Package 'LinkHD'

November 12, 2026

Type Package

Title LinkHD: a versatile framework to explore and integrate heterogeneous data

Version 1.26.0

Author Laura M. Zingaretti [aut, cre]

Maintainer ``Laura M Zingaretti" <m.lau.zingaretti@gmail.com>

Imports scales, cluster, graphics, ggpubr, gridExtra, vegan, rio, MultiAssayExperiment, emmeans, reshape2, data.table

Depends R(>= 3.6.0), methods, ggplot2, stats

VignetteBuilder knitr

Suggests MASS (>= 7.3.0), knitr, rmarkdown, BiocStyle

biocViews Classification,MultipleComparison,Regression,Software

output BiocStyle::html_document

vignette > %\VignetteIndexEntry{LinkHD:Multiple Heterogeneous (communities) Data Integration } %\VignetteEngine{knitr::rmarkdown} \usepackage[utf8]{inputenc}

Description Here we present Link-HD, an approach to integrate heterogeneous datasets, as a generalization of STATIS-ACT ("Structuration des Tableaux A Trois Indices de la Statistique–Analyse Conjointe de Tableaux"), a family of methods to join and compare information from multiple subspaces. However, STATIS-ACT has some drawbacks since it only allows continuous data and it is unable to establish relationships between samples and features. In order to tackle these constraints, we incorporate multiple distance options and a linear regression based Biplot model in order to stablish relationships between observations and variable and perform variable selection.

License GPL-3

Encoding UTF-8

Collate 'Read_Data.R' 'DataProcessing.R' 'DistStatis-Class.R'
'LinkData.R' 'CompromisePlot.R' 'VarSelection.R' 'GlobalPlot.R'
'CorrelationPlot.R' 'VarSelection-Class.R' 'Auxiliares.R'
'ComputeDistance.R' 'dAB.R' 'OTU2Taxa.R'

LazyData true

NeedsCompilation no

Roxygen list(wrap=FALSE)

RoxygenNote 6.1.1

git_url https://git.bioconductor.org/packages/LinkHD

git_branch RELEASE_3_23

git_last_commit 59c6970

git_last_commit_date 2026-04-28

Repository Bioconductor 3.23

Date/Publication 2026-11-11

Contents

CompromisePlot . . . 3
compromise_coords . . . 4
Compromise_matrix . . . 5
correl . . . 5
CorrelationPlot . . . 6
dAB . . . 7
DataProcessing . . . 9
DistStatis-class . . . 10
Euclid_Im . . . 11
GlobalPlot . . . 11
Inertia_comp . . . 12
Inertia_RV . . . 13
LinkData . . . 14
OTU2Taxa . . . 16
Read_Data . . . 17
RQO . . . 18
Ruminotypes . . . 19
sign_values . . . 20
Taraoceans . . . 21
Trajectories . . . 22
Variables . . . 23
VarSelection . . . 24
VarSelection-class . . . 25
VarTable . . . 26
Var_coordinates . . . 27

Index **29**

CompromisePlot *Compromise-Plot*

Description

Plot a CompromisePlot of a DiStatis object

Usage

```
## S4 method for signature 'DistStatis'
CompromisePlot(x,x_lab=NULL, y_lab=NULL,
Name=NULL, pchPoints=2, colObs=NULL,...)
```

Arguments

x	DistStatis class object.
x_lab	a character indicating x_label. Default is x.
y_lab	a character indicating y_label. Default is y.
Name	a character indicating plot title.
pchPoints	pch for points in scatter plot.
colObs	is a character indicating the color for the observations. By Default is the QR (indicating the Quality of Representation of observations)
...	additional parameters from ggplot2 library

Value

plotted CompromisePlot/s of the component/s of the given DistStatis object.

Author(s)

Laura M. Zingaretti

Examples

```
{

data(Taraoceans)
pro.phylo <- Taraoceans$taxonomy[ ,'Phylum']
TaraOc<-list(Taraoceans$phychem,as.data.frame(Taraoceans$pro.phylo),
as.data.frame(Taraoceans$pro.NOGs))
TaraOc_1<-scale(TaraOc[[1]])
Normalization<-lapply(list(TaraOc[[2]],TaraOc[[3]]),
function(x){DataProcessing(x,Method='Compositional')})
colnames(Normalization[[1]])=pro.phylo
colnames(Normalization[[2]])=Taraoceans$GO
TaraOc<-list(TaraOc_1,Normalization[[1]],Normalization[[2]])
names(TaraOc)<-c('phychem','pro_phylo','pro_NOGs')
```

```
TaraOc<-lapply(TaraOc,as.data.frame)
Output<-LinkData(TaraOc,Scale =FALSE,Distance = c('ScalarProduct','Euclidean','Euclidean'))
CompromisePlot(Output) +
theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
panel.background = element_blank(), axis.line = element_line(colour = 'black'))


}
```

---

compromise_coords *compromise_coords*

Description

Accessor to compromise coordinates from LinkData output.

Usage

```
## S4 method for signature 'DistStatis'
compromise_coords(x)
```

Arguments

x	an object from DistSatis class.

Value

compromise_coords coordinates of observations in the compromise configuration from LinkData function

Examples

```
{
data(Taraoceans)
pro.phylo <- Taraoceans$taxonomy[ ,'Phylum']
TaraOc<-list(Taraoceans$phychem,as.data.frame(Taraoceans$pro.phylo)
,as.data.frame(Taraoceans$pro.NOGs))
TaraOc_1<-scale(TaraOc[[1]])
Normalization<-lapply(list(TaraOc[[2]],TaraOc[[3]]),
function(x){DataProcessing(x,Method='Compositional')})
colnames(Normalization[[1]])=pro.phylo
colnames(Normalization[[2]])=Taraoceans$GO
TaraOc<-list(TaraOc_1,Normalization[[1]],Normalization[[2]])
names(TaraOc)<-c('phychem','pro_phylo','pro_NOGs')
TaraOc<-lapply(TaraOc,as.data.frame)
Output<-LinkData(TaraOc,Scale =FALSE,Distance = c('ScalarProduct','Euclidean','Euclidean'))
compromise_coords(Output)
}
```

Compromise_matrix *Compromise_matrix*

Description

Accessor to Compromise Matrix from LinkData output.

