of the dendrogram) |
| dlim | The vector of the different limit distances |
| cutdlim | The select dlim for the cut of the current partition |
| DiMatrix | The matrix of Weighted Euclidian distances |
| silhouetteData | The silhouette data of the selected partition |
| AveSilData | The average silhouette value for each partition |
| Moran | The Moran index for each groups for each partitions |
| lonlat | Logical parameter if your coordinates are in longitude latitude format or not. |

Author(s)

A. COULON L. BELLANGER P. HUSI

Examples

```
## loading data
library(SPARTAAS)
data(datarcheo)

##First you need to run the mapclust function.
#This function allow you to obtain one partition
# object <- mapclust( coord = ..., var = ..., label = ...)
OldClassif <- mapclust(datarcheo$coord, datarcheo$var, datarcheo$label, n = 4)

##In order to cut the dendrogram for another dlim
NewCut <- mapclust_cut_tree(classification = OldClassif, dlim = 0.37)
##In order to cut the dendrogram for another number of cluster
NewCut2 <- mapclust_cut_tree(classification = OldClassif, nb_grp = 4)

#See evaluation using Silhouette width by running:
NewCut$silhouette
#If the plot is empty try to increase the height of the window (full screen).

#See summary of the data by running:
summary(NewCut$silhouetteData)
```

| | |
|---------|-------------------------------|
| overlap | <i>Temporal overlap index</i> |
|---------|-------------------------------|

Description

The overlap index is the ratio between internal overlap and total overlap over time. We define the total overlap limit as follows: the minimum of the lower limits of the pair of individuals and the maximum of the upper limits. We define the limit of the internal overlap as follows: the maximum of the lower limits and the minimum of the upper limits.

Usage

```
overlap(temporal)
```

Arguments

| | |
|----------|-----------------------------------------------------------------------------------------------|
| temporal | A data frame with tree columns: the name of the element, the lower limit and the upper limit. |
|----------|-----------------------------------------------------------------------------------------------|

Details

The lower and upper limits must be number.

The dissimilarity between time ranges is calculated from the ratio of the Overlap of the time ranges (Distance in the case where the two ranges are disjoint) by the cumulative extent of the two.

As the ratio is bounded between -1 and 1, we add 1 to make it positive and normalise it so that it is between 0 and 1.

This overlap index must then be transformed into a dissimilarity index between sets. We do 1 - ratio to do this. It is equal to 0 if the two time ranges are identical and 1 when they are infinitely distant.

Value

| | |
|---|-----------------------------------------------------|
| D | The dissimilarity matrix base on the overlap index. |
|---|-----------------------------------------------------|

Author(s)

A. COULON
L. BELLANGER
P. HUSI

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
library(SPARTAAS)
data(dataceram)
dist <- overlap(dataceram$timerange)
```

seriograph *Plot seriograph (B. DESACHY).*

Description

Visualization of contingency data over time. **Rows** must be individuals (archaeological site,...) and **columns** must be categories (type,...).

Usage

