Package ‘robCompositions’

November 8, 2018

Type Package
Title Robust Estimation for Compositional Data
Version 2.0.9
Date 2018-11-07
Depends R (>= 3.0.0), robustbase, ggplot2, data.table, e1071, pls
LinkingTo Rcpp
Imports car, cvTools, rrcov, cluster, fpc, GGally, kernlab, MASS, mclust, Rcpp, sROC, VIM, zCompositions
Suggests knitr
VignetteBuilder knitr
Maintainer Matthias Templ <matthias.templ@gmail.com>
Description Methods for analysis of compositional data including robust methods, imputation, methods to replace rounded zeros, (robust) outlier detection for compositional data, (robust) principal component analysis for compositional data, (robust) factor analysis for compositional data, (robust) discriminant analysis for compositional data (Fisher rule), robust regression with compositional predictors and (robust) Anderson-Darling normality tests for compositional data as well as popular log-ratio transformations (addLR, cenLR, isomLR, and their inverse transformations). In addition, visualisation and diagnostic tools are implemented as well as high and low-level plot functions for the ternary diagram.
License GPL (>= 2)
LazyLoad yes
LazyData true
RoxygenNote 6.1.0
NeedsCompilation yes
Author Matthias Templ [aut, cre],
Karel Hron [aut],
Peter Filzmoser [aut],
Petra Kynclova [ctb],
Jan Walach [ctb],
R topics documented:

robCompositions-package .................................................. 4
addLR .............................................................................. 6
addLRinv ......................................................................... 8
aDist ................................................................................ 9
adjust ............................................................................. 11
adtest .............................................................................. 12
adtestWrapper ................................................................. 13
ageCatWorld ................................................................... 15
alcohol ............................................................................. 16
alcoholreg ....................................................................... 17
arcticLake ....................................................................... 17
balances .......................................................................... 18
balZav ............................................................................... 20
biomarker ....................................................................... 20
biplot.factanal .................................................................. 22
biplot.pcaCoDa ................................................................ 23
bootnComp ..................................................................... 24
cancer ............................................................................. 25
cancerMN ....................................................................... 26
ced ................................................................................... 27
cenLR ............................................................................. 28
cenLRinv ........................................................................ 29
chorizonDL ..................................................................... 30
clustCoDa ....................................................................... 32
clustCoDa_qmode ............................................................ 34
coffee .............................................................................. 35
compareMahal ................................................................. 36
constSum ......................................................................... 37
coord .............................................................................. 38
corCoDa ......................................................................... 39
daCoDa ........................................................................... 40
daFisher .......................................................................... 41
economy ......................................................................... 44
educFM ............................................................................ 45
election ........................................................................... 46
electionATbp ................................................................... 47
employment ..................................................................... 48
employment_df ............................................................... 48
expenditures .................................................................... 49
R topics documented:

- expendituresEU
- GDPsatis
- gems
- gjovik
- gm
- gmean_sum
- govexp
- haplogroups
- ilr.2x2
- impAll
- impCoda
- impKNNa
- impRZalr
- impRZilr
- imputeBDLs
- imputeUDLs
- ind2x2
- indTab
- instw
- int2x2
- intArray
- intTab
- isic32
- laborForce
- lifeExpGdp
- lmCoDaX
- machineOperators
- mcad
- missPatterns
- mortality
- mortality_tab
- nutrients
- nutrients_branded
- orthbasis
- outCoDa
- payments
- pcaCoDa
- perturbation
- pfa
- phd
- phd_totals
- pivotCoord
- plot.imp
- plot.pcaCoDa
- precipitation
- print.imp
- production
- pTab

50 51 52 54 56 57 58 59 60 61 62 63 65 67 68 71 73 74 75 76 77 78 79 79 80 81 82 83 84 85 86 87 89 90 91 93 94 96 97 99 100 101 102 105 106 107 108 109
robCompositions-package

Robust Estimation for Compositional Data.

Description

The package contains methods for imputation of compositional data including robust methods, (robust) outlier detection for compositional data, (robust) principal component analysis for compositional data, (robust) factor analysis for compositional data, (robust) discriminant analysis (Fisher rule) and (robust) Anderson-Darling normality tests for compositional data as well as popular logratio transformations (alr, clr, ilr, and their inverse transformations).

Details

<table>
<thead>
<tr>
<th>Package:</th>
<th>robCompositions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Package</td>
</tr>
<tr>
<td>Version:</td>
<td>1.3.3</td>
</tr>
<tr>
<td>Date:</td>
<td>2009-11-28</td>
</tr>
<tr>
<td>License:</td>
<td>GPL 2</td>
</tr>
<tr>
<td>LazyLoad:</td>
<td>yes</td>
</tr>
</tbody>
</table>
Author(s)
Matthias Templ, Peter Filzmoser, Karel Hron,
Maintainer: Matthias Templ <templ@tuwien.ac.at>

References

Examples

```
## k nearest neighbor imputation
data(expenditures)
expenditures[1,3]
expenditures[1,3] <- NA
impKNNa(expenditures)$xImp[1,3]

## iterative model based imputation
data(expenditures)
x <- expenditures
x[1,3]
x[1,3] <- NA
xi <- impCoda(x)$xImp
xi[1,3]
s1 <- sum(x[1,-3])
impS <- sum(xi[1,-3])
xi[,3] * s1/impS

xi <- impKNNa(expenditures)
xi
summary(xi)
## Not run: plot(xi, which=1)
plot(xi, which=2)
plot(xi, which=3)

## pca
data(expenditures)
```
addLR
Additive log-ratio transformation

Description
The additive log-ratio transformation moves D-part compositional data from the simplex into a (D-1)-dimensional real space.

Usage
addLR(x, ivar = ncol(x), base = exp(1))
Arguments

- **x**: D-part compositional data
- **ivar**: Rationing part
- **base**: a positive or complex number: the base with respect to which logarithms are computed. Defaults to \( \exp(1) \).

Details

The compositional parts are divided by the rationing part before the logarithm is taken.

Value

A list of class “alr” which includes the following content:

- **x.alr**: the transformed data
- **varx**: the rationing variable
- **ivar**: the index of the rationing variable, indicating the column number of the rationing variable in the data matrix \( x \)
- **cnames**: the column names of \( x \)

The additional information such as **cnames** or **ivar** is useful when a back-transformation is applied on the ‘same’ data set.

Author(s)

Matthias Templ

References


See Also

- **addLRinv**, **pivotCoord**

Examples

```r
data(arcticLake)
X <- arcticLake
X.alr <- addLR(X, 2)
Y <- addLRinv(X.alr)
## This exactly fulfills:
addLRinv(addLR(X, 3))
data(expenditures)
X <- expenditures
Y <- addLRinv(addLR(X, 5))
head(X)
```
**addLRinv**

Additive logistic transformation

Inverse additive log-ratio transformation, often called additive logistic transformation.

Usage

```r
addLRinv(x, cnames = NULL, ivar = NULL, useClassInfo = TRUE)
```

Arguments

- `x`: data set, object of class “alr”, “matrix” or “data.frame”
- `cnames`: column names. If the object is of class “alr” the column names are chosen from therein.
- `ivar`: index of the rationing part. If the object is of class “alr” the column names are chosen from therein. If not and ivar is not provided by the user, it is assumed that the rationing part was the last column of the data in the simplex.
- `useClassInfo`: if FALSE, the class information of object `x` is not used.

Details

The function allows also to preserve absolute values when class info is provided. Otherwise only the relative information is preserved.

Value

the transformed data matrix

Author(s)

Matthias Templ

References

See Also

pivotCoordInv, cenLRinv, cenLR, addLR

Examples

data(arcticLake)
x <- arcticLake
x.alr <- addLR(x, 2)
y <- addLRinv(x.alr)
## This exactly fulfills:
addLRinv(addLR(x, 3))
data(expenditures)
x <- expenditures
y <- addLRinv(addLR(x, 5, 2))
head(x)
head(y)
## --> absolute values are preserved as well.

## preserve only the ratios:
addLRinv(x.alr, ivar=2, useClassInfo=FALSE)

---

aDist  Aitchison distance

Description

Computes the Aitchison distance between two observations, between two data sets or within observations of one data set.

Usage

aDist(x, y = NULL)

iprod(x, y)

Arguments

x  a vector, matrix or data.frame

y  a vector, matrix or data.frame with equal dimension as x or NULL.

Details

This distance measure accounts for the relative scale property of the Aitchison distance. It measures the distance between two compositions if x and y are vectors. It evaluates the sum of the distances between x and y for each row of x and y if x and y are matrices or data frames. It computes a n times n distance matrix (with n the number of observations/compositions) if only x is provided.
The underlying code is partly written in C and allows a fast computation also for large data sets whenever \( y \) is supplied.

**Value**

The Aitchison distance between two compositions or between two data sets, or a distance matrix in case codey is not supplied.

**Author(s)**

Matthias Templ, Bernhard Meindl

**References**


**See Also**

pivotCoord

**Examples**

```r
data(expenditures)
x <- xOrig <- expenditures
## Aitchison distance between two 2 observations:
aDist(x[1, ], x[2, ])

## Aitchison distance of x:
aDist(x)

## Example of distances between matrices:
## set some missing values:

## impute them:
xImp <- impCoda(x, method="ltsReg")$xImp

## calculate the relative Aitcheson distance between xOrig and xImp:
aDist(xOrig, xImp)

data("expenditures")
aDist(expenditures)
x <- expenditures[, 1]
y <- expenditures[, 2]
```

*aDist*
adjust

adjust(x, y)
adjust(expenditures, expenditures)

Description
Results from the model based iterative methods provides the results in another scale (but the ratios are still the same). This function rescale the output to the original scale.

Usage
adjust(x)

Arguments
x object from class ‘imp’

Details
It is self-explaining if you try the examples.

Value
The object of class ‘imp’ but with the adjusted imputed data.

Author(s)
Matthias Templ

References

See Also
impCoda
Examples

```r
data(expenditures)
x <- expenditures
xi <- impCoda(x)
x
xi$xImp
adjust(xi)$xImp
```

---

**adtest**

**Anderson-Darling Normality Tests**

**Description**

This function provides three kinds of Anderson-Darling Normality Tests (Anderson and Darling, 1952).

**Usage**

```r
adtest(x, R = 1000, locscatt = "standard")
```

**Arguments**

- `x` either a numeric vector, or a data.frame, or a matrix
- `R` Number of Monte Carlo simulations to obtain p-values
- `locscatt` standard for classical estimates of mean and (co)variance. robust for robust estimates using `covMcd()` from package robustbase

**Details**

Three version of the test are implemented (univariate, angle and radius test) and it depends on the data which test is chosen.

If the data is univariate the univariate Anderson-Darling test for normality is applied.

If the data is bivariate the angle Anderson-Darling test for normality is performed out.

If the data is multivariate the radius Anderson-Darling test for normality is used.

If ‘locscatt’ is equal to “robust” then within the procedure, robust estimates of mean and covariance are provided using `covMcd()` from package robustbase.

To provide estimates for the corresponding p-values, i.e. to compute the probability of obtaining a result at least as extreme as the one that was actually observed under the null hypothesis, we use Monte Carlo techniques where we check how often the statistic of the underlying data is more extreme than statistics obtained from simulated normal distributed data with the same (column-wise-) mean(s) and (co)variance.
Value

statistic The result of the corresponding test statistic
method The chosen method (univariate, angle or radius)
p.value p-value

Note

These functions are use by adtestWrapper.

Author(s)

Karel Hron, Matthias Templ

References


See Also

adtestWrapper

Examples

```r
adtest(rnorm(100))
data(machineOperators)
x <- machineOperators
adtest(pivotCoord(x[,1:2]))
adtest(pivotCoord(x[,1:3]))
adtest(pivotCoord(x))
adtest(pivotCoord(x[,1:2]), locscatt="robust")
```

Description

A set of Anderson-Darling tests (Anderson and Darling, 1952) are applied as proposed by Aitchison (Aichison, 1986).
Usage

adtestWrapper(x, alpha = 0.05, R = 1000, robustEst = FALSE)

## S3 method for class 'adtestWrapper'
print(x, ...)

## S3 method for class 'adtestWrapper'
summary(object, ...)

Arguments

x compositional data of class data.frame or matrix
alpha significance level
R Number of Monte Carlo simulations in order to provide p-values.
robustEst logical
... additional parameters for print and summary passed through
object an object of class adtestWrapper for the summary method

Details

First, the data is transformed using the ‘ilr’-transformation. After applying this transformation
- all (D-1)-dimensional marginal, univariate distributions are tested using the univariate Anderson-
Darling test for normality.
- all 0.5 (D-1)(D-2)-dimensional bivariate angle distributions are tested using the Anderson-Darling
angle test for normality.
- the (D-1)-dimensional radius distribution is tested using the Anderson-Darling radius test for nor-
mality.