Usage

```
## S4 method for signature 'DistStatis'
Compromise_matrix(x)
```

Arguments

x	an object from DistSatis class.

Value

Compromise_matrix: Compromise matrix from LinkData object

Examples

```
{
data(Taraoceans)
pro.phylo <- Taraoceans$taxonomy[ ,'Phylum']
TaraOc<-list(Taraoceans$phychem,as.data.frame(Taraoceans$pro.phylo)
,as.data.frame(Taraoceans$pro.NOGs))
TaraOc_1<-scale(TaraOc[[1]])
Normalization<-lapply(list(TaraOc[[2]],TaraOc[[3]]),
function(x){DataProcessing(x,Method='Compositional')})
colnames(Normalization[[1]])=pro.phylo
colnames(Normalization[[2]])=Taraoceans$GO
TaraOc<-list(TaraOc_1,Normalization[[1]],Normalization[[2]])
names(TaraOc)<-c('phychem','pro_phylo','pro_NOGs')
TaraOc<-lapply(TaraOc,as.data.frame)
Output<-LinkData(TaraOc,Scale =FALSE,Distance = c('ScalarProduct','Euclidean','Euclidean'))
Compromise_matrix(Output)
}
```

correl *correl*

Description

Accessor to RV (Vectorial correlation coefficient) from LinkData output.

Usage

```
## S4 method for signature 'DistStatis'
correl(x)
```

Arguments

x	an object from DistSatis class.

Value

RV correlation coefficient for each input table to LinkData function

Examples

```
{
data(Taraoceans)
pro.phylo <- Taraoceans$taxonomy[ ,'Phylum']
TaraOc<-list(Taraoceans$phychem,as.data.frame(Taraoceans$pro.phylo)
,as.data.frame(Taraoceans$pro.NOGs))
TaraOc_1<-scale(TaraOc[[1]])
Normalization<-lapply(list(TaraOc[[2]],TaraOc[[3]]),
function(x){DataProcessing(x,Method='Compositional')})
colnames(Normalization[[1]])=pro.phylo
colnames(Normalization[[2]])=Taraoceans$GO
TaraOc<-list(TaraOc_1,Normalization[[1]],Normalization[[2]])
names(TaraOc)<-c('phychem','pro_phylo','pro_NOGs')
TaraOc<-lapply(TaraOc,as.data.frame)
Output<-LinkData(TaraOc,Scale =FALSE,Distance = c('ScalarProduct','Euclidean','Euclidean'))
correl(Output)
}
```

CorrelationPlot *Correlation-Plot*

Description

Plot a CorrelationPlot of a DistStatis object

Usage

```
## S4 method for signature 'DistStatis'
CorrelationPlot(x,...)
```

Arguments

x	an object from DistSatis class.
...	additional parameters from ggplot2 library

Value

correlation plot between tables from a DistStatis object.

Author(s)

Laura M. Zingaretti

Examples

```
{

data(Taraoceans)
pro.phylo <- Taraoceans$taxonomy[ ,'Phylum']
TaraOc<-list(Taraoceans$phychem,as.data.frame(Taraoceans$pro.phylo),
as.data.frame(Taraoceans$pro.NOGs))
TaraOc_1<-scale(TaraOc[[1]])
Normalization<-lapply(list(TaraOc[[2]],TaraOc[[3]]),
function(x){DataProcessing(x,Method='Compositional')})
colnames(Normalization[[1]])=pro.phylo
colnames(Normalization[[2]])=Taraoceans$GO
TaraOc<-list(TaraOc_1,Normalization[[1]],Normalization[[2]])
names(TaraOc)<-c('phychem','pro_phylo','pro_NOGs')
TaraOc<-lapply(TaraOc,as.data.frame)
Output<-LinkData(TaraOc,Scale =FALSE,
Distance = c('ScalarProduct','Euclidean','Euclidean'))
CorrelationPlot(Output) +
theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
panel.background = element_blank(),
axis.line = element_line(colour = 'black'))


}
```

---

dAB *dAB*

Description

Function to estimate differential abundance (if nCluster in LinkData function is at least 2). The function uses a non parametric kruskal-wallis test follow up by corrected p-values. The function is robust since it doesn't assume normality on data distribution. This function calculates the differential abundance (at OTU level) betweeen all the communities data It is only used when CLusters (enterotypes-like) is activated in LinkData function. The function takes into account the compositional nature of the OTUs dataset. The differential expression is an alternative way to perform variable selection

Usage

```
dAB(x, Data, adjust.methods = "BH", threshold = 0.05)
```

Arguments

x	is an object of DistStatis Class.
Data	should be the same imput list than in LinkData object. If you integrated microbial communities and other types of data, please be careful: choose only the microbial communities as input to dab object!!!!
adjust.methods	character, correction method. Choose one between: c('holm', 'hochberg', 'hommel', 'bonferroni', 'BH', 'BY', 'fdr', 'none').
threshold	fixed pre-defined threshold value, which is referred to as the level of significance.

Value

Diferentialb: a list with selected OTUs and their p-values.

Author(s)

Laura M Zingatetti

References

1.
2. Kruskal, W. H., & Wallis, W. A. (1952). Use of ranks in one-criterion variance analysis. Journal of the American statistical Association, 47(260), 583-621.
3. Benjamini, Y., and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. Journal of the Royal Statistical Society Series B 57, 289–300.
4. Wright, S. P. (1992). Adjusted P-values for simultaneous inference. Biometrics 48, 1005–1013. (Explains the adjusted P-value approach.)