```
seriograph(cont, order, insert, show, permute)
```

Arguments

| | |
|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| cont | Contingency table or hclustcompro object. Note: Your contingency table must have the rows sorted in chronological order. (the order parameter allows you to change the order of the rows if necessary) |
| order | Vector for change the order of the rows (use row's names or cluster names if cont is a hclustcompro object, as a character vector). The oldest one (at the bottom) must be at the end of the vector. Missing names will not be plotted. You can remove row by simply remove the name in the vector. |
| show | The element to plot. This should be (an unambiguous abbreviation of) one of "both", "EPPM" or "frequency". |
| permute | Logical for permute columns in order to show seriation. |
| insert | Vector with the position after where you want insert one or more Hiatus. Could be a list with two vector: position and label to print instead of hiatus. (see last examples) |

Details

Seriograph

We have chosen the serigraph (DESACHY 2004). This tool makes it possible to highlight the evolution of ceramics over time as well as to understand the commercial relations thanks to the imported ceramics. The percentages of each category of ceramics per set are displayed. The percentages are calculated independently for each set (row). The display of the percentages allows comparison of the different sets but does not provide information on the differences in numbers. To fill this gap, the proportion of the numbers in each class is displayed on the seriograph (weight column).

We can generalized this representation for other contingency data or with [hclustcompro](#) object.

The visualization of positive deviations from the average percentage allows us to observe a series that results from changes in techniques and materials dedicated to ceramic manufacturing over time.

In order to facilitate the exploitation of the data tables, we propose here a computerised graphic processing tool (EPPM serigraph - for Ecart Positif aux Pourcentages Moyens - positive deviation from the average percentage), which does not require specialised statistical skills and is adapted to

the case of stratified sites, where the study of the evolution of artefacts can be based on the relative chronology provided by the excavation.

The treatment consists firstly of transforming this table of counts into a table of percentages, the total number in each set (each row) being reduced to 100; these are the proportions, or frequencies, of the types in the sets are thus compared.

The display of positive deviations from the mean percentages (EPPM) shows in black on a lighter background the percentage shares that are higher than the mean percentage of the variable, so as to highlight the most significant part of the values in the table. This display is simply adapted to the seriograph: when a percentage is greater than the average percentage of the type, the excess share (called here EPPM: positive deviation from the average percentage) is shown in black, centred around the axis of the type, on the grey background of the percentage bar.

The table is then transformed into a graphic matrix where these percentages are expressed, for each type, by horizontal bars centred on the column axis. When the rows are ordered chronologically, the silhouette formed by the superposition of these frequency bars makes it possible to visualise the evolution over time of the type concerned.

The display of the percentages allows comparison of the different sets but does not provide information on the differences in numbers. To fill this gap, the proportion of the numbers in each class is displayed on the seriograph (weight column).

The processing technique applies to sets whose chronological order is not known; the lines of the graph are to be reorganised so as to obtain boat-shaped silhouettes following the hypothesis of a chronological evolution corresponding to the seriation model.

Positive deviation from the average percentage (EPPM in French)

The average percentage is calculated for each ceramic category (columns) on the total number of accounts (all classes combined). From the average percentage we recover for each category and for each rows the difference between the percentage of the category in the class with the average percentage. The EPPM corresponds to the notion of independence deviation (between rows and columns, between categories and time classes) in a chi-square test approach. Although this approach is fundamental in statistical analysis, independence deviations are here purely indicative and are not associated with a p_value that could determine the significance of deviations.

Weight

Weight is the number of observations divided by the total number of observations. It indicates for each row the percentage of the data set used to calculate the frequencies of the elements (row).

Permutation

order argument:

The rows of the contingency table are initially in the order of appearance (from top to bottom). It must be possible to re-order the classes in a temporal way (You can also order as you want your contingency table).

permute argument:

In addition, it is possible to swap ceramic categories (contingency table columns) in order to highlight a serialization phenomenon. Matrix permutation uses an algorithm called "reciprocal averages". Each line is assigned a rank ranging from 1 to n the number of lines. A barycentre is calculated for each column by weighting according to the row rank. Finally, the columns are reorganized by sorting them by their barycentre.

Insert

It's possible to insert a row in the seriograph in order to represent a archeological hiatus or other temporal discontinuities.

Value

The function returns a list (class: seriograph).

| | |
|-------------|---------------------------------------------------------------------------------------|
| seriograph | The seriograph plot |
| dendrogram | If cont is a hclustcompro object return the dendrogram with the period order as label |
| contingency | Data frame of the contingencies data group by cluster |
| frequency | Data frame of the frequencies data group by cluster |
| ecart | Data frame of the gap data group by cluster |

Author(s)

B. DESACHY
 A. COULON
 L. BELLANGER
 P. HUSI

References

Desachy B. (2004). Le sériographe EPPM : un outil informatisé de sériation graphique pour tableaux de comptages. In: Revue archéologique de Picardie, n°3-4, 2004. Céramiques domestiques et terres cuites architecturales. Actes des journées d'étude d'Amiens (2001-2002-2003) pp. 39-56 doi: [10.3406/pica.2004.2396](https://doi.org/10.3406/pica.2004.2396)

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
library(SPARTAAS)
data(datangkor)

## network stratigraphic data (Network)
network <- datangkor$stratigraphy

## contingency table
cont <- datangkor$contingency

## default
seriograph(cont)

seriograph(cont,show = "EPPM")
```

```
seriograph(cont,show = "frequency")

## Don't allow permutation of columns
seriograph(cont, permute = FALSE)

## insert Hiatus (position, 1 -> after first row from bottom: oldest)
seriograph(cont,insert = 2)
seriograph(cont,insert = c(2,3))

## insert costum label element
insert <- list(
  position = c(2,3),
  label = c("Hiatus.100years","Missing data")
)
seriograph(cont,insert = insert)

## change order with cluster name (letters on dendrogram) to sort them in a chronological order
seriograph(cont,order=c("AI03","AI09","AI01","AI02","AI04","APQR01","A005",
"A003","AI05","A001","APQR02","AI07","AI08","A002","AI06","A004","APQR03"))
## by omitting the row names, you delete the corresponding rows
seriograph(cont,order=c("AI02","AI08","APQR03","AI09"))
```