A print and a summary method are implemented. The latter one provides a similar output is prop-
osed by (Pawlowsky-Glahn, et al. 2008). In addition to that, p-values are provided.

Value

res a list including each test result
check information about the rejection of the null hypothesis
alpha the underlying significance level
info further information which is used by the print and summary method.
est “standard” for standard estimation and “robust” for robust estimation

Author(s)

Matthias Templ and Karel Hron
References


See Also

adtest, pivotCoord

Examples

```r
data(machineOperators)
a <- adtestWrapper(machineOperators, R=50) # choose higher value of R
summary(a)
```

---

**ageCatWorld**  
*child, middle and elderly population*

Description

Percentages of children, middle generation and elderly population in 195 countries.

Usage

```r
data(ageCatWorld)
```

Format

A data frame with 195 rows and 4 variables

Details

- `<15` Percentage of people with age below 15
- `15-60` Percentage of people with age between 15 and 60
- `60+` Percentage of people with age above 60
- `country` country of origin

The rows sum up to 100.

Author(s)

extracted by Karel Hron and Eva Fiserova, implemented by Matthias Templ
## References


## Examples

```r
data(ageCatWorld)
str(ageCatWorld)
summary(ageCatWorld)
rowSums(ageCatWorld[, 1:3])
ternaryDiag(ageCatWorld[, 1:3])
plot(pivotCoord(ageCatWorld[, 1:3]))
```

### alcohol

<table>
<thead>
<tr>
<th>alcohol</th>
<th>alcohol consumptions by country and type of alcohol</th>
</tr>
</thead>
</table>

### Description

- **country** Country
- **year** Year
- **beer** Consumption of pure alcohol on beer (in percentages)
- **wine** Consumption of pure alcohol on wine (in percentages)
- **spirits** Consumption of pure alcohol on spirits (in percentages)
- **other** Consumption of pure alcohol on other beverages (in percentages)

### Usage

```r
data(alcohol)
```

### Format

A data frame with 193 rows and 6 variables

### Author(s)

Matthias Templ <matthias.templ@tuwien.ac.at>

### Source

Transfered from the World Health Organisation website.

### Examples

```r
data("alcohol")
str(alcohol)
summary(alcohol)
```
alcoholreg

*alcoholreg*

*regional alcohol per capita (15+) consumption by WHO region*

**Description**

- **country** Country
- **year** Year
- **recorded** Recorded alcohol consumption
- **unrecorded** Unrecorded alcohol consumption

**Usage**

data(alcoholreg)

**Format**

A data frame with 6 rows and 4 variables

**Author(s)**

Matthias Templ <matthias.templ@tuwien.ac.at>

**Source**

Transferred from the World Health Organisation website.

**Examples**

data("alcoholreg")
alcoholreg

---

*arcticLake*

*arctic lake sediment data*

**Description**

Sand, silt, clay compositions of 39 sediment samples at different water depths in an Arctic lake. This data set can be found on page 359 of the Aitchison book (see reference).

**Usage**

data(arcticLake)
balances

Format
A data frame with 39 rows and 3 variables

Details
- sand numeric vector of percentages of sand
- silt numeric vector of percentages of silt
- clay numeric vector of percentages of clay
The rows sum up to 100, except for rounding errors.

Author(s)
Matthias Templ <matthias.templ@tuwien.ac.at>

References

Examples

data(arcticLake)
str(arcticLake)
summary(arcticLake)
rowSums(arcticLake)
ternaryDiag(arcticLake)
plot(pivotCoord(arcticLake))

balances                  Balance calculation

Description
Given a D-dimensional compositional data set and a sequential binary partition, the function balances calculates the balances in order to express the given data in the (D-1)-dimensional real space.

Usage
balances(x, y)

Arguments
- x data frame or matrix, typically compositional data
- y binary partition
Details

The sequential binary partition constructs an orthonormal basis in the (D-1)-dimensional hyperplane in real space.

Value

| balances | The balances represent orthonormal coordinates which allow an interpretation in sense of groups of compositional parts. Output is a matrix, the D-1 columns contain balance coordinates of the observations in the rows.
| V | A D\times(D-1) contrast matrix associated with the orthonormal basis, corresponding to the sequential binary partition (in clr coordinates).

Author(s)

Veronika Pintar, Karel Hron, Matthias Templ

References


Examples

```r
data(expenditures, package = "robCompositions")
y1 <- data.frame(c(1,1,1,-1,-1),c(1,1,-1,0,0),
                 c(0,0,-1,0,0),c(0,0,0,1,-1))
y2 <- data.frame(c(1,1,1,-1,-1),c(1,0,-1,0,0),
                 c(1,1,1,-1,1),c(0,-1,0,1,0))
y3 <- data.frame(c(1,1,1,-1),c(-1,-1,-1,0,0),
                 c(-1,1,1,0,0),c(-1,1,0,0,0))
y4 <- data.frame(c(1,1,1,-1),c(0,0,0,-1,1),
                 c(-1,1,0,0),c(-1,1,0,0,0))
y5 <- data.frame(c(1,1,1,-1),c(-1,-1,1,0,0),
                 c(0,0,0,-1,1),c(-1,1,0,0,0))
b1 <- balances(expenditures, y1)
b2 <- balances(expenditures, y5)
b1$balances
b2$balances

data(machineOperators)
sbp <- data.frame(c(1,1,-1,-1),c(-1,1,0,0),
                   c(0,0,1,-1))
b1$balances(machineOperators, sbp)
```
balZav  

*New average symmetric coordinates*

**Description**

New average symmetric coordinates

**Usage**

`balZav(x)`

**Arguments**

- `x`: compositional data

**Value**

`zNav`: A matrix of new average symmetric coordinates.

**Author(s)**

Petra Kynclova

biomarker  

**Description**

The function for identification of biomakers and outlier diagnostics as described in paper "Robust biomarker identification in a two-class problem based on pairwise log-ratios"

**Usage**

```r
biomarker(x, cut = qnorm(0.975, 0, 1), g1, g2, type = "tau",
          diag = TRUE, plot = FALSE, diag.plot = FALSE)
```

```r
## S3 method for class 'biomarker'
plot(x, cut = qnorm(0.975, 0, 1), type = "Vstar",
     ...)
```

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

```r
## S3 method for class 'biomarker'
summary(object, ...)
```
Arguments

- `x` data frame
- `cut` cut-off value, initially set as 0.975 quantile of standard normal distribution
- `g1` vector with locations of observations of group 1
- `g2` vector with locations of observations of group 2
- `type` type of estimation of the variation matrix. Possible values are "sd", "mad" and "tau", representing Standard deviation, Median absolute deviation and Tau estimator of scale
- `diag` logical, indicating whether outlier diagnostic should be computed
- `plot` logical, indicating whether Vstar values should be plotted
- `diag.plot` logical, indicating whether outlier diagnostic plot should be made
- `...` further arguments can be passed through
- `object` object of class biomarker

Details

Robust biomarker identification and outlier diagnostics

The method computes variation matrices separately with observations from both groups and also together with all observations. Then, $V$ statistics is then computed and normalized. The variables, for which according $V^*$ values are bigger than the cut-off value are considered as biomarkers.

Value

The function returns object of type "biomarker". Functions `print`, `plot` and `summary` are available.

- `biom.ident` List of $V$, Vstar, biomarkers
- `V` Values of $V$ statistics
- `Vstar` Normalizes values of $V$ statistics ($V^*$ values))
- `biomarkers` Logical value, indicating if certain variable was identified as biomarker
- `diag` Outlier diagnostics (returned only if `diag=TRUE`)

Author(s)

Jan Walach

See Also

`plot.biomarker`
Examples

# Data simulation
set.seed(4523)
n <- 40; p <- 50
r <- runif(p, min = 1, max = 10)
conc <- runif(p, min = 0, max = 1)*5+matrix(1,p,1)*5
a <- conc*r
S <- rnorm(n,0,0.3) %*% t(rep(1,p))
B <- matrix(rnorm(n*p,0,0.8),n,p)
R <- rep(1,n) %*% t(r)
M <- matrix(rnorm(n*p,0,0.021),n,p)
# Fifth observation is an outlier
M[5,] <- M[5,]*3 + sample(c(0.5,-0.5),replace=TRUE,p)
C <- rep(1,n) %*% t(conc)
C[1:20,c(2,15,28,40)] <- C[1:20,c(2,15,28,40)]+matrix(1,20,4)*1.8
X <- (1-S)*(C*R+B)*exp(M)
# Biomarker identification
b <- biomarker(X, g1 = 1:20, g2 = 21:40, type = "tau")

biplot.factanal  Biplot method

Description

Provides robust compositional biplots.

Usage

## S3 method for class 'factanal'
biplot(x, ...)

Arguments

x
  object of class ‘factanal’
...

Details

The robust compositional biplot according to Aitchison and Greenacre (2002), computed from resulting (robust) loadings and scores, is performed.

Value

The robust compositional biplot.

Author(s)

M. Templ, K. Hron
Reference


See Also

`pfa`

Examples

```r
data(expenditures)
res.rob <- pfa(expenditures, factors=2, scores = "regression")
biplot(res.rob)
```

### Description

Provides robust compositional biplots.

### Usage

```r
## S3 method for class 'pcaCoDa'
biplot(x, y, ...)
```

### Arguments

- `x` object of class `pcaCoDa`
- `y` ...
- `...` arguments passed to plot methods

### Details

The robust compositional biplot according to Aitchison and Greenacre (2002), computed from (robust) loadings and scores resulting from `pcaCoDa`, is performed.

### Value

The robust compositional biplot.

### Author(s)

M. Templ, K. Hron
References


See Also

cCoDa, plot.pcaCoDa

Examples

data(coffee)
p1 <- pcaCoDa(coffee[, -1])
p1
biplot(p1)

## with labels for the scores:
data(arcticLake)
rownames(arcticLake) <- paste(sample(letters[1:26], nrow(arcticLake), replace=TRUE),
1:nrow(arcticLake), sep="")
pc <- pcaCoDa(arcticLake, method="classical")
biplot(pc, xlab=rownames(arcticLake))

——

**bootnComp**

*Bootstrap to find optimal number of components*

Description

Combined bootstrap and cross validation procedure to find optimal number of PLS components

Usage

`bootnComp(X, y, R = 99, plotting = FALSE)`

Arguments

- `X` predictors as a matrix
- `y` response
- `R` number of bootstrap replicates
- `plotting` if TRUE, a diagnostic plot is drawn for each bootstrap replicate
cancer

Details

Heavily used internally in function impRZilr.

Value

Including other information in a list, the optimal number of components

Author(s)

Matthias Templ

See Also

impRZilr

Examples

```r
## we refer to impRZilr()
```

cancer

hospital discharges on cancer and distribution of age

Description

Hospital discharges of in-patients on neoplasms (cancer) per 100,000 inhabitants (year 2007) and population age structure.

Format

A data set on 24 compositions on 6 variables.

Details

- country
country
- year
year
- p1 percentage of population with age below 15
- p2 percentage of population with age between 15 and 60
- p3 percentage of population with age above 60
- discharges hospital discharges of in-patients on neoplasms (cancer) per 100,000 inhabitants

The response (discharges) is provided for the European Union countries (except Greece, Hungary and Malta) by Eurostat. As explanatory variables we use the age structure of the population in the same countries (year 2008). The age structure consists of three parts, age smaller than 15, age between 15 and 60 and age above 60 years, and they are expressed as percentages on the overall population in the countries. The data are provided by the United Nations Statistics Division.
### Author(s)

conversion to R by Karel Hron and Matthias Templ <matthias.tplm@tuwien.ac.at>

### Source


### References


### Examples

```r
data(cancer)
str(cancer)
```

### Description

Two main types of malignant neoplasms cancer affecting colon and lung, respectively, in male and female populations. For this purpose population data (2012) from 35 OECD countries were collected.

### Format

A data set on 35 compositional tables on 4 parts (row-wise sorted cells) and 5 variables.

### Details

- `cancerMN`
  - `malignant neoplasms cancer`

### Author(s)

conversion to R by Karel Hron and integration by Matthias Templ <matthias.tplm@tuwien.ac.at>

### Source

http://www.oecd.org
**ced**

Examples

```r
data(cancerMN)
head(cancerMN)
rowSums(cancerMN[, 2:5])
```

---

**ced**  
*compositional error deviation*

**Description**

Normalized Aitchison distance between two data sets

**Usage**

```r
ced(x, y, ni)
```

**Arguments**

- `x`: matrix or data frame
- `y`: matrix or data frame of the same size as `x`
- `ni`: normalization parameter. See details below.

**Details**

This function has been mainly written for procedures that evaluate imputation or replacement of rounded zeros. The `ni` parameter can thus, e.g. be used for expressing the number of rounded zeros.