Examples

```
{
data(Taraoceans)
pro.phylo <- Taraoceans$taxonomy[ ,'Phylum']
TaraOc<-list(Taraoceans$phychem,
as.data.frame(Taraoceans$pro.phylo),as.data.frame(Taraoceans$pro.NOGs))
TaraOc_1<-scale(TaraOc[[1]])
Normalization<-lapply(list(TaraOc[[2]],TaraOc[[3]]),
function(x){DataProcessing(x,Method='Compositional')})
colnames(Normalization[[1]])=pro.phylo
colnames(Normalization[[2]])=Taraoceans$GO
TaraOc<-list(TaraOc_1,Normalization[[1]],Normalization[[2]])
names(TaraOc)<-c('phychem','pro_phylo','pro_NOGs')
TaraOc<-lapply(TaraOc,as.data.frame)
Output<-LinkData(TaraOc,Scale =FALSE,Distance =
c('ScalarProduct','Euclidean','Euclidean'),nCluster=3)
dAB(Output,Data=list(TaraOc[[2]]))
}
```

DataProcessing *DataProcessing*

Description

function to Perform external datas' pre-processing. This function allows an external pre-processing of the datasets including on the analysis in three ways: Standard, Compositional (centered log ratio) and frequencies.

Usage

```
DataProcessing(Data = NULL, Method = "Standard")
```

Arguments

Data	a numeric data.frame.
Method	character indicating the method used to Data preprocessing. If data are continous, use 'Standard'. If Data are compositional, please use 'Compositional' and clr (centered log-ratios functions) transformations are performed. To compositional data, you also could use the option 'TSS' Total Sum Scaling follow up bray (Bray-Curtis) in distance option. The function also allows to processing frequencies- like data through 'FreqNorm' option. Note that when you use Compositional, we first sum 1 to all the counts (in order to performs the log transformation before).

Value

a data.frame with normalized data.

Author(s)

Laura M Zingatetti

Examples

```
{
data(Taraoceans)
Data<-Taraoceans$phychem
Data<-DataProcessing(Data,Method='Standard')
}
```

`DistStatis-class` *Class* `DistStatis` *DistStatis S4 class (linkHD:Multiple Heterogeneous Dataset Integration) Statis with Distance options implementation.*

Description

Class `DistStatis` DistStatis S4 class (linkHD:Multiple Heterogeneous Dataset Integration) Statis with Distance options implementation.

Features

1. DistStatis (implements Statis method incorporating Distance options to integrate multiple heterogeneous datasets)
2. Implement a LM (Linear Model) to variable selection
3. Incorporate a method to variable clustering
4. Incorporate some visualization tools: Compromise visualization, Relationship-visualization

Fields

- RV: Vectorial Correlation Matrix between studies.
- Inertia.RV: Inertia (%) explained for all tables.
- Euclid.Im: Euclidean Image of all studies.
- Inertia.comp: Inertia (%) explained for all dimensions of compromise matrix.
- Compromise.Coords: Projection of all observations in compromise (Coords).
- Compromise.Matrix: Compromise Matrix from statis methodology.
- RQO: Representation Quality of observations in compromise matrix.
- TableProjections: Projection of each table on Compromise configuration

@slot RV: Vectorial Correlation Matrix between studies. @slot Inertia.RV: Inertia (%) explained for all tables. @slot Euclid.Im: Euclidean Image of all studies. @slot Inertia.comp: Inertia (%) explained for all dimensions of compromise matrix. @slot Compromise.Coords: Projection of all observations in compromise (Coords). @slot Compromise.Matrix: Compromise Matrix from statis methodology. @slot RQO: Representation Quality of observations in compromise matrix. @slot TableProjections: Projection of each table on Compromise configuration

DistStatis-general-functions

DistStatis Getters for their respective slots.

@author Laura M Zingaretti

Examples

```
{
showClass('DistStatis')
}
```

Euclid_Im *Euclid_Im*

Description

Accessor to the Observations Image Euclidean, i.e. the projections from LinkData output.

Usage

```
## S4 method for signature 'DistStatis'
Euclid_Im(x)
```

Arguments

x	an object from DistSatis class.

Value

Euclid_Im Euclidean image of the input tables in LinData function.

Examples

```
{
data(Taraoceans)
pro.phylo <- Taraoceans$taxonomy[ ,'Phylum']
TaraOc<-list(Taraoceans$phychem,as.data.frame(Taraoceans$pro.phylo)
,as.data.frame(Taraoceans$pro.NOGs))
TaraOc_1<-scale(TaraOc[[1]])
Normalization<-lapply(list(TaraOc[[2]],TaraOc[[3]]),
function(x){DataProcessing(x,Method='Compositional')})
colnames(Normalization[[1]])=pro.phylo
colnames(Normalization[[2]])=Taraoceans$GO
TaraOc<-list(TaraOc_1,Normalization[[1]],Normalization[[2]])
names(TaraOc)<-c('phychem','pro_phylo','pro_NOGs')
TaraOc<-lapply(TaraOc,as.data.frame)
Output<-LinkData(TaraOc,Scale =FALSE,Distance = c('ScalarProduct','Euclidean','Euclidean'))
Euclid_Im(Output)
}
```

GlobalPlot *Global-Plot*

Description

this function outputs a plot from a DistStatis object. The plot shows the projection of the all common observation onto each subspace used at the integration step

Usage

```
## S4 method for signature 'DistStatis'
GlobalPlot(x)
```

Arguments

x	DistStatis class object.

Value

plotted GlobalPlot/s of the component/s of the given DistStatis object.