serio_app

Launch the shiny application.

Description

see [seriograph](#). You can also check the wiki on the application.

Usage

```
serio_app()
```

Value

No return value

Author(s)

A. COULON

L. BELLANGER

P. HUSI

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.

library(SPARTAAS)
if(interactive()){
  serio_app()
}
```

timerange

Plot the timerange of observations sorted by cluster.

Description

Vizualisation of cluster timerange.

Usage

```
timerange(
  data,
  cluster = NULL,
  add = NULL,
  density = NULL,
  color = NULL,
  reorder = FALSE
)
```

Arguments

| | |
|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| data | data.frame (id, inf, sup) for each observation |
| cluster | vector number cluster of observations |
| add | data.frame of information to display on hover. |
| density | vector of the density for each observation. |
| color | vector of the color. if you want the colors to correspond to something other than clusters. Character vector of the same length as the number of observations |
| reorder | Logical for reorder the color. If TRUE, the first color is for the leftmost cluster on the graph. If FALSE, the first color is for cluster number 1, wherever it is.. |

Value

The function returns a list.

| | |
|------|---------------------|
| plot | The timerange plot. |
|------|---------------------|

Author(s)

A. COULON
 B. MARTINEAU
 L. BELLANGER
 P. HUSI

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
library(SPARTAAS)
data <- data.frame(
  id = c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20),
  Lower_bound = c(400,401,401,350,500,460,482,432,399,
    489,750,740,704,700,758,789,802,755,750,820),
  Upper_bound = c(550,689,755,700,780,700,700,699,650,
    850,1100,1100,1010,889,999,999,1050,1002,1000,1100)
)

cluster = c(1,1,1,1,1,1,1,1,1,1,2,2,2,2,2,2,2,2,2,2)

add <- data.frame(
  Site = c("Angers", "Angers", "Angers", "Angers", "Angers",
    "Angers", "Angers", "Angers", "Angers", "Angers",
    "Blois", "Blois", "Blois", "Blois", "Blois",
    "Blois", "Blois", "Blois", "Blois", "Blois")
)

timerange(data, cluster, add)

## with sub group (cluster 1 is sub divided in 2: 1.1 and 1.2)
cluster_with_sub = c(1.1,1.1,1.1,1.1,1.1,1.1,1.2,1.2,1.2,1.2,1.2,2,2,2,2,2,2,2,2,2,2)

timerange(data, cluster_with_sub, add)

## with density
density <- c(32,34,35,19,9,25,19,29,28,18,10,13,9,10,9,6,3,7,7,1)
timerange(data=data, cluster=cluster, density=density)
```

Index

adjacency, 3
arrondi, 4

CAdist, 5, 12

datacancer, 6
dataceram, 7
datangkor, 8
datarcheo, 9

hclust, 10, 10, 12
hclustcompro, 11, 13, 21, 29
hclustcompro_app, 13
hclustcompro_detail_resampling, 14
hclustcompro_select_alpha, 10, 12, 13, 17
hclustcompro_subdivide, 12, 20

mapclust, 21, 26
mapclust_app, 25
mapclust_cut_tree, 22, 26

overlap, 28

perioclust (hclustcompro), 11

serio_app, 32
seriograph, 13, 29, 32
SPARTAAS (SPARTAAS-package), 2
SPARTAAS-package, 2

timerange, 33