**Value**

The compositional error distance

**Author(s)**

Matthias Templ

**References**


**See Also**

rdcm
Examples

data(expenditures)
x <- expenditures
x[1,3] <- NA
xi <- impKNNa(x)$ximp
ced(expenditures, xi, ni = sum(is.na(x))

---

cenLR

Centred log-ratio transformation

Description

The cenLR transformation moves D-part compositional data from the simplex into a D-dimensional real space.

Usage

cenLR(x, base = exp(1))

Arguments

x multivariate data ideally of class data.frame or matrix
base a positive or complex number: the base with respect to which logarithms are computed. Defaults to exp(1).

Details

Each composition is divided by the geometric mean of its parts before the logarithm is taken.

Value

The transformed data, including

x clr clr transformed data
gm the geometric means of the original composition.

Note

The resulting transformed data set is singular by definition.

Author(s)

Matthias Templ

References

**cenLRinv**

See Also

cenLRinv, addLR, pivotCoord, addLRinv, pivotCoordInv

Examples

data(expenditures)
eclr <- cenLR(expenditures)
inv eclr <- cenLRinv(eclr)
head(expenditures)
head(inv eclr)
head(pivotCoordInv(eclr$x$ clr))

---

**cenLRinv**  
*Inverse centred log-ratio transformation*

Description

Applies the inverse centred log-ratio transformation.

Usage

cenLRinv(x, useClassInfo = TRUE)

Arguments

x  
an object of class “clr”, “data.frame” or “matrix”

useClassInfo  
if the object is of class “clr”, the useClassInfo is used to determine if the class information should be used. If yes, also absolute values may be preserved.

Value

the transformed data set.

Author(s)

Matthias Templ

References


See Also

cenLR, addLR, pivotCoord, addLRinv, pivotCoordInv
**Examples**

```r
data(expenditures)
eclr <- cenLR(expenditures, 2)
inveclr <- cenLRinv(eclr)
head(expenditures)
head(inveclr)
head(cenLRinv(eclr$x.clr))
```

---

**chorizonDL**  
*C-horizon of the Kola data with rounded zeros*

---

**Description**

This data set is almost the same as `chorizon` data set in package `mvoutlier` and `chorizonDL`, except that values below the detection limit are coded as zeros, and detection limits provided as attributes to the data set and less variables are included.

**Format**

A data frame with 606 observations on the following 62 variables.

*ID*  a numeric vector  
*XCOO* a numeric vector  
*YCOO* a numeric vector  
*Ag* concentration in mg/kg  
*Al* concentration in mg/kg  
*Al_XRF* concentration in wt. percentage  
*As* concentration in mg/kg  
*Ba* concentration in mg/kg  
*Ba_INAA* concentration in mg/kg  
*Be* concentration in mg/kg  
*Bi* concentration in mg/kg  
*Ca* concentration in mg/kg  
*Ca_XRF* concentration in wt. percentage  
*Cd* concentration in mg/kg  
*Ce_INAA* concentration in mg/kg  
*Co* concentration in mg/kg  
*Co_INAA* concentration in mg/kg  
*Cr* concentration in mg/kg  
*Cr_INAA* concentration in mg/kg
Cu concentration in mg/kg
Eu_INAA concentration in mg/kg
Fe concentration in mg/kg
Fe_XRF concentration in wt. percentage
HF_INAA concentration in mg/kg
K concentration in mg/kg
K_XRF concentration in wt. percentage
La concentration in mg/kg
La_INAA concentration in mg/kg
Li concentration in mg/kg
Lu_INAA concentration in mg/kg
Mg concentration in mg/kg
Mg_XRF concentration in wt. percentage
Mn concentration in mg/kg
Mn_XRF concentration in wt. percentage
Na concentration in mg/kg
Na_XRF concentration in wt. percentage
Nd_INAA concentration in mg/kg
Ni concentration in mg/kg
P concentration in mg/kg
P_XRF concentration in wt. percentage
Pb concentration in mg/kg
S concentration in mg/kg
Sc concentration in mg/kg
Sc_INAA concentration in mg/kg
Si concentration in mg/kg
Si_XRF concentration in wt. percentage
Sm_INAA concentration in mg/kg
Sr concentration in mg/kg
Th_INAA concentration in mg/kg
Ti concentration in mg/kg
Ti_XRF concentration in wt. percentage
V concentration in mg/kg
Y concentration in mg/kg
Yb_INAA concentration in mg/kg
Zn concentration in mg/kg
LOI concentration in wt. percentage
**pH**  ph value  
**ELEV**  elevation  
**COUN**  country  
**ASP**  a numeric vector  
**TOPC**  a numeric vector  
**LITO**  information on lithography

**Note**

For a more detailed description of this data set, see `chorizon` in package `mvoutlier`.

**Source**


**References**


**See Also**

`chorizon`, `chorizondl`

**Examples**

```r
data(chorizondl, package = "robCompositions")
dim(chorizondl)
colnames(chorizondl)
zeroPatterns(chorizondl)
```

---

**clustCoDa**  
*Cluster analysis for compositional data*

**Description**

Clustering in orthonormal coordinates or by using the Aitchison distance

**Usage**

```r
clustCoDa(x, k = NULL, method = "Mclust", scale = "robust",
  transformation = "pivotCoord", distMethod = NULL, iter.max = 100,
  vals = TRUE, alt = NULL, bic = NULL, verbose = TRUE)
```

```r
# S3 method for class 'clustCoDa'
plot(x, y, ..., normalized = FALSE,
  which.plot = "clusterMeans", measure = "silwidths")
```
Arguments

- **x**: compositional data represented as a data.frame
- **k**: number of clusters
- **method**: clustering method. One of Mclust, cmeans, kmeansHartigan, cmeansUfcl, pam, clara, fanny, ward.D2, single, hclustComplete, average, mcquitty, median, centroid
- **scale**: If orthonormal coordinates should be normalized.
- **transformation**: default is the isometric log-ratio transformation. Can only used when distMethod is not Aitchison.
- **distMethod**: Distance measure to be used. If “Aitchison”, then transformation should be “identity”.
- **iter.max**: parameter if kmeans is chosen. The maximum number of iterations allowed
- **vals**: if cluster validity measures should be calculated
- **alt**: a known partitioning can be provided (for special cluster validity measures)
- **bic**: if TRUE then the BIC criteria is evaluated for each single cluster as validity measure
- **verbose**: if TRUE additional print output is provided
- **y**: the y coordinates of points in the plot, optional if x is an appropriate structure.
- **...**: additional parameters for print method passed through
- **normalized**: results gets normalized before plotting. Normalization is done by z-transformation applied on each variable.
- **which.plot**: currently the only plot. Plot of cluster centers.
- **measure**: cluster validity measure to be considered for which.plot equals “partMeans”

Details

The compositional data set is either internally represented by orthonormal coordiantes before a cluster algorithm is applied, or - depending on the choice of parameters - the Aitchison distance is used.

Value

all relevant information such as cluster centers, cluster memberships, and cluster statistics.

Author(s)

Matthias Templ (accessing the basic features of hclust, Mclust, kmeans, etc. that are all written by others)

References


clustCoDa_qmode

Q-mode cluster analysis for compositional parts

Description
Clustering in using the variation matrix of compositional parts

Usage
clustCoDa_qmode(x, method = "ward.D2")

Arguments
x    compositional data represented as a data.frame
method  hclust method

Value
a hclust object

Author(s)
Matthias Templ (accessing the basic features of hclust that are all written by other authors)

References

Examples
data(expenditures)
x <- expenditures
c1 <- clustCoDa_qmode(x)
plot(c1)
c2 <- clustCoDa_qmode(x, method = "single")
plot(c2)
coffee data set

Description

30 commercially available coffee samples of different origins.

Usage

data(coffee)

Format

A data frame with 30 observations and 7 variables.

Details

- `sort` sort of coffee
- `acit` acetic acid
- `metpyr` methylpyrazine
- `furfu` furfural
- `furfualc` furfuryl alcohol
- `dimeth` 2,6 dimethylpyrazine
- `metU` 5-methylfurfural

In the original data set, 15 volatile compounds (descriptors of coffee aroma) were selected for a statistical analysis. We selected six compounds (compositional parts) on three sorts of coffee.

Author(s)

Matthias Templ <matthias.templ@tuwien.ac.at>, Karel Hron

References


Examples

data(coffee)
str(coffee)
summary(coffee)
compareMahal  

*Compares Mahalanobis distances from two approaches*

**Description**

Mahalanobis distances are calculated for each zero pattern. Two approaches are used. The first one estimates Mahalanobis distance for observations belonging to one each zero pattern each. The second method uses a more sophisticated approach described below.

**Usage**

```r
compareMahal(x, imp = "KNNa")
```

```r
# S3 method for class 'mahal'
plot(x, y, ...)
```

**Arguments**

- `x` data frame or matrix
- `imp` imputation method
- `y` unused second argument for the plot method
- `...` additional arguments for plotting passed through

**Value**

- `df` a data.frame containing the Mahalanobis distances from the estimation in sub-groups, the Mahalanobis distances from the imputation and covariance approach, an indicator specifying outliers and an indicator specifying the zero pattern
- `df2` a groupwise statistics.

**Author(s)**

Matthias Templ, Karel Hron

**References**


**See Also**

`impKNNa`, `pivotCoord`
constSum

Examples

data(arcticLake)
# generate some zeros
arcticLake[1:10, 1] <- 0
arcticLake[11:20, 2] <- 0
m <- compareMahal(arcticLake)
plot(m)

constSum(x, const = 1, na.rm = TRUE)

Arguments

x multivariate data ideally of class data.frame or matrix
const constant, the default equals 1.
na.rm removing missing values.

Value

The data for which the row sums are equal to const.

Author(s)

Matthias Templ

Examples

data(expenditures)
constSum(expenditures)
constSum(expenditures, 100)
Coordinate representation of CoDa tables

Description
General approach to orthonormal coordinates for compositional tables

Usage
coord(x, SBPr, SBPc)

## S3 method for class 'coord'
print(x, ...)

Arguments
- x: an object of class “table”, “data.frame” or “matrix”
- SBPr: sequential binary partition for rows
- SBPc: sequential binary partition for columns
- ...: further arguments passed to the print function

Details
A contingency or probability table is by definition a compositional data set. This approach considers the related special properties of compositional tables. It constructs orthonormal coordinates for compositional tables using the isometric log-ratio approach for given sequential binary partitions on rows and columns.

Value
Row and column balances, odds ratios, particularly
- row_balances: row balances
- row_bin: binary partition for rows
- col_balances: column balances
- col_bin: binary partition for columns
- odds_ratios_coord: odds ratio coordinates

Author(s)
Kamila Facevicova, and minor adaption by Matthias Templ

References
Kamila Facevicova, Karel Hron, Valentin Todorov, Matthias Templ (201x) General approach to coordinate representation of compositional tables. Submitted to JRSS B.
Examples

```r
x <- rbind(c(1,5,3,6,8,4),c(6,4,9,5,8,12),c(15,2,68,42,11,6),
           c(20,15,4,6,23,8),c(11,20,35,26,44,8))
x
SBPc <- rbind(c(1,1,1,1,-1,-1),c(1,-1,-1,1,0,0),c(0,1,1,-1,0,0),
              c(0,1,-1,0,0,0),c(0,0,0,0,1,-1))
SBPc
SBPr <- rbind(c(1,1,1,-1,-1),c(1,1,-1,0,0),c(1,-1,0,0,0),c(0,0,0,1,-1))
SBPr
result <- coord(x, SBPr,SBPc)
result
data(socomexp)
```

Correlations for compositional data

Description

This function computes correlation coefficients between compositional parts based on symmetric balances.

Usage

```r
corCoDa(x, ...)
```

Arguments

- `x` a matrix or data frame with compositional data
- `...` additional arguments for the function `cor`

Value

A compositional correlation matrix.

Author(s)

Petra Kynclova

References

Kynclova, P., Hron, K., Filzmoser, P. Correlation between compositional parts based on symmetric balances. Submitted to *Mathematical Geosciences*. 
Examples

data(expenditures)
corCoDa(expenditures)
x <- arcticLake
corCoDa(x)

dacoDa

Linear and quadratic discriminant analysis for compositional data.

Description

Linear and quadratic discriminant analysis for compositional data using either robust or classical estimation.

Usage

dacoDa(x, grp = TRUE, method = "classical", rule = "linear", ...)

Arguments

x a matrix or data frame containing the explanatory variables
grp grouping variable: a factor specifying the class for each observation.
coda TRUE, when the underlying data are compositions.
method "classical" or "robust"
rule a character, either “linear” (the default) or “quadratic”.
... additional arguments for the functions passed through

Details

An ilr-transformation is applied to compositional data (if coda==TRUE). For linear discriminant analysis the functions LdaClassic (classical) and Linda (robust) from the package rrcov are used. Similarly, quadratic discriminant analysis uses the functions QdaClassic and QdaCov (robust) from the same package.