Author(s)

Laura M. Zingaretti

Examples

```
{

data(Taraoceans)
pro.phylo <- Taraoceans$taxonomy[ ,'Phylum']
TaraOc<-list(Taraoceans$phychem,as.data.frame(Taraoceans$pro.phylo),
as.data.frame(Taraoceans$pro.NOGs))
TaraOc_1<-scale(TaraOc[[1]])
Normalization<-lapply(list(TaraOc[[2]],TaraOc[[3]]),
function(x){DataProcessing(x,Method='Compositional')})
colnames(Normalization[[1]])=pro.phylo
colnames(Normalization[[2]])=Taraoceans$GO
TaraOc<-list(TaraOc_1,Normalization[[1]],Normalization[[2]])
names(TaraOc)<-c('phychem','pro_phylo','pro_NOGs')
TaraOc<-lapply(TaraOc,as.data.frame)
Output<-LinkData(TaraOc,Scale =FALSE,
Distance = c('ScalarProduct','Euclidean','Euclidean'))
GlobalPlot(Output) +
theme(panel.grid.major = element_blank(),
panel.grid.minor = element_blank(),
panel.background = element_blank(),
axis.line = element_line(colour = 'black'))


}
```

Inertia_comp *Inertia_comp*

Description

Accessor to explained inertia of compromise axis from LinkData output.

Usage

```
## S4 method for signature 'DistStatis'
Inertia_comp(x)
```

Arguments

x	an object from DistSatis class.

Value

Inertia_comp explained inertia for Compromise matrix from LinkData object

Examples

```
{
data(Taraoceans)
pro.phylo <- Taraoceans$taxonomy[ ,'Phylum']
TaraOc<-list(Taraoceans$phychem,as.data.frame(Taraoceans$pro.phylo)
,as.data.frame(Taraoceans$pro.NOGs))
TaraOc_1<-scale(TaraOc[[1]])
Normalization<-lapply(list(TaraOc[[2]],TaraOc[[3]]),
function(x){DataProcessing(x,Method='Compositional')})
colnames(Normalization[[1]])=pro.phylo
colnames(Normalization[[2]])=Taraoceans$GO
TaraOc<-list(TaraOc_1,Normalization[[1]],Normalization[[2]])
names(TaraOc)<-c('phychem','pro_phylo','pro_NOGs')
TaraOc<-lapply(TaraOc,as.data.frame)
Output<-LinkData(TaraOc,Scale =FALSE,Distance = c('ScalarProduct','Euclidean','Euclidean'))
Inertia_comp(Output)
}
```

Inertia_RV *Inertia_RV*

Description

Accessor to Inertia_RV from LinkData output.

Usage

```
## S4 method for signature 'DistStatis'
Inertia_RV(x)
```

Arguments

x	an object from DistSatis class.

Value

Inertia_RV explained inertia for RV matrix from LinkData object

Examples

```
{
data(Taraoceans)
pro.phylo <- Taraoceans$taxonomy[ ,'Phylum']
TaraOc<-list(Taraoceans$phychem,as.data.frame(Taraoceans$pro.phylo)
,as.data.frame(Taraoceans$pro.NOGs))
TaraOc_1<-scale(TaraOc[[1]])
Normalization<-lapply(list(TaraOc[[2]],TaraOc[[3]]),
function(x){DataProcessing(x,Method='Compositional')})
colnames(Normalization[[1]])=pro.phylo
colnames(Normalization[[2]])=Taraoceans$GO
TaraOc<-list(TaraOc_1,Normalization[[1]],Normalization[[2]])
names(TaraOc)<-c('phychem','pro_phylo','pro_NOGs')
TaraOc<-lapply(TaraOc,as.data.frame)
Output<-LinkData(TaraOc,Scale =FALSE,Distance = c('ScalarProduct','Euclidean','Euclidean'))
Inertia_RV(Output)
}
```

`LinkData` *LinkData: multiple heterogeneous dataset integration*

Description

Integrating multiple Heterogeneous Datasets stored into a list. This function makes Statis using Distances options. Statis is part of the PCA family and is based on singular value decomposition (SVD) and the generalized singular value decomposition (GSVD) of a matrix. This methodology aims to analyze several data sets of variables that were collected on the same set of observations. Originally, the comparisons were drawn from the compute of the scalar product between the different tables. In our approach, the condition is relaxing allowing the incorporation of different distances.

Usage

```
LinkData(Data, Distance = c(), Center = FALSE, Scale = FALSE,
  CorrelVector = TRUE, nCluster = 0, cl_method = "pam")
```

Arguments

`Data`	should be a list of dataframes or ExpressionSet data with the same length of the number of tables to be integrate. In each dataframe, the Observations (common elements on Statis) should be in rows and the variables should be in columns. Data also might be a MultiAssayExperiment object from MultiAssayExperiment package, a software for multi-omics experiments integration in Bioconductor.
`Distance`	Vector indicating which distance (including scalar product) should be applied to each study. If is missing, the scalar product is used. The vector lenght must be equal to the length of Data. Distance options: ScalarProduct, euclidean, manhattan, canberra, pearson, pearsonabs, spearman, spearmanabs, mahalanobis, BrayCurtis distance (please, use option Bray). For binary data, the distance can be jaccard, simple_matching, sokal_Sneath, Roger_Tanimoto, Dice, Hamman,

	Ochiai, Phi_Pearson, 'Gower&Legendre. Note that, use pre-processing option as compositional and Euclidean is the same than use Aitchison distance for compositional data.
Center	Logical. If TRUE, the data frame is centered by the mean. By default is FALSE. If you have tables with different characteristics (continous phenotypes, frecuencies, compositional data), we strongly recomendate normalize datasets as a previous step through DataProcessing option.
Scale	A logical value indicating whether the column vectors should be standardized by the rows weight, by default is FALSE. Note that all data into the list will be scaled. If you don't need normalizing all data, you could set this parameter as False and perform the normalization step externally by using DataProcessing function. If you have tables with different characteristics (continous phenotypes, frecuencies, compositional data), we strongly recomendate normalize datasets as a previous step through DataProcessing option.
CorrelVector	Logical. If TRUE (default), the RV matrix is computed using vectorial correlation, else the Hilbert-Smith distance is used.
nCluster	this variable indicates if common elements on the dataset should be grouped (by default is zero, i.e. no-cluster).
cl_method	categorical (pam or kmeans). pam is a robust version of classical kmeans algorithm.

Value

LinkData	DistStatis class object with the corresponding completed slots according to the given model

Author(s)

Laura M Zingatetti

References

1. Escoufier, Y. (1976). Operateur associe a un tableau de donnees. Annales de laInsee, 22-23, 165-178.
2. Escoufier, Y. (1987). The duality diagram: a means for better practical applications. En P. Legendre & L. Legendre (Eds.), Developments in Numerical Ecology, pp. 139-156, NATO Advanced Institute, Serie G. Berlin: Springer.
3. L'Hermier des Plantes, H. (1976). Structuration des Tableaux a Trois Indices de la Statistique. [These de Troisieme Cycle]. University of Montpellier, France.

Examples

```
{
data(Taraoceans)
pro.phylo <- Taraoceans$taxonomy[ ,'Phylum']
TaraOc<-list(Taraoceans$phychem,as.data.frame(Taraoceans$pro.phylo)
,as.data.frame(Taraoceans$pro.NOGs))
TaraOc_1<-scale(TaraOc[[1]])
```

```
Normalization<-lapply(list(TaraOc[[2]],TaraOc[[3]]),
function(x){DataProcessing(x,Method='Compositional')})
colnames(Normalization[[1]])=pro.phylo
colnames(Normalization[[2]])=Taraoceans$GO
TaraOc<-list(TaraOc_1,Normalization[[1]],Normalization[[2]])
names(TaraOc)<-c('phychem','pro_phylo','pro_NOGs')
TaraOc<-lapply(TaraOc,as.data.frame)
Output<-LinkData(TaraOc,Scale =FALSE,Distance = c('ScalarProduct','Euclidean','Euclidean'))
}
```

OTU2Taxa *OTU2Taxa*

Description

This function aggregates OTUs into their taxonomic characteristics (genus or level)and it analyses the most significant selected genera into each table. To each genera, the function returns the hypergeometric distribution function P(x>=X) to each count. The function also returns filtered data by counts higher than one. In both cases, we implemented -log(p+0.05), then a higher value means more significant, i.e., it is an enrichment genus or family.

Usage

```
OTU2Taxa(Selection, TaxonInfo, tableName, AnalysisLev = "Genus")
```

Arguments

Selection	list or data frame from VarSelection or dAB function
TaxonInfo	data.frame with taxonomic table associated to Data input. For instance, if Data comes from 16_S level, TaxoInfo should be a data.frame with 16_S associated taxonomic information. Note that the first column of this table must have the OTUs ids.
tableName	a character indicating the table name. For instance, if your data comes from 16_S, this parameter should be '16_S'. Note that, this argument must mutch with the names from the input list into LinkData function.
AnalysisLev	It is a character indicating if data should be aggregate to genera or family level

Value

List. The first element of this list contains all the selected taxa with their associated value from the hyperg distribution -log(p+0.05); the second element of this list have only taxas counting up to 1.

Author(s)

Laura M Zingatetti

References

1.
2. Da Wei Huang, B. T. S., & Lempicki, R. A. (2009). Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists. Nucleic acids research, 37(1), 1.
3. Zheng, Q., & Wang, X. J. (2008). GOEAST: a web-based software toolkit for Gene Ontology enrichment analysis. Nucleic acids research, 36(suppl_2), W358-W363.

Examples

```
{
data('Ruminotypes')
Normalization<-lapply(list(Ruminotypes$`16_S`,Ruminotypes$Archaea,Ruminotypes$`18_S`),
function(x){DataProcessing(x,Method='Compositional')})
Dataset<-Normalization
names(Dataset)<-c('16_S','Archaea','18_S')
#Running LinkData
Output<-LinkData(Dataset,Distance=rep('euclidean',3),
Scale = FALSE,Center=FALSE,nCluster = 3)
Select_Var<-VarSelection(Output,Data=Dataset,Crit = 'Rsquare',perc=0.9)
SignTaxa<-OTU2Taxa(Selection=VarTable(Select_Var),
TaxonInfo=Ruminotypes$Taxa_16S,tableName='16_S',AnalysisLev = 'Family')
Selected<-SignTaxa$TotalUp1
}
```

`Read_Data` *Read_Data: a fast way to data reading.*

Description

this function read all dataset in a folder and returns list needed to Link_Data function input.

Usage

```
Read_Data(Path = "")
```

Arguments

`Path`	path to folder containing all dataset to integrate

Value

`List`	List including all dataset into the parent directory. List names inherit the names of the files

Author(s)

Laura M Zingatetti

Examples

```
## Not run:
Datos<-Read_Data('Path to parent folder',common_elements=1)

## End(Not run)
```

RQO *RQO*

Description

Accessor to RQO (

Usage

```
## S4 method for signature 'DistStatis'
RQO(x)
```

Arguments

x	an object from DistSatis class.

Value

RQO Representation Quality of the observations in the compromise configuration from LinkData object

Examples

```
{
data(Taraoceans)
pro.phylo <- Taraoceans$taxonomy[ ,'Phylum']
TaraOc<-list(Taraoceans$phychem,as.data.frame(Taraoceans$pro.phylo)
,as.data.frame(Taraoceans$pro.NOGs))
TaraOc_1<-scale(TaraOc[[1]])
Normalization<-lapply(list(TaraOc[[2]],TaraOc[[3]]),
function(x){DataProcessing(x,Method='Compositional')})
colnames(Normalization[[1]])=pro.phylo
colnames(Normalization[[2]])=Taraoceans$GO
TaraOc<-list(TaraOc_1,Normalization[[1]],Normalization[[2]])
names(TaraOc)<-c('phychem','pro_phylo','pro_NOGs')
TaraOc<-lapply(TaraOc,as.data.frame)
Output<-LinkData(TaraOc,Scale =FALSE,Distance = c('ScalarProduct','Euclidean','Euclidean'))
RQO(Output)
}
```

`Ruminotypes` *Whole Ruminotypes dataset communities*

Description

Ruminotypes dataset contains communities (16_S, 18_S and Archaea) measuares from 65 loose-housed lactating Holstein cows. The study aims evaluating the relationships between communities and methane emmision yield

Format

A list with seven components:

16_S matrix with 61 rows and 1198 columns. Each row represents a sample and each column represent one normalized OTU.