The classical linear and quadratic discriminant rules are invariant to ilr and clr transformations. The robust rules are invariant to ilr transformations if affine equivariant robust estimators of location and covariance are taken.

Value

An S4 object of class LdaClassic, Linda, QdaClassic or QdaCov. See package rrcov for details.

Author(s)

Jutta Gamper
References


See Also

`LdaClassic`, `Linda`, `QdaClassic`, `QdaCov`

Examples

```r
## toy data (non-compositional)
require(MASS)
x1 <- mvrnorm(20,c(0,0,0),diag(3))
x2 <- mvrnorm(30,c(3,0,0),diag(3))
x3 <- mvrnorm(40,c(0,3,0),diag(3))
X <- rbind(x1,x2,x3)
grp=c(rep(1,20),rep(2,30),rep(3,40))

clas1 <- daCoDa(X, grp, codalogical, method = "classical", rule="linear")
summary(clas1)
## predict runs only with newest version of rrcov
## Not run:
predict(clas1)

## End(Not run)
# specify different prior probabilities
clas2 <- daCoDa(X, grp, codalogical, prior=c(1/3, 1/3, 1/3))
summary(clas2)

## compositional data
data(coffee)
x <- coffee[coffee$sort!="robusta",2:7]
group <- droplevels(coffee$sort[coffee$sort!="robusta"])
cof.cl <- daCoDa(x, group, method="classical", rule="quadratic")
cof.rob <- daCoDa(x, group, method="robust", rule="quadratic")
## predict runs only with newest version of rrcov
## Not run:
predict(cof.cl)@ct
predict(cof.rob)@ct

## End(Not run)
```

---

**daFisher**

*Discriminant analysis by Fisher Rule.*

**Description**

Discriminant analysis by Fishers rule using CoDa methods.
Usage

daFisher(x, grp, coda = TRUE, method = "classical",
plotScore = FALSE, ...)

## S3 method for class 'daFisher'
print(x, ...)

## S3 method for class 'daFisher'
predict(object, ..., newdata)

## S3 method for class 'daFisher'
summary(object, ...)

Arguments

x       a matrix or data frame containing the explanatory variables (training set)
grp      grouping variable: a factor specifying the class for each observation.
coda     TRUE, when the underlying data are compositions.
method   "classical" or "robust" estimation.
plotScore TRUE, if the scores should be plotted automatically.
...      additional arguments for the print method passed through
object   object of class "daFisher"
newdata  new data in the appropriate form (CoDa, etc)

Details

The Fisher rule leads only to linear boundaries. However, this method allows for dimension reduction and thus for a better visualization of the separation boundaries. For the Fisher discriminant rule (Fisher, 1938; Rao, 1948) the assumption of normal distribution of the groups is not explicitly required, although the method looses its optimality in case of deviations from normality.

The classical Fisher discriminant rule is invariant to ilr and clr transformations. The robust rule is invariant to ilr transformations if affine equivariant robust estimators of location and covariance are taken.

Robustification is done (method “robust”) by estimating the columnwise means and the covariance by the Minimum Covariance Estimator.

Value

an object of class “daFisher” including the following elements

B        Between variance of the groups
W        Within variance of the groups
loadings loadings
scores    fisher scores
mc        table indicating misclassifications
mcrate  misclassification rate
coda    coda
grp     grouping
grppred predicted groups
xc      xc
meanj   meanj
cv      cv
pj      pj
meanov  meanov
fdiscr  fdiscr

**Author(s)**

Peter Filzmoser, Matthias Templ.

**References**


**See Also**

Linda

**Examples**

```r
## toy data (non-compositional)
require(MASS)
x1 <- mvrnorm(20,c(0,0,0),diag(3))
x2 <- mvrnorm(30,c(3,0,0),diag(3))
x3 <- mvrnorm(40,c(0,3,0),diag(3))
X <- rbind(x1,x2,x3)
grp=c(rep(1,20),rep(2,30),rep(3,40))

#par(mfrow=c(1,2))
d1 <- daFisher(X,grp.grp,method="classical",coda=FALSE)
d2 <- daFisher(X,grp.grp,method="robust",coda=FALSE)
d2
summary(d2)
predict(d2, newdata = X)

## example with olive data:
## Not run:
```
data(olive, package = "RnavGraph")
# exclude zeros (alternatively impute them if
# the detection limit is known using impRZilr())
ind <- which(olive == 0, arr.ind = TRUE)[,1]
olives <- olive[-ind, ,]
x <- olives[, 4:10]
grp <- olives$Region # 3 groups
res <- daFisher(x, grp)
res
summary(res)
res <- daFisher(x, grp, plotScore = TRUE)
res <- daFisher(x, grp, method = "robust")
res
summary(res)
predict(res, newdata = x)
res <- daFisher(x, grp, plotScore = TRUE, method = "robust")

# 9 regions
grp <- olives$Area
res <- daFisher(x, grp, plotScore = TRUE)
res
summary(res)
predict(res, newdata = x)

## End(Not run)

economy

---

economic indicators

description
Household and government consumptions, gross capital formation and import and exports of goods
and services.

usage
data(economy)

format
A data frame with 30 observations and 7 variables

details
- country country name
- country2 country name, short version
- HHconsumption Household and NPISH final consumption expenditure
- GOVconsumption Final consumption expenditure of general government
• capital  Gross capital formation
• exports  Exports of goods and services
• imports  Imports of goods and services

Author(s)
Peter Filzmoser, Matthias Templ <matthias.temp@tuwien.ac.at>

References

Examples

data(economy)
str(economy)

<table>
<thead>
<tr>
<th>educFM</th>
<th>education level of father (F) and mother (M)</th>
</tr>
</thead>
</table>

Description
Education level of father (F) and mother (M) in percentages of low (l), medium (m), and high (h) of 31 countries in Europe.

Usage
data(educFM)

Format
A data frame with 31 observations and 8 variables

Details
• country  community code
• F.l  percentage of females with low education level
• F.m  percentage of females with medium education level
• F.h  percentage of females with high education level
• F.l  percentage of males with low education level
• F.m  percentage of males with medium education level
• F.h  percentage of males with high education level
Author(s)

Peter Filzmoser, Matthias Templ

Source


Examples

data(educFM)
str(educFM)

Description

Results of an election in Germany 2013 in different federal states

Usage

data(election)

Format

A data frame with 16 observations and 8 variables

Details

Votes for the political parties in the elections (compositional variables), and their relation to the unemployment rate and the average monthly income (external non-compositional variables). Votes are for the Christian Democratic Union and Christian Social Union of Bavaria, also called The Union (CDU/CSU), Social Democratic Party (SDP), The Left (DIE LINKE), Alliance '90/The Greens (GRÜNE), Free Democratic Party (FDP) and the rest of the parties participated in the elections (other parties). The votes are examined in absolute values (number of valid votes). The unemployment in the federal states is reported in percentages, and the average monthly income in Euros.

- **CDU_CSU** Christian Democratic Union and Christian Social Union of Bavaria, also called The Union
- **SDP** Social Democratic Party
- **GRUENE** Alliance '90/The Greens
- **FDP** Free Democratic Party
- **DIE_LINKE** The Left
- **other_parties** Votes for the rest of the parties participated in the elections
- **unemployment** Unemployment in the federal states in percentages
- **income** Average monthly income in Euros
### Author(s)
Petra Klynclova, Matthias Templ

### Source
German Federal Statistical Office

### References

### Examples
```r
data(election)
str(election)
```

### Description
Results the Austrian presidential election in October 2016.

### Usage
data(electionATbp)

### Format
A data frame with 2202 observations and 10 variables

### Details
Votes for the candidates Hofer and Van der Bellen.

- **GKZ** Community code
- **Name** Name of the community
- **Eligible** eligible votes
- **Votes_total** total votes
- **Votes_invalid** invalid votes
- **Votes_valid** valid votes
- **Hofer_total** votes for Hofer
- **Hofer_perc** votes for Hofer in percentages
- **VanderBellen_total** votes for Van der Bellen
- **VanderBellen_perc** votes for Van der Bellen in percentages
Author(s)
Peter Filzmoser

Source
OpenData Austria, https://www.data.gv.at/

Examples

data(electionATbp)
str(electionATbp)

employment
employment in different countries by gender and status.

Description
employment in different countries by gender and status.

Usage
data(employment)

Format
A three-dimensional table

Examples
data(employment)
str(employment)
employment

employment_df
Employment in different countries by gender and status.

Description
• genderfactor
• statusfactor, defining if part or full time work
• countrycountry
• valueemployment
expenditures

Usage

```r
data(expenditures)
```

Format

A data frame with 20 observations on the following 5 variables.

Details

- housing  housing (including fuel and light)
- foodstuffs  foodstuffs
- alcohol  alcohol and tobacco
- other  other goods (including clothing, footwear and durable goods)
- services  services (including transport and vehicles)

This data set contains household expenditures on five commodity groups of 20 single men. The variables represent housing (including fuel and light), foodstuff, alcohol and tobacco, other goods (including clothing, footwear and durable goods) and services (including transport and vehicles). Thus they represent the ratios of the men’s income spent on the mentioned expenditures.

Author(s)

Matthias Templ <matthiastempl@tuwien.ac.at>, Karel Hron
References


Examples

```r
data(expenditures)
## imputing a missing value in the data set using k-nearest neighbor imputation:
expenditures[1,3] <- NA
impKNNa(expenditures)$xImp[1,3]
```

**Description**

Mean consumption expenditure of households at EU-level. The final consumption expenditure of households encompasses all domestic costs (by residents and non-residents) for individual needs.

**Format**

A data frame with 27 observations on the following 12 variables.

- `food` numeric vector
- `alcohol` numeric vector
- `clothing` numeric vector
- `housing` numeric vector
- `furnishings` numeric vector
- `health` numeric vector
- `transport` numeric vector
- `communications` numeric vector
- `recreation` numeric vector
- `education` numeric vector
- `restaurants` numeric vector
- `other` numeric vector

**Source**

Eurostat

**Examples**

```r
data(expendituresEU)
```
GDP satisfaction

Description

Satisfaction of GDP in 31 countries. The GDP is measured per capita from the year 2012.

Usage

data(GDPsatis)

Format

A data frame with 31 observations and 8 variables

Details

- country  community code
- gdp  GDP per capita in 2012
- very.bad  satisfaction very bad
- bad  satisfaction bad
- moderately.bad  satisfaction moderately bad
- moderately.good  satisfaction moderately good
- good  satisfaction good
- very.good  satisfaction very good

Author(s)

Peter Filzmoser, Matthias Templ

Source


Examples

data(GDPsatis)
str(GDPsatis)
GEMAS geochemical data set

Description

Geochemical data set on agricultural and grazing land soil

Usage

data(gemas)

Format

A data frame with 2108 observations and 30 variables

Details

- country name
- longitude latitude in WGS84
- Xcoord UTM zone east
- Ycoord UTM zone north
- MeanTempAnnual mean temperature
- AnnPrec Annual mean precipitation
- soilclass soil class
- sand sand
- silt silt
- clay clay
- Al Concentration of aluminum (in mg/kg)
- Ba Concentration of barium (in mg/kg)
- Ca Concentration of calcium (in mg/kg)
- Cr Concentration of chromium (in mg/kg)
- Fe Concentration of iron (in mg/kg)
- K Concentration of potassium (in mg/kg)
- Mg Concentration of magnesium (in mg/kg)
- Mn Concentration of manganese (in mg/kg)
- Na Concentration of sodium (in mg/kg)
- Nb Concentration of niobium (in mg/kg)
- Ni Concentration of nickel (in mg/kg)
- P Concentration of phosphorus (in mg/kg)
• Si  Concentration of silicium (in mg/kg)
• Sr  Concentration of strontium (in mg/kg)
• Ti  Concentration of titanium (in mg/kg)
• V   Concentration of vanadium (in mg/kg)
• Y   Concentration of yttrium (in mg/kg)
• Zn  Concentration of zinc (in mg/kg)
• Zr  Concentration of zirconium (in mg/kg)
• L0I Loss on ignition (in wt-percent)

The sampling, at a density of 1 site/2500 sq. km, was completed at the beginning of 2009 by collecting 2211 samples of agricultural soil (Ap-horizon, 0-20 cm, regularly ploughed fields), and 2118 samples from land under permanent grass cover (grazing land soil, 0-10 cm), according to an agreed field protocol. All GEMAS project samples were shipped to Slovakia for sample preparation, where they were air dried, sieved to <2 mm using a nylon screen, homogenised and split to subsamples for analysis. They were analysed for a large number of chemical elements. In this sample, the main elements by X-ray fluorescence are included as well as the composition on sand, silt, clay.

Author(s)

GEMAS is a cooperation project between the EuroGeoSurveys Geochemistry Expert Group and Eurometaux. Integration in R, Peter Filzmoser and Matthias Templ.

References


Examples

data(gemas)
str(gemas)
## sample sites
## Not run:
require(ggmap)
map <- get_map("europe", source = "stamen", maptype = "watercolor", zoom=4)
ggmap(map) + geom_point(aes(x=longitude, y=latitude), data=gemas)
map <- get_map("europe", zoom=4)
ggmap(map) + geom_point(aes(x=longitude, y=latitude), data=gemas, size=0.8)

## End(Not run)
Description

Gjovik geochemical data set

Format

A data frame with 615 observations and 63 variables.

- ID  a numeric vector
- MAT  type of material
- mE32wgs longitude
- mN32wgs latitude
- XC00 X coordinates
- YC00 Y coordinates
- ALT altitude
- kmNS some distance north-south
- kmSN some distance south-north
- LITHO lithologies
- Ag  a numeric vector
- Al  a numeric vector
- As  a numeric vector
- Au  a numeric vector
- B   a numeric vector
- Ba  a numeric vector
- Be  a numeric vector
- Bi  a numeric vector
- Ca  a numeric vector
- Cd  a numeric vector
- Ce  a numeric vector
- Co  a numeric vector
- Cr  a numeric vector
- Cs  a numeric vector
- Cu  a numeric vector
- Fe  a numeric vector
- Ga  a numeric vector
- Ge  a numeric vector
• Hf  a numeric vector
• Hg  a numeric vector
• In  a numeric vector
• K   a numeric vector
• La  a numeric vector
• Li  a numeric vector
• Mg  a numeric vector
• Mn  a numeric vector
• Mo  a numeric vector
• Na  a numeric vector
• Nb  a numeric vector
• Ni  a numeric vector
• P   a numeric vector
• Pb  a numeric vector
• Pd  a numeric vector
• Pt  a numeric vector
• Rb  a numeric vector
• Re  a numeric vector
• S   a numeric vector
• Sb  a numeric vector
• Sc  a numeric vector
• Se  a numeric vector
• Sn  a numeric vector
• Sr  a numeric vector
• Ta  a numeric vector
• Te  a numeric vector
• Th  a numeric vector
• Ti  a numeric vector
• Tl  a numeric vector
• U   a numeric vector
• V   a numeric vector
• W   a numeric vector
• Y   a numeric vector
• Zn  a numeric vector
• Zr  a numeric vector
Details
Geochemical data set. 41 sample sites have been investigated. At each site, 15 different sample materials have been collected and analyzed for the concentration of more than 40 chemical elements. Soil: CHO - C horizon, OHO - O horizon. Mushroom: LAC - milkcap. Plant: BIL - birch leaves, BLE - blueberry leaves, BLU - blueberry twigs, BTW - birch twigs, CLE - cowberry leaves, COW - cowberry twigs, EQU - horsetail, FER - fern, HYL - terrestrial moss, PIB - pine bark, SNE - spruce needles, SPR - spruce twigs.

Author(s)
Peter Filzmoser, Dominika Miksova

References

Examples

data(gjovik)
str(gjovik)

<table>
<thead>
<tr>
<th>gm</th>
<th>gmean</th>
</tr>
</thead>
</table>

Description
This function calculates the geometric mean.

Usage
gm(x)

Arguments
x a vector

Details
gm calculates the geometric mean for all positive entries of a vector. Please note that there is a faster version available implemented with Rcpp but it currently do not pass CRAN checks cause of use of Rcpp11 features. This C++ version accounts for over- and underflows. It is placed in inst/doc.

Author(s)
Matthias Templ
Examples

```r
gm(c(3, 5, 3, 6, 7))
```

---

**Description**

Computes the geometric mean(s) of a numeric vector, matrix or data.frame

**Usage**

```r
function(x, margin = NULL)
```

**Arguments**

- `x` : matrix or data.frame with numeric entries
- `margin` : a vector giving the subscripts which the function will be applied over, 1 indicates rows, 2 indicates columns, 3 indicates all values.

**Details**

`gmean_sum` calculates the totals based on geometric means while `gmean` calculates geometric means on rows (margin = 1), on columns (margin = 2), or on all values (margin = 3)

**Value**

- geometric means (if `gmean` is used) or totals (if `gmean_sum` is used)

**Author(s)**

Matthias Templ

**Examples**

```r
data("precipitation")
gmean_sum(precipitation)
gmean_sum(precipitation, margin = 2)
gmean_sum(precipitation, margin = 1)
gmean_sum(precipitation, margin = 3)
addmargins(precipitation)
addmargins(precipitation, FUN = gmean_sum)
addmargins(precipitation, FUN = mean)
addmargins(precipitation, FUN = gmean)
data("arcticLake", package = "robCompositions")
```
govexp
government spending

gmean(arcticLake$sand)
gmean(as.numeric(arcticLake[, ]))
gmean(arcticLake)
gmean(arcticLake, margin = 1)
gmean(arcticLake, margin = 2)
gmean(arcticLake, margin = 3)

Description

Government expenditures based on COFOG categories

Format

A (tidy) data frame with 5140 observations on the following 4 variables.

- **country**  Country of origin
- **category** The COFOG expenditures are divided into the following ten categories: general public services; defence; public order and safety; economic affairs; environmental protection; housing and community amenities; health; recreation, culture and religion; education; and social protection.
- **year**  Year
- **value**  COFOG spendings/expenditures

Details

The general government sector consists of central, state and local governments, and the social security funds controlled by these units. The data are based on the system of national accounts, a set of internationally agreed concepts, definitions, classifications and rules for national accounting. The classification of functions of government (COFOG) is used as classification system. The central government spending by category is measured as a percentage of total expenditures.

Author(s)

translated from [https://data.oecd.org/](https://data.oecd.org/) and restructured by Matthias Templ

Source

OECD: [https://data.oecd.org/](https://data.oecd.org/)

Examples

data(govexp)
str(govexp)
### haplogroups

<table>
<thead>
<tr>
<th>haplogroups</th>
<th>haplogroups data.</th>
</tr>
</thead>
</table>

**Description**

Distribution of European Y-chromosome DNA (Y-DNA) haplogroups by region in percentage.

**Format**

A data frame with 38 observations on the following 12 variables.

- **I1** pre-Germanic (Nordic)
- **I2b** pre-Celto-Germanic
- **I2a1** Sardinian, Basque
- **I2a2** Dinaric, Danubian
- **N1c1** Uralo-Finnic, Baltic, Siberian
- **R1a** Balto-Slavic, Mycenaean Greek, Macedonia
- **R1b** Italic, Celtic, Germanic; Hitite, Armenian
- **G2a** Caucasian, Greco-Anatolien
- **E1b1b** North and Eastern Afrika, Near Eastern, Balkanic
- **J2** Mesopotamian, Minoan Greek, Phoenician
- **J1** Semitic (Arabic, Jewish)
- **T** Near-Eastern, Egyptian, Ethiopian, Arabic

**Details**

Human Y-chromosome DNA can be divided in genealogical groups sharing a common ancestor, called haplogroups.

**Source**


**Examples**

```r
data(haplogroups)
```
ilr.2x2

ilr coordinates in 2x2 tables

Description

ilr coordinates of original, independent and interaction table using SBP1 and SBP2

Usage

ilr.2x2(x, margin = 1, type = "independence", version = "book")

Arguments

x

a 2x2 table

margin

for 2x2 tables available for a whole set of another dimension. For example, if 2x2 tables are available for every country.

type

choose between “independence” or “interaction” table

version

the version used in the “paper” below or the version of the upcoming “book”.

Value

The ilr coordinates

Author(s)

Kamila Facevicova, Matthias Templ

References


Examples

data(employment)

ilr.2x2(employment[,"AUT"])

ilr.2x2(employment[,"AUT"], version = "paper")

ilr.2x2(employment, margin = 3, version = "paper")

ilr.2x2(employment[,"AUT"], type = "interaction")
Description

Parametric replacement of rounded zeros and missing values for compositional data using classical and robust methods based on ilr-transformations with special choice of balances. Values under detection limit should be saved with the negative value of the detection limit (per variable). Missing values should be coded as NA.

Usage

impAll(x)

Arguments

x  data frame

Details

This is a wrapper function that calls impRZilr() for the replacement of zeros and impCoda for the imputation of missing values sequentially. The detection limit is automatically derived from negative numbers in the data set.

Value

The imputed data set.

Note

This function is mainly used by the compositionsGUI.

References


See Also

impCoda, impRZilr
ImpCoda

Imputation of missing values in compositional data

Description

This function offers different methods for the imputation of missing values in compositional data. Missing values are initialized with proper values. Then iterative algorithms try to find better estimations for the former missing values.

Usage

impCoda(x, maxit = 10, eps = 0.5, method = "ltsReg",
        closed = FALSE, init = "KNN", k = 5, dl = rep(0.05, ncol(x)),
        noise = 0.1, bruteforce = FALSE)

Arguments

x data frame or matrix
maxit maximum number of iterations
eps convergence criteria
method imputation method
closed imputation of transformed data (using ilr transformation) or in the original space (closed equals TRUE)
init method for initializing missing values
k number of nearest neighbors (if init $==$ "KNN")
dl detection limit(s), only important for the imputation of rounded zeros
noise amount of adding random noise to predictors after convergency
bruteforce if TRUE, imputations over dl are set to dl. If FALSE, truncated (Tobit) regression is applied.

Details

eps: The algorithm is finished as soon as the imputed values stabilize, i.e. until the sum of Aitchison distances from the present and previous iteration changes only marginally (eps).
method: Several different methods can be chosen, such as ‘ltsReg’: least trimmed squares regression is used within the iterative procedure. ‘lm’: least squares regression is used within the iterative procedure. ‘classical’: principal component analysis is used within the iterative procedure. ‘ltsReg2’: least trimmed squares regression is used within the iterative procedure. The imputed values are perturbed in the direction of the predictor by values drawn form a normal distribution with mean and standard deviation related to the corresponding residuals and multiplied by noise.
**Value**

- xOrig: Original data frame or matrix
- xImp: Imputed data
- criteria: Sum of the Aitchison distances from the present and previous iteration
- iter: Number of iterations
- maxit: Maximum number of iterations
- w: Amount of imputed values
- wind: Index of the missing values in the data

**Author(s)**

Matthias Templ, Karel Hron

**References**


**See Also**

impKNNa, pivotCoord

**Examples**

```r
data(expenditures)
x <- expenditures
x[1,3]
x[1,3] <- NA
xi <- impCoda(x)$xImp
xi[1,3]
s1 <- sum(x[1,-3])
imps <- sum(xi[1,-3])
xi[,3] * s1/imps
```

---

**Description**

This function offers several k-nearest neighbor methods for the imputation of missing values in compositional data.
impKNNa

Usage

impKNNa(x, method = "knn", k = 3, metric = "Aitchison",
agg = "median", primitive = FALSE, normknn = TRUE, das = FALSE,
adj = "median")

Arguments

x       data frame or matrix
method  method (at the moment, only “knn” can be used)
k       number of nearest neighbors chosen for imputation
metric “Aitchison” or “Euclidean”
agg     “median” or “mean”, for the aggregation of the nearest neighbors
primitive if TRUE, a more enhanced search for the $k$-nearest neighbors is obtained (see details)
normknn An adjustment of the imputed values is performed if TRUE
das     deprecated. if TRUE, the definition of the Aitchison distance, based on simple
         logratios of the compositional part, is used (Aitchison, 2000) to calculate dis-
         tances between observations. if FALSE, a version using the clr transformation
         is used.
adj     either ‘median’ (default) or ‘sum’ can be chosen for the adjustment of the nearest
         neighbors, see Hron et al., 2010.

Details

The Aitchison metric should be chosen when dealing with compositional data, the Euclidean
metric otherwise.

If primitive == FALSE, a sequential search for the $k$-nearest neighbors is applied for every
missing value where all information corresponding to the non-missing cells plus the information in
the variable to be imputed plus some additional information is available. If primitive == TRUE,
a search of the $k$-nearest neighbors among observations is applied where in addition to the variable
to be imputed any further cells are non-missing.

If normknn is TRUE (preferred option) the imputed cells from a nearest neighbor method are adjusted
with special adjustment factors (more details can be found online (see the references)).

Value

xOrig   Original data frame or matrix
xImp    Imputed data
w       Amount of imputed values
wind    Index of the missing values in the data
metric  Metric used

Author(s)

Matthias Templ
References


See Also
impCoda

Examples

data(expenditures)
x <- expenditures
x[1,3]
x[1,3] <- NA
xi <- impKNNa(x)$xImp
xi[1,3]

impRZalr alr EM-based Imputation for Rounded Zeros

Description
A modified EM alr-algorithm for replacing rounded zeros in compositional data sets.