Archea a matrix with 61 rows (samples) and 453 normalized columns (Archea IDs).

18_S a matrix with 61 rows and 107 normalized columns (protozoa level).

phenotype data frame with 61 rows and 5 columns representing methane emission levels with a set of corrections.

Taxa_16S a matrix with 1198 rows and 9 columns indicating the Taxa information for 16_S OTU.

Taxa_18S a matrix with 112 rows and 18 columns indicating the Taxa information for 18_S samples.

Taxa_Archea a matrix with 453 rows and 7 columns indicating the Taxa information for archea samples.

References

Ramayo-Caldas Y, Zingaretti LM, Bernard A, Estellé J, Popova M, Pons N, Bellot P, Mach N, Rau A, Roume H, Pérez-Enciso M, Faverdin N, Edouard N, Dusko S, Morgavi DP, Renand G. Identification of rumen microbial biomarkers linked to methane emission in Holstein dairy cows In press.

Examples

```
data(Ruminotypes)
```

sign_values *sign_values*

Description

Accessor to R2 or p values of the selected variables from VarSelection output.

Usage

```
## S4 method for signature 'VarSelection'
sign_values(x)
```

Arguments

x	an object from VarSelection class.

Value

sign_values, data.frame with the R2 or FDR p-value for each of the selected variables

Examples

```
{
data(Taraoceans)
pro.phylo <- Taraoceans$taxonomy[ ,'Phylum']
TaraOc<-list(Taraoceans$phychem,as.data.frame(Taraoceans$pro.phylo),
as.data.frame(Taraoceans$pro.NOGs))
TaraOc_1<-scale(TaraOc[[1]])
Normalization<-lapply(list(TaraOc[[2]],TaraOc[[3]]),
function(x){DataProcessing(x,Method='Compositional')})
colnames(Normalization[[1]])=pro.phylo
colnames(Normalization[[2]])=Taraoceans$GO
TaraOc<-list(TaraOc_1,Normalization[[1]],Normalization[[2]])
names(TaraOc)<-c('phychem','pro_phylo','pro_NOGs')
TaraOc<-lapply(TaraOc,as.data.frame)
Output<-LinkData(TaraOc,Scale =FALSE,
Distance = c('ScalarProduct','Euclidean','Euclidean'))
Selection<-VarSelection(Output,TaraOc,Crit='Rsquare',perc=0.95)
sign_values(Selection)
}
```

Taraoceans *TARA Oceans datasets*

Description

TARA Oceans was an expedition allowing to the study of plankton communities and their interactions with enviromental variables. This dataset was taken from mixkernel package (https://cran.r-project.org/web/packages/mixKernel/index.html). Data constists on 139 prokaryotic-enriched samples collected from 68 stations and spread across threedepth layers: the surface (SRF), the deep chlorophyll maximum (DCM) layer and the mesopelagic(MES) zones. Samples were located in height different oceans or seas: Indian Ocean (IO), Mediter-ranean Sea (MS), North Atlantic Ocean (NAO), North Pacific Ocean (NPO), Red Sea (RS), SouthAtlantic Ocean (SAO), South Pacific Ocean (SPO) and South Ocean (SO).

Usage

```
data("Taraoceans")
```

Format

A list with seven components:

phychemdata matrix with 139 rows and 22 columns. Each row represents a sample and each column an environmental variable.

pro.phylo a matrix with 139 rows (samples) and 356 columns (prokaryotic OTUs).

taxonomy a matrix with 356 rows (prokaryotic OTUs) and 6 columns indicating the taxonomy of each OTU.

phylogenetic.tree a phylo object (see package 'ape') representing the prokaryotic OTUs

pro.NOGs a matrix with 139 rows (samples) and 638 columns (NOGs).

GO a list with the names of Gene Ontologies.

sample a list containing three following entries (all three are character vectors):name(samplename),ocean(oceanic region of the sample) and depth(sample depth)

References

Sunagawa S., Coelho L.P., Chaffron S., Kultima J.R., Labadie K., Salazar F., Djahanschiri B., ZellerG., Mende D.R., Alberti A., Cornejo-Castillo F., Costea P.I., Cruaud C., d'Oviedo F., Engelen S.,Ferrera I., Gasol J., Guidi L., Hildebrand F., Kokoszka F., Lepoivre C., Lima-Mendez G., PoulainJ., Poulos B., Royo-Llonch M., Sarmento H., Vieira-Silva S., Dimier C., Picheral M., Searson S.,Kandels-Lewis S.,TaraOceans coordinators, Bowler C., de Vargas C., Gorsky G., Grimsley N.,Hingamp P., Iudicone D., Jaillon O., Not F., Ogata H., Pesant S., Speich S., Stemmann L., SullivanM., Weissenbach J., Wincker P., Karsenti E., Raes J., Acinas S. and Bork P. (2015). Structure andfunction of the global ocean microbiome.Science,348, 6237

Examples

```
data(Taraoceans)
```

Trajectories *Trajectories*

Description

Accessor to projections into the common configuration, i.e. compromise of each input table from LinkData output.

Usage

```
## S4 method for signature 'DistStatis'
Trajectories(x)
```

Arguments

x	an object from DistSatis class.