Usage
impRZalr(x, pos = ncol(x), dl = rep(0.05, ncol(x) - 1), eps = 1e-04,
maxit = 50, bruteforce = FALSE, method = "lm", step = FALSE,
nComp = "boot", R = 10, verbose = FALSE)

Arguments
x Compositional data
pos Position of the rationing variable for alr transformation
dl Detection limit for each part
eps convergence criteria
maxit maximum number of iterations
bruteforce if TRUE, imputations over dl are set to dl. If FALSE, truncated (Tobit) regression is applied.
method either "lm" (default) or "MM"
step if TRUE, a stepwise (AIC) procedure is applied when fitting models
nComp if determined, it fixes the number of pls components. If “boot”, the number of pls components are estimated using a bootstraped cross validation approach.
R number of bootstrap samples for the determination of pls components. Only important for method “pls”.
verbose additional print output during calculations.

Details

Statistical analysis of compositional data including zeros runs into problems, because log-ratios cannot be applied. Usually, rounded zeros are considerer as missing not at random missing values. The algorithm first applies an additive log-ratio transformation to the compositions. Then the rounded zeros are imputed using a modified EM algorithm.

Value

xOrig Original data frame or matrix
xImp Imputed data
wind Index of the missing values in the data
iter Number of iterations
eps eps

Author(s)

Matthias Templ and Karel Hron

See Also

impRZalr

Examples

data(arcticLake)
x <- arcticLake
## generate rounded zeros artificially:
x[x[,1] < 5, 1] <- 0
x[x[,2] < 47, 2] <- 0
xia <- impRZalr(x, pos=3, dl=c(5,47), eps=0.05)
xia$xImp
**imprZilr**  
*EM-based replacement of rounded zeros in compositional data*

**Description**
Parametric replacement of rounded zeros for compositional data using classical and robust methods based on ilr-transformations with special choice of balances.

**Usage**
```r
imprZilr(x, maxit = 10, eps = 0.1, method = "pls", dl = rep(0.05, ncol(x)), variation = FALSE, nComp = "boot", bruteforce = FALSE, noisemethod = "residuals", noise = FALSE, R = 10, correction = "normal", verbose = FALSE)
```

**Arguments**
- `x`: data.frame or matrix
- `maxit`: maximum number of iterations
- `eps`: convergence criteria
- `method`: either "lm", "MM" or "pls"
- `dl`: Detection limit for each variable. zero for variables with variables that have no detection limit problems.
- `variation`: matrix is used to first select number of parts
- `nComp`: if determined, it fixes the number of pls components. If "boot", the number of pls components are estimated using a bootstrapped cross validation approach.
- `bruteforce`: sets imputed values above the detection limit to the detection limit. Replacement above the detection limit are only exceptionally occur due to numerical instabilities. The default is FALSE!
- `noisemethod`: adding noise to imputed values. Experimental
- `noise`: TRUE to activate noise (experimental)
- `R`: number of bootstrap samples for the determination of pls components. Only important for method "pls".
- `correction`: normal or density
- `verbose`: additional print output during calculations.

**Details**
Statistical analysis of compositional data including zeros runs into problems, because log-ratios cannot be applied. Usually, rounded zeros are considerer as missing not at random missing values. The algorithm iteratively imputes parts with rounded zeros whereas in each step (1) an specific ilr transformation is applied (2) tobit regression is applied (3) the rounded zeros are replaced by the expected values (4) the corresponding inverse ilr transformation is applied. After all parts are imputed, the algorithm starts again until the imputations do not change.
Value

- `x` imputed data
- `criteria` change between last and second last iteration
- `iter` number of iterations
- `maxit` maximum number of iterations
- `wind` index of zeros
- `nComp` number of components for method pls
- `method` chosen method

Author(s)

Matthias Templ and Peter Filzmoser

See Also

`imprzalr`

Examples

```r
data(arcticLake)
x <- arcticLake
## generate rounded zeros artificially:
x[x[,1] < 5, 1] <- 0
x[x[,2] < 44, 2] <- 0
xia <- imprzalr(x, dl=c(5,44,0), eps=0.01, method="lm")
xia$x
```

Usage

```r
imputeBDLs(x, maxit = 10, eps = 0.1, method = "subPLS",
dl = rep(0.05, ncol(x)), variation = TRUE, nPred = NULL,
nComp = "boot", bruteforce = FALSE, noisemethod = "residuals",
noise = FALSE, R = 10, correction = "normal", verbose = FALSE,
test = FALSE)

adjustImputed(xImp, xOrig, wind)
```
checkData(x, dl)

## S3 method for class 'replaced'
print(x, ...)

Arguments

- **x**: data.frame or matrix
- **maxit**: maximum number of iterations
- **eps**: convergency criteria
- **method**: either "lm", "lmrob" or "pls"
- **dl**: Detection limit for each variable. zero for variables with variables that have no detection limit problems.
- **variation,**: if TRUE those predictors are chosen in each step, who's variation is lowest to the predictor.
- **nPred,**: if determined and variation equals TRUE, it fixes the number of predictors
- **nComp,**: if determined, it fixes the number of pls components. If "boot", the number of pls components are estimated using a bootstrapped cross validation approach.
- **bruteforce**: sets imputed values above the detection limit to the detection limit. Replacement above the detection limit are only expectionally occur due to numerical instabilities. The default is FALSE!
- **noisemethod**: adding noise to imputed values. Experimental
- **noise**: TRUE to activate noise (experimental)
- **R**: number of bootstrap samples for the determination of pls components. Only important for method “pls”.
- **correction**: normal or density
- **verbose**: additional print output during calculations.
- **test**: an internal test situation (this parameter will be deleted soon)
- **xImp**: imputed data set
- **xOrig**: original data set
- **wind**: index matrix of rounded zeros
- **...**: further arguments passed through the print function

Details

Statistical analysis of compositional data including zeros runs into problems, because log-ratios cannot be applied. Usually, rounded zeros are considerer as missing not at random missing values.

The algorithm iteratively imputes parts with rounded zeros whereas in each step (1) an specific ilr transformation is applied (2) tobit regression is applied (3) the rounded zeros are replaced by the expected values (4) the corresponding inverse ilr transformation is applied. After all parts are imputed, the algorithm starts again until the imputations do not change.
Value

  x  imputed data
criteria change between last and second last iteration
iter number of iterations
maxit maximum number of iterations
wind index of zeros
nComp number of components for method pls
method chosen method

Author(s)

Matthias Templ, method subPLS from Jiajia Chen

References


See Also

imputeBDLs

Examples

```r
p <- 10
n <- 50
k <- 2
T <- matrix(rnorm(n*k), ncol=k)
B <- matrix(runif(p*k,-1,1),ncol=k)
X <- T %*% t(B)
E <- matrix(rnorm(n*p, 0,0.1), ncol=p)
XE <- X + E
data <- data.frame(pivotCoordInv(XE))
col <- ncol(data)
row <- nrow(data)
DL <- matrix(rep(0),ncol=col,nrow=1)
for(j in seq(1,col,2))
{DL[j] <- quantile(data[,j],probs=0.06,na.rm=FALSE)}

for(j in 1:col)
{data[data[,j]<DL[j],j] <- 0}

imp <- imputeBDLs(data,dl=DL,maxit=10,eps=0.1,R=10,method="subPLS")
```
imputeUDLs

Imputation of values above an upper detection limit in compositional data

Description

Parametric replacement of values above upper detection limit for compositional data using classical and robust methods (possibly also the pls method) based on ilr-transformations with special choice of balances.

Usage

imputeUDLs(x, maxit = 10, eps = 0.1, method = "lm", dl = NULL, variation = TRUE, nPred = NULL, nComp = "boot", bruteforce = FALSE, noisemethod = "residuals", noise = FALSE, R = 10, correction = "normal", verbose = FALSE)
Arguments

- **x**: data.frame or matrix
- **maxit**: maximum number of iterations
- **eps**: convergency criteria
- **method**: either "lm", "lmrob" or "pls"
- **dl**: Detection limit for each variable. zero for variables with variables that have no detection limit problems.
- **variation**: if TRUE those predictors are chosen in each step, who’s variation is lowest to the predictor.
- **nPred**: if determined and variation equals TRUE, it fixes the number of predictors
- **nComp**: if determined, it fixes the number of pls components. If “boot”, the number of pls components are estimated using a bootstraped cross validation approach.
- **bruteforce**: sets imputed values above the detection limit to the detection limit. Replacement above the detection limit are only exepctionally occur due to numerical instabilities. The default is FALSE!
- **noisemethod**: adding noise to imputed values. Experimental
- **noise**: TRUE to activate noise (experimental)
- **R**: number of bootstrap samples for the determination of pls components. Only important for method “pls”.
- **correction**: normal or density
- **verbose**: additional print output during calculations.

Details

**imputeUDLs**

An imputation method for right-censored compositional data. Statistical analysis is not possible with values reported in data, for example as ">10000". These values are replaced using tobit regression.

The algorithm iteratively imputes parts with values above upper detection limit whereas in each step (1) a specific ilr transformation is applied (2) tobit regression is applied (3) the values above upper detection limit are replaced by the expected values (4) the corresponding inverse ilr transformation is applied. After all parts are imputed, the algorithm starts again until the imputations only change marginally.

Value

- **x**: imputed data
- **criteria**: change between last and second last iteration
- **iter**: number of iterations
- **maxit**: maximum number of iterations
- **wind**: index of values above upper detection limit
- **nComp**: number of components for method pls
- **method**: chosen method
ind2x2

Author(s)

Peter Filzmoser, Dominika Miksova based on function imputeBDLs code

References


See Also

imputeBDLs

Examples

data(gemas)  # read data
dat <- gemas[gemas$COUNTRY=="HEL",c(12:29)]
UDL <- apply(dat,2,max)
names(UDL) <- names(dat)
UDL["Mn"] <- quantile(dat,"Mn", probs = 0.8)  # UDL present only in one variable
whichudl <- dat[,"Mn"] > UDL["Mn"]
  # classical method
imp.lm <- dat
imp.lm[whichudl,"Mn"] <- Inf
res.lm <- imputeUDLs(imp.lm, dl=UDL, method="lm", variation=TRUE)
imp.lm <- res.lm$x

---------

ind2x2  Independence 2x2 table

Description

Estimates the expected frequencies from an 2x2 table under the null hypotheses of independence.

Usage

ind2x2(x, margin = 3, pTabMethod = c("dirichlet", "half", "classical"))
**Arguments**

- **x**
  - a 2x2 table

- **margin**
  - if multidimensional table (larger than 2-dimensional), then the margin determines on which dimension the independent tables should be estimated.

- **ptabMethod**
  - ‘classical’ that is function `prop.table()` from package base or method “half” that add 1/2 to each cell to avoid zero problems.

**Value**

The independence table(s) with either relative or absolute frequencies.

**Author(s)**

Kamila Facevicova, Matthias Templ

**References**


**Examples**

```r
data(employment)
ind2x2(employment)
```

---

**Description**

Estimates the expected frequencies from an m-way table under the null hypotheses of independence.

**Usage**

```r
indTab(x, margin = c("gmean_sum", "sum"), frequency = c("relative", "absolute"), pTabMethod = c("dirichlet", "half", "classical"))
```

**Arguments**

- **x**
  - an object of class table

- **margin**
  - determines how the margins of the table should be estimated (default via geometric mean margins)

- **frequency**
  - indicates whether absolute or relative frequencies should be computed.

- **ptabMethod**
  - to estimate the probability table. Default is ‘dirichlet’. Other available methods: ‘classical’ that is function `prop.table()` from package base or method “half” that add 1/2 to each cell to avoid zero problems.
Because of the compositional nature of probability tables, the independence tables should be estimated using geometric margins.

The independence table(s) with either relative or absolute frequencies.

Matthias Templ


`data(precipitation)`
`tab1 <- indTab(precipitation)`
`tab1`
`sum(tab1)`

```r
## Not run:
data("PreSex", package = "vcd")
indTab(PreSex)
## End(Not run)
```

`data(instw)`

value added, output and input for different ISIC codes and countries.

- `ctct`
- isicISIC classification, Rev 3.2
- `va`value added
- `ouToutput`
- `inpinput`
- `IS03country code`
- `mhtmht`

`data(instw)`
Format

A data.frame with 1555 rows and 7 columns.

Examples

data(int2x2)
head(int2x2)

int2x2  Interaction 2x2 table

Description

Estimates the interactions from an 2x2 table under the null hypotheses of independence.

Usage

int2x2(x, margin = 3, pTabMethod = c("dirichlet", "half", "classical"))

Arguments

x a 2x2 table
margin if multidimensional table (larger than 2-dimensional), then the margin determines on which dimension the independene tables should be estimated.
pTabMethod to estimate the propability table. Default is ‘dirichlet’. Other available methods: ‘classical’ that is function prop.table() from package base or method “half” that add 1/2 to each cell to avoid zero problems.

Value

The independence table(s) with either relative or absolute frequencies.

Author(s)

Kamila Facevicova, Matthias Templ

References


Examples

data(employment)
int2x2(employment)
intArray

Description

Estimates the interaction array

Usage

intArray(x)

Arguments

x an object of class "intTab"

Details

Estimates the interaction array using an ilr transformation of the interaction table.

Value

The interaction array

Author(s)

Matthias Templ

References


See Also

intTab

Examples

data(precipitation)
tab1prob <- prop.table(precipitation)
tab1 <- indTab(precipitation)
tabINT <- intTab(tab1prob, tab1)
tabINT
**intTab**

*Interaction table*

**Description**

Estimates the interaction table based on clr and inverse clr transformations.

**Usage**

```r
intTab(x, y, frequencies = c("relative", "absolute"))
```

**Arguments**

- `x` an object of class table
- `y` the corresponding independence table which is of class “intTab”.
- `frequencies` indicates whether absolute or relative frequencies should be computed.

**Details**

Because of the compositional nature of probability tables, the independence tables should be estimated using geometric margins.

**Value**

- `intTab` The interaction table(s) with either relative or absolute frequencies.
- `signs` The sign illustrates if there is an excess of probability (plus), or a deficit (minus) regarding to the estimated probability table and the independence table in the clr space.

**Author(s)**

Matthias Templ

**References**


**Examples**

```r
data(precipitation)
tablprob <- prop.table(precipitation)
tabl <- indTab(precipitation)
intTab(tablprob, tabl)
```
isic32  ISIC codes by name

Description

- code: ISIC code, Rev 3.2
- description: Description of ISIC codes

Usage

data(isic32)

Format

A data.frame with 24 rows and 2 columns.

Examples

data(instw)
instw

laborForce  labour force by status in employment

Description

Labour force by status in employment for 124 countries, latest update: December 2009

Format

A data set on 124 compositions on 9 variables.

Details

- country
- year
- employeesW percentage female employees
- employeesM percentage male employees
- employersW percentage female employers
- employersM percentage male employers
- ownW percentage female own-account workers and contributing family workers
- ownM percentage male own-account workers and contributing family workers
- source HS: household or labour force survey. OE: official estimates. PC: population census
Author(s)
conversion to R by Karel Hron and Matthias Templ <matthias.templ@tuwien.ac.at>

Source

References

Examples

```r
data(laborForce)
str(laborForce)
```

---

**lifeExpGdp**

*life expectancy and GDP (2008) for EU-countries*

Description
Social-economic data for compositional regression.

Format
A data set on 27 compositions on 9 variables.

Details
- country
country
- agriculture GDP on agriculture, hunting, forestry, fishing (ISIC A-B, x1)
- manufacture GDP on mining, manufacturing, utilities (ISIC C-E, x2)
- construction GDP on construction (ISIC F, x3)
- wholesales GDP on wholesale, retail trade, restaurants and hotels (ISIC G-H, x4)
- transport GDP on transport, storage and communication (ISIC I, x5)
- other GDP on other activities (ISIC J-P, x6)
- lifeExpMen life expectancy for men and women
- lifeExpWomen life expectancy for men and women

Author(s)
conversion to R by Karel Hron and Matthias Templ <matthias.templ@tuwien.ac.at>
Source


References


Examples

data(lifeExpGdp)
str(lifeExpGdp)

----

lmCoDaX

Classical and robust regression of non-compositional response on compositional predictors

Description

Delivers appropriate inference for regression of y on a compositional matrix X.

Usage

lmCoDaX(y, X, method = "robust")

Arguments

y The response which should be non-compositional
X The compositional predictors as a matrix, data.frame or numeric vector
method If robust, LTS-regression is applied, while with method equals “classical”, the conventional least squares regression is applied.

Details

Compositional explanatory variables should not be directly used in a linear regression model because any inference statistic can become misleading. While various approaches for this problem were proposed, here an approach based on the isometric logratio (ilr) transformation is used.

Value

An object of class ‘lts’ or ‘lm’ and two summary objects.

Author(s)

Peter Filzmoser
References


See Also

lm

Examples

```r
## How the total household expenditures in EU Member
## States depend on relative contributions of
## single household expenditures:
data(expendituresEU)
y <- as.numeric(apply(expendituresEU, 1, sum))
ImCoDoAx(y, expendituresEU, method="classical")
ImCoDoAx(y, expendituresEU, method="robust")
```

machineOperators  machine operators

Description

Compositions of eight-hour shifts of 27 machine operators

Usage

data(machineOperators)

Format

A data frame with 27 observations on the following 4 variables.

Details

- `hqproduction` high-quality production
- `lqproduction` low-quality production
- `setting` machine settings
- `repair` machine repair


Author(s)

Matthias Templ <matthias.templ@tuwien.ac.at>
References


Examples

data(machineOperators)
str(machineOperators)
supply(machineOperators)
rowSums(machineOperators)

mcd

metabolomics mcd data set

Description

The aim of the experiment was to ascertain novel biomarkers of MCAD (Medium chain acyl-CoA dehydrogenase) deficiency. The data consists of 25 patients and 25 controls and the analysis was done by LC-MS. Rows represent patients and controls and columns represent chemical entities with their quantity.

Usage

data(mcd)

Format

A data frame with 50 observations and 279 variables

Details

- group patient group
- . . . the remaining variables columns are represented by m/z which are chemical characterizations of individual chemical components on exact mass measurements.

References


Examples

data(mcd)
str(mcd)
Description

Analysis of the missing or the zero patterns structure of a data set.

Usage

missPatterns(x)

teropatterns(x)

Arguments

x        a data frame or matrix.

Details

Here, one pattern defines those observations that have the same structure regarding their missingness or zeros. For all patterns a summary is calculated.

Value

groups    List of the different patterns and the observation numbers for each pattern
        the names of the patterns coded as vectors of 0-1’s

tabcomb   the pattern structure - all combinations of zeros or missings in the variables

tabcombPlus the pattern structure - all combinations of zeros or missings in the variables including the size of those combinations/patterns, i.e. the number of observations that belongs to each pattern.

rsum     the number of zeros or missing values in each row of the data set.

rindex   the index of zeros or missing values in each row of the data set

Author(s)

Matthias Templ. The code is based on a previous version from Andreas Alfons and Matthias Templ from package VIM

See Also

aggr
Examples

```r
data(expenditures)
## set NA's artificial:
expenditures[expenditures < 300] <- NA
## detect the NA structure:
missPatterns(expenditures)
```

---

<table>
<thead>
<tr>
<th>mortality</th>
<th>mortality and life expectancy in the EU</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Description

- `country` country name
- `country2` country name, short version
- `sex` gender
- `lifeExpectancy` life expectancy
- `infectious` certain infectious and parasitic diseases (A00-B99)
- `neoplasms` malignant neoplasms (C00-C97)
- `endocrine` endocrine nutritional and metabolic diseases (E00-E90)
- `mental` mental and behavioural disorders (F00-F99)
- `nervous` diseases of the nervous system and the sense organs (G00-H95)
- `circulatory` diseases of the circulatory system (I00-I99)
- `respiratory` diseases of the respiratory system (J00-J99)
- `digestive` diseases of the digestive system (K00-K93)

Usage

```r
data(mortality)
```

Format

A data frame with 60 observations and 12 variables

Author(s)

Peter Filzmoser, Matthias Templ &lt;matthias.templ@tuwien.ac.at&gt;

References

Examples

```r
data(mortality)
str(mortality)
## totals (mortality)
aggregate(mortality[,5:ncol(mortality)],
   list(mortality$country2), sum)
```

Description

Mortality data by gender, unknown year

Usage

```r
data(mortality_tab)
```

Format

A table

Details

- female mortality rates for females by age groups
- male mortality rates for males by age groups

Author(s)

Matthias Templ

Examples

```r
data(mortality_tab)
mortality_tab
```
Description

Nutrients on more than 40 components and 965 generic food products

Usage

data(nutrients)

Format

A data frame with 965 observations on the following 50 variables.

Details

- ID  ID, for internal use
- ID_v4  ID V4, for internal use
- ID_swissfir  ID, for internal use
- name_d  Name in German
- name_f  Name in French
- name_i  Name in Italian
- name_e  Name in Spanish
- category_d  Category name in German
- category_f  Category name in French
- category_i  Category name in Italy
- category_e  Category name in Spanish
- gravity  specific gravity
- `energy_kj`  `energy in kJ per 100g edible portion`
- energy_kcal  energy in kcal per 100g edible portion
- protein  protein in gram per 100g edible portion
- alcohol  alcohol in gram per 100g edible portion
- water  water in gram per 100g edible portion
- carbohydrates  carbohydrate in gram per 100g edible portion
- starch  starch in gram per 100g edible portion
- sugars  sugars in gram per 100g edible portion
- `dietary_fibres`  `dietar fibres in gram per 100g edible portion`
- fat  fat in gram per 100g edible portion
- cholesterol  cholesterol in milligram per 100g edible portion
• fattyacids_monounsaturated  fatty acids monounsaturated in gram per 100g edible portion
• fattyacids_saturated  fatty acids saturated in gram per 100g edible portion
• fatty_acids_polyunsaturated fatty acids polyunsaturated in gram per 100g edible portion
• vitaminA  vitamin A in retinol equivalent per 100g edible portion
• 'all-trans_retinol_equivalents  'all trans-retinol equivalents in gram per 100g edible portion
• 'beta-carotene-activity  'beta-carotene activity in beta-carotene equivalent per 100g edible portion
• 'beta-carotene  'beta-carotene in micogram per 100g edible portion
• vitaminB1  vitamin B1 in milligram per 100g edible portion
• vitaminB2  vitamin B2 in milligram per 100g edible portion
• vitaminB6  vitamin B6 in milligram per 100g edible portion
• vitaminB12  vitamin B12 in micogram per 100g edible portion
• niacin  niacin in milligram per 100g edible portion
• folate  folate in micogram per 100g edible portion
• pantothenic_acid  pantothenic acid in millgram per 100g edible portion
• vitaminC  vitamin C in milligram per 100g edible portion
• vitaminD  vitamin D in micogram per 100g edible portion
• vitaminE  vitamin E in alpha-tocopherol equivalent per 100g edible portion
• Na  Sodium in millgram per 100g edible portion
• K  Potassium in millgram per 100g edible portion
• Ca  Calcium
• Mg  Magnesium
• P  Phosphorus
• Fe  Iron
• I  Iodide in millgram per 100g edible portion
• Zn  Zink
• unit  a factor with levels per 100g edible portion per 100ml food volume

Author(s)
Translated from the Swiss nutrition data base by Matthias Templ <matthias.templ@tuwien.ac.at>

Source
From the Swiss nutrition data base 2015 (second edition), see http://www.sge-ssn.ch/shop/produkt/schweizer-naehrwerttabelle/
nutrients_branded

References

http://www.sge-ssn.ch/shop/produkt/schweizer-naehrwerttabelle/

Examples

data(nutrients)
str(nutrients)
head(nutrients[, 41:49])

<table>
<thead>
<tr>
<th>nutrients_branded</th>
<th>nutrient contents (branded)</th>
</tr>
</thead>
</table>

Description

Nutrients on more than 10 components and 9618 branded food products

Usage

data(nutrients_branded)

Format

A data frame with 9618 observations on the following 18 variables.

Details

- name_D  name (in German)
- category_D  factor specifying the category names
- category_F  factor specifying the category names
- category_I  factor specifying the category names
- category_E  factor specifying the category names
- gravity  specific gravity
- energy_kJ  energy in kJ
- 'energy_kcal 'energy in kcal
- protein  protein in gram
- alcohol  alcohol in gram
- water  water in gram
- carbohydrates_available  available carbohydrates in gram
- sugars  sugars in gram
- dietary_fibres  dietary fibres in gram
- fat_total  total fat in gram
- fatty_acids_saturated  saturated acids fat in gram
- Na  Sodium in gram
- unit  a factor with levels per 100g edible portion per 100ml food volume
Orthonormal basis from cenLR transformed data to pivotCoord transformated data.

For the chosen balances for “pivotCoord”, this is the orthonormal basis that transfers the data from centered logratio to isometric logratio.

The orthonormal basis.

Karel Hron, Matthias Templ. Some code lines of this function are a copy from function gsi.buildilr from
Outlier detection for compositional data using standard and robust statistical methods.