Value

Trajectories contains a list of the projections of each input table into the common configuration, i.e. the compromise from LinkData object

Examples

```
{
data(Taraoceans)
pro.phylo <- Taraoceans$taxonomy[ ,'Phylum']
TaraOc<-list(Taraoceans$phychem,as.data.frame(Taraoceans$pro.phylo)
,as.data.frame(Taraoceans$pro.NOGs))
TaraOc_1<-scale(TaraOc[[1]])
Normalization<-lapply(list(TaraOc[[2]],TaraOc[[3]]),
function(x){DataProcessing(x,Method='Compositional')})
colnames(Normalization[[1]])=pro.phylo
colnames(Normalization[[2]])=Taraoceans$GO
TaraOc<-list(TaraOc_1,Normalization[[1]],Normalization[[2]])
names(TaraOc)<-c('phychem','pro_phylo','pro_NOGs')
TaraOc<-lapply(TaraOc,as.data.frame)
Output<-LinkData(TaraOc,Scale =FALSE,Distance = c('ScalarProduct','Euclidean','Euclidean'))
Trajectories(Output)
}
```

Variables *Variables*

Description

Accessor to selected Variables from VarSelection output.

Usage

```
## S4 method for signature 'VarSelection'
Variables(x)
```

Arguments

x	an object from VarSelection class.

Value

Variables list of selected variables from VarSelection object

Examples

```
{
data(Taraoceans)
pro.phylo <- Taraoceans$taxonomy[ ,'Phylum']
TaraOc<-list(Taraoceans$phychem,as.data.frame(Taraoceans$pro.phylo),
as.data.frame(Taraoceans$pro.NOGs))
TaraOc_1<-scale(TaraOc[[1]])
Normalization<-lapply(list(TaraOc[[2]],TaraOc[[3]]),
function(x){DataProcessing(x,Method='Compositional')})
colnames(Normalization[[1]])=pro.phylo
colnames(Normalization[[2]])=Taraoceans$GO
TaraOc<-list(TaraOc_1,Normalization[[1]],Normalization[[2]])
names(TaraOc)<-c('phychem','pro_phylo','pro_NOGs')
TaraOc<-lapply(TaraOc,as.data.frame)
Output<-LinkData(TaraOc,Scale =FALSE,
Distance = c('ScalarProduct','Euclidean','Euclidean'))
Selection<-VarSelection(Output,TaraOc,Crit='Rsquare',perc=0.95)
Variables(Selection)
}
```

VarSelection *Variable Selection*

Description

Function to do variable selection using a Regression Biplot methodology. This function calculates the regression biplot on the compromise matrix. Biplot can be understood as the decomposition of a target matrix ($Y=XB$). Here, Y is the matrix containing all variables taken into account in the analisis,X is the matrix containing the explaining variables, i.e., the coordinates of compromise matrix and finally, B are the regression coefficients to be estimated. Then, the method is interpreted as a general linear regression into the X matrix ($Y_hat=X(X'X)^(-1)X'Y$) and the matrix $X(X'X)^(-1)X'$ is the projection matrix onto the compromise configuration. We use a classical linear model to obtain the regressors coefficients, however the model could be extended and alternatives methods are able to use. The quality of the regression biplot is measured using the proportion of explained variance by each regression (adjusted r squared coefficient).

Usage

```
VarSelection(x, Data, intercept = FALSE, model = "LM",
  Crit = "Rsquare", perc = 0.9, nDims = 2, Normalize = FALSE)
```

Arguments

x	is an object of DistStatis Class.
Data	should be a list of data.frame or ExpressionSet data with the same length of the number of tables to be integrate. In each dataframe, the Observations (common elements on Statis) should be in rows and the variables should be in columns. Data are the same data used to obtained the compromise configuration.It also can be a MultissayExperiment object, please check help of LinkData function and the package vignette.
intercept	Logical. If is TRUE, the models with intercept are computed, else the intercept is zero.
model	character. 'LM' for classical lm model. We've planned to implemening alternative models in the future.
Crit	Character indicating the variable selection criteria.You could chose 'Rsquare' or 'p-val'.
perc	The value of percentil that indicate how much data than are selected.
nDims	Numeric that indicates the number of dimensions to use for do the model. Default is 2.
Normalize	Logical. If is TRUE, the response variable in each model is normalized.

Value

a

VarSelection	VarSelection class with the corresponding completed slots according to the given model

Author(s)

Laura M Zingatetti

References

1.

2. Gabriel, K. (1971). The biplot graphic display of matrices with application to principal component analysis. Biometrika 58(3), 453–467.

3. Gower, J. & Hand, D. (1996). Biplots, Monographs on statistics and applied probability. 54. London: Chapman and Hall., 277 pp.

4. Greenacre, M. J. (2010). Biplots in practice. Fundacion BBVA.

Examples

```
{
data(Taraoceans)
pro.phylo <- Taraoceans$taxonomy[ ,'Phylum']
TaraOc<-list(Taraoceans$phychem,as.data.frame(Taraoceans$pro.phylo),
as.data.frame(Taraoceans$pro.NOGs))
TaraOc_1<-scale(TaraOc[[1]])
Normalization<-lapply(list(TaraOc[[2]],TaraOc[[3]]),
function(x){DataProcessing(x,Method='Compositional')})
colnames(Normalization[[1]])=pro.phylo
colnames(Normalization[[2]])=Taraoceans$GO
TaraOc<-list(TaraOc_1,Normalization[[1]],Normalization[[2]])
names(TaraOc)<-c('phychem','pro_phylo','pro_NOGs')
TaraOc<-lapply(TaraOc,as.data.frame)
Output<-LinkData(TaraOc,Scale =FALSE,
Distance = c('ScalarProduct','Euclidean','Euclidean'))
Selection<-VarSelection(Output,TaraOc,Crit='Rsquare',perc=0.95)
}
```

VarSelection-class *class VarSelection*

Description

Class `VarSelection` S4 class (linkHD: integrating multiple heterogeneous datasets) VarSelection is a class to perform variable selection from a DistStatis object.

Features

1. class to perform variable selection using Linear Regression Biplot onto the Compromise-Subespace
2. This method allow variable selection and classification

Fields

- Variables return all the selected variables (and the frecuency of selection).
- Coordinates represent the coordenates (Betas coeficients on LM) of the selected variables.
- VarTable data.frame indicating which table selected variables come from.
- values data.frame contains the R2 or pvalue (fdr) of selected variables (it depends of the Crit used).

Accesors

- Variables return all the selected variables (and the frecuency of selection).
- Coordinates represent the coordenates (Betas coeficients on LM) of the selected variables.
- VarTable dataframe indicating the table that each selected variable comes from.
- values data.frame which contains the R2 or pvalue (fdr) of selected variables (it depends of the Crit used).

VarSelection-class-general-functions

print Generated basic output for VarSelection class

@author Laura M Zingaretti

References

1. Gabriel, K. (1971). The biplot graphic display of matrices with application to principal component analysis. Biometrika 58(3), 453–467.
2. Gower, J. & Hand, D. (1996). Biplots, Monographs on statistics and applied probability. 54. London: Chapman and Hall., 277 pp.

Examples

```
{
showClass('VarSelection')
}
```

VarTable *VarTable*

Description

Accessor to Table with the selected variables from VarSelection output.

Usage

```
## S4 method for signature 'VarSelection'
VarTable(x)
```

Arguments

x	an object from VarSelection class.

Value

VarTable data.frame with the name of input tables in the LinkData function

Examples

```
{
data(Taraoceans)
pro.phylo <- Taraoceans$taxonomy[ ,'Phylum']
TaraOc<-list(Taraoceans$phychem,as.data.frame(Taraoceans$pro.phylo),
as.data.frame(Taraoceans$pro.NOGs))
TaraOc_1<-scale(TaraOc[[1]])
Normalization<-lapply(list(TaraOc[[2]],TaraOc[[3]]),
function(x){DataProcessing(x,Method='Compositional')})
colnames(Normalization[[1]])=pro.phylo
colnames(Normalization[[2]])=Taraoceans$GO
TaraOc<-list(TaraOc_1,Normalization[[1]],Normalization[[2]])
names(TaraOc)<-c('phychem','pro_phylo','pro_NOGs')
TaraOc<-lapply(TaraOc,as.data.frame)
Output<-LinkData(TaraOc,Scale =FALSE,
Distance = c('ScalarProduct','Euclidean','Euclidean'))
Selection<-VarSelection(Output,TaraOc,Crit='Rsquare',perc=0.95)
VarTable(Selection)
}
```

Var_coordinates *Var_coordinates*

Description

Accessor to the coordinates projections into the compromise configuration of the selected variables from VarSelection output.

Usage

```
## S4 method for signature 'VarSelection'
Var_coordinates(x)
```

Arguments

x	an object from VarSelection class.

Value

Var_Coordinates, Coordinates of variables into the common configuration, i.e. the compromise from LinkData function

Examples

```
{
data(Taraoceans)
pro.phylo <- Taraoceans$taxonomy[ ,'Phylum']
TaraOc<-list(Taraoceans$phychem,as.data.frame(Taraoceans$pro.phylo),
as.data.frame(Taraoceans$pro.NOGs))
TaraOc_1<-scale(TaraOc[[1]])
Normalization<-lapply(list(TaraOc[[2]],TaraOc[[3]]),
function(x){DataProcessing(x,Method='Compositional')})
colnames(Normalization[[1]])=pro.phylo
colnames(Normalization[[2]])=Taraoceans$GO
TaraOc<-list(TaraOc_1,Normalization[[1]],Normalization[[2]])
names(TaraOc)<-c('phychem','pro_phylo','pro_NOGs')
TaraOc<-lapply(TaraOc,as.data.frame)
Output<-LinkData(TaraOc,Scale =FALSE,
Distance = c('ScalarProduct','Euclidean','Euclidean'))
Selection<-VarSelection(Output,TaraOc,Crit='Rsquare',perc=0.95)
Var_coordinates(Selection)
}
```

Index

∗ **datasets**
    Ruminotypes, 19
    Taraoceans, 21

compromise_coords, 4
compromise_coords,DistStatis-method *(*compromise_coords*)*, 4
Compromise_matrix, 5
Compromise_matrix,DistStatis-method *(*Compromise_matrix*)*, 5
CompromisePlot, 3
CompromisePlot,DistStatis-method *(*CompromisePlot*)*, 3
correl, 5
correl,DistStatis-method *(*correl*)*, 5
CorrelationPlot, 6
CorrelationPlot,DistStatis-method *(*CorrelationPlot*)*, 6

dAB, 7
DataProcessing, 9
DistStatis-class, 10

Euclid_Im, 11
Euclid_Im,DistStatis-method *(*Euclid_Im*)*, 11

GlobalPlot, 11
GlobalPlot,DistStatis-method *(*GlobalPlot*)*, 11

Inertia_comp, 12
Inertia_comp,DistStatis-method *(*Inertia_comp*)*, 12
Inertia_RV, 13
Inertia_RV,DistStatis-method *(*Inertia_RV*)*, 13

LinkData, 14

OTU2Taxa, 16

Read_Data, 17
RQO, 18
RQO,DistStatis-method *(*RQO*)*, 18
Ruminotypes, 19

sign_values, 20
sign_values,VarSelection-method *(*sign_values*)*, 20

Taraoceans, 21
Trajectories, 22
Trajectories,DistStatis-method *(*Trajectories*)*, 22

Var_coordinates, 27
Var_coordinates,VarSelection-method *(*Var_coordinates*)*, 27
Variables, 23
Variables,VarSelection-method *(*Variables*)*, 23
VarSelection, 24
VarSelection-class, 25
VarTable, 26
VarTable,VarSelection-method *(*VarTable*)*, 26