**Usage**

```r
outCoDa(x, quantile = 0.975, method = "robust", h = 1/2,
        coda = TRUE)
```

**Examples**

```r
data(expenditures)
V <- orthbasis(ncol(expenditures))
xcen <- cenLR(expenditures)$x.clr
xi <- as.matrix(xcen) %*% V$V
xi
xi2 <- pivotCoord(expenditures)
xi2
```

**Arguments**

- `x` : compositional data
- `quantile` : quantile, corresponding to a significance level, is used as a cut-off value for outlier identification: observations with larger (squared) robust Mahalanobis distance are considered as potential outliers.
- `method` : either “robust” (default) or “standard”
- `h` : the size of the subsets for the robust covariance estimation according the MCD-estimator for which the determinant is minimized (the default is \((n+p+1)/2\)).
- `coda` : if TRUE, data transformed to coordinate representation before outlier detection.
- `...` : additional parameters for print and plot method passed through
- `y` : unused second plot argument for the plot method
- `which` : 1 ... MD against index 2 ... distance-distance plot
Details

The outlier detection procedure is based on (robust) Mahalanobis distances after a isometric logratio transformation of the data. Observations with squared Mahalanobis distance greater equal a certain quantile of the Chi-squared distribution are marked as outliers.

If method “robust” is chosen, the outlier detection is based on the homogeneous majority of the compositional data set. If method “standard” is used, standard measures of location and scatter are applied during the outlier detection procedure.

plot method: the Mahalanobis distance are plotted against the index. The dashed line indicates the (1 - alpha) quantile of the Chi-squared distribution. Observations with Mahalanobis distance greater than this quantile could be considered as compositional outliers.

Value

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mahaDist</td>
<td>resulting Mahalanobis distance</td>
</tr>
<tr>
<td>limit</td>
<td>quantile of the Chi-squared distribution</td>
</tr>
<tr>
<td>outlierIndex</td>
<td>logical vector indicating outliers and non-outliers</td>
</tr>
<tr>
<td>method</td>
<td>method used</td>
</tr>
</tbody>
</table>

Note

It is highly recommended to use the robust version of the procedure.

Author(s)

Matthias Templ, Karel Hron

References


See Also

pivotCoord

Examples

data(expenditures)
oD <- outCoDa(expenditures)
oD

## providing a function:
oD <- outCoDa(expenditures, coda = log)
Description

Payments splitted by different NACE categories and kind of employment in Austria 2004

Usage

data(payments)

Format

A data frame with 535 rows and 11 variables

Details

• nace    NACE classification, 2 digits
• oenace_2008  Corresponding Austrian NACE classification (in German)
• year    year
• month   month
• localunit  local unit ID
• spay     special payments (total)
• spay_wc   special payments for white color workers
• spay_bc   special payments for blue color workers
• spay_traintrade  special payments for trainees in trade business
• spay_home special payments for home workers
• spay_traincomm special payments for trainees in commercial business

Author(s)

Matthias Templ <matthias.templ@tuwien.ac.at>

Source

statCube data base at www.statistik.ac.at. The product and all material contained therein are protected by copyright with all rights reserved by the Bundesanstalt Statistik Oesterreich (STATISTICS AUSTRIA). It is permitted to reproduce, distribute, make publicly available and process the content for non-commercial purposes. Prior to any use for commercial purposes a written consent of STATISTICS AUSTRIA must be obtained. Any use of the contained material must be correctly reproduced and clearly cite the source STATISTICS AUSTRIA. If tables published by STATISTICS AUSTRIA are partially used, displayed or otherwise changed, a note must be added at an adequate position to show data was extracted or adapted.
Examples

```r
data(payments)
str(payments)
summary(payments)
```

---

**pcaCoDa**

*Robust principal component analysis for compositional data*

---

**Description**

This function applies robust principal component analysis for compositional data.

**Usage**

```r
pcaCoDa(x, method = "robust", mult_comp = NULL, external = NULL)
```

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

```r
## S3 method for class 'pcaCoDa'
summary(object, ...)
```

**Arguments**

- `x` composition data
- `method` must be either “robust” (default) or “classical”
- `mult_comp` a list of numeric vectors holding the indices of linked compositions
- `external` external non-compositional variables
- `...` additional parameters for print method passed through
- `object` object of class pcaCoDa

**Details**

The compositional data set is transformed using the ilr tranformation. Afterwards, robust principal component analysis is performed. Resulting loadings and scores are back-transformed to the clr space where the compositional biplot can be shown.

`mult_comp` is used when there are more than one group of compositional parts in the data. To give an illustrative example, lets assume that one variable group measures angles of the inner ear-bones of animals which sum up to 100 and another one having percentages of a whole on the thickness of the inner ear-bones included. Then two groups of variables exists which are both compositional parts. The ilr-transformation is then internally applied to each group independently whenever the `mult_comp` is set correctly.
Value

scores           scores in clr space
loadings         loadings in clr space
eigenvalues      eigenvalues of the clr covariance matrix
method           method
princompOutputClr   output of princomp needed in plot.pcaCoDa

Author(s)


References


See Also

`print.pcaCoDa`, `summary.pcaCoDa`, `biplot.pcaCoDa`, `plot.pcaCoDa`

Examples

data(arcticLake)

## robust estimation (default):
res.rob <- pcaCoDa(arcticLake)
res.rob
summary(res.rob)
plot(res.rob)

## classical estimation:
res.cla <- pcaCoDa(arcticLake, method="classical")
biplot(res.cla)

## just for illustration how to set the mult_comp argument:
data(expenditures)
pl <- pcaCoDa(expenditures, mult_comp=list(c(1,2,3),c(4,5)))
pl

## example with external variables:
data(election)
# transform external variables
election$unemployment <- log((election$unemployment/100)/(1-election$unemployment/100))
election$income <- scale(election$income)
Perturbation and powering for two compositions.

Usage

perturbation(x, y)

powering(x, a)

Arguments

x (compositional) vector containing positive values
y (compositional) vector containing positive values or NULL for powering
a constant, numeric vector of length 1

Value

Result of perturbation or powering

Author(s)

Matthias Templ

References


Examples

data(expenditures)
x <- expenditures[1,]
y <- expenditures[2,]
perturbation(x, y)
powering(x, 2)
**Description**

Computes the principal factor analysis of the input data which are transformed and centered first.

**Usage**

\[
pfa(x, \text{factors}, \text{robust} = \text{TRUE}, \text{data} = \text{NULL}, \text{covmat} = \text{NULL}, \text{n.obs} = \text{NA}, \text{subset}, \text{na.action}, \text{start} = \text{NULL}, \text{scores} = \text{c("none", "regression", "Bartlett"), rotation = "varimax", maxiter = 5, control = \text{NULL}, ...})
\]

**Arguments**

- **x** (robustly) scaled input data
- **factors** number of factors
- **robust** default value is TRUE
- **data** default value is NULL
- **covmat** (robustly) computed covariance or correlation matrix
- **n.obs** number of observations
- **subset** if a subset is used
- **na.action** what to do with NA values
- **start** starting values
- **scores** which method should be used to calculate the scores
- **rotation** if a rotation should be made
- **maxiter** maximum number of iterations
- **control** default value is NULL
- **...** arguments for creating a list

**Details**

The main difference to usual implementations is that uniquenesses are no longer of diagonal form. This kind of factor analysis is designed for centered log-ratio transformed compositional data. However, if the covariance is not specified, the covariance is estimated from isometric log-ratio transformed data internally, but the data used for factor analysis are backtransformed to the clr space (see Filzmoser et al., 2009).
Value

loadings  A matrix of loadings, one column for each factor. The factors are ordered in decreasing order of sums of squares of loadings.

uniqueness  uniqueness

correlation  correlation matrix

criteria  The results of the optimization: the value of the negative log-likelihood and information of the iterations used.

factors  the factors

dof  degrees of freedom

method  “principal”

n.obs  number of observations if available, or NA

call  The matched call.

STATISTIC, PVAL  The significance-test statistic and p-value, if they can be computed

Author(s)

Peter Filzmoser, Karel Hron, Matthias Templ

References


Examples

data(expenditures)
x <- expenditures
res.rob <- pfa(x, factors=1)
res.cla <- pfa(x, factors=1, robust=FALSE)

```r
## the following produce always the same result:
res1 <- pfa(x, factors=1, covmat="covMcd")
res2 <- pfa(x, factors=1, covmat=covMcd(pivotCoord(x))$cov)
res3 <- pfa(x, factors=1, covmat=covMcd(pivotCoord(x)))
```
Description

PhD students in Europe based on the standard classification system splitted by different kind of studies (given as percentages).

Format

A data set on 32 compositions and 11 variables.

Details

Due to unknown reasons the rowSums of the percentages is not always 100.

- country country of origin (German)
- countryEN country of origin (English)
- country2 country of origin, 2-digits
- total total phd students (in 1.000)
- male male phd students (in 1.000)
- female total phd students (in 1.000)
- technical phd students in natural and technical sciences
- socio-economic-low phd students in social sciences, economic sciences and law sciences
- human phd students in human sciences including teaching
- health phd students in health and life sciences
- agriculture phd students in agriculture

Source

Eurostat

References


Examples

data(phd)
str(phd)
**phd_totals**

*PhD students in the EU (totals)*

---

**Description**

PhD students in Europe by different kind of studies.

**Format**

A data set on 29 compositions and 5 variables.

**Details**

- technical  phd students in natural and technical sciences
- socio-economic-low  phd students in social sciences, economic sciences and law sciences
- human  phd students in human sciences including teaching
- health  phd students in health and life sciences
- agriculture  phd students in agriculture

**Source**

Eurostat

**References**


**Examples**

```r
data("phd_totals")
str(phd_totals)
```
pivotCoord

Pivot coordinates and their inverse

Description
Isometric log-ratio transformations and its inverse transformation with a special choice of balances.

Usage
pivotCoord(x, pivotvar = 1, fast = FALSE, method = "pivot",
  base = exp(1), norm = "orthonormal")

isomLR(x, fast = FALSE, base = exp(1), norm = "sqrt((D-1)/(D-i+1))")

isomLRinv(x)

pivotCoordInv(x, norm = "orthonormal")

isomLRp(x, fast = FALSE, base = exp(1), norm = "sqrt((D-1)/(D-i+1))")

isomLRinvp(x)

Arguments

x
  object of class data.frame or matrix. Positive values only.

pivotvar
  pivotal variable. If any other number than 1, the data are resorted in that sense that the pivotvar is shifted to the first part.

fast
  if TRUE, it is approx. 10 times faster but numerical problems in case of high-dimensional data numerical instabilities may occur. Only available for method "pivot".

method
  pivot takes the method described in the description. Method "symm" uses symmetric balances (parameters pivotvar and norm have then no effect)

base
  a positive or complex number: the base with respect to which logarithms are computed. Defaults to exp(1).

norm
  if FALSE then the normalizing constant is not used, if TRUE sqrt((D-1)/(D-i+1)) is used (default). The user can also specify a self-defined constant.

Details
This transformation moves D-part compositional data from the simplex into a (D-1)-dimensional real space isometrically. From our choice of (pivot) balances, all the relative information of one part is separated from the remaining parts.

Value
The data represented in pivot coordinates
Author(s)
Matthias Templ, Karel Hron, Peter Filzmoser

References

Examples

```r
require(MASS)
Sigma <- matrix(c(5.05, 4.95, 4.95, 5.05), ncol=2, byrow=TRUE)
z <- pivotCoordInv(mvrnorm(100, mu=c(0,2), Sigma=Sigma))

data(expenditures)
## first variable as pivot variable
pivotCoord(expenditures)
## third variable as pivot variable
pivotCoord(expenditures, 3)

x <- exp(mvrnorm(2000, mu=rep(1,10), diag(10)))
system.time(pivotCoord(x))
system.time(pivotCoord(x, fast=TRUE))

## without normalizing constant
pivotCoord(expenditures, norm = "orthogonal") # or:
pivotCoord(expenditures, norm = "1")
## other normalization
pivotCoord(expenditures, norm = "-sqrt((D-i)/(D-i+1))")

# symmetric balances (results in 2-dim symmetric balances)
pivotCoord(expenditures, method = "symm")
```

plot.imp

*Plot method for objects of class imp*

Description

This function provides several diagnostic plots for the imputed data set in order to see how the imputed values are distributed in comparison with the original data values.
Usage

## S3 method for class 'imp'
plot(x, ..., which = 1, ord = 1:ncol(x),
    colcomb = "missnonmiss", plotvars = NULL, col = c("skyblue", "red"),
    alpha = NULL, lty = par("lty"), xaxt = "s",
    xaxlabels = NULL, las = 3, interactive = TRUE, pch = c(1, 3),
    ask = prod(par("mfcol")) < length(which) && dev.interactive(),
    center = FALSE, scale = FALSE, id = FALSE, seg1 = 0.02,
    segl = TRUE)

Arguments

x object of class ‘imp’
...
other parameters to be passed through to plotting functions.
which if a subset of the plots is required, specify a subset of the numbers 1:3.
ord determines the ordering of the variables
colcomb if colcomb="missnonmiss", observations with missings in any variable are high-
lighted. Otherwise, observations with missings in any of the variables specified
by colcomb are highlighted in the parallel coordinate plot.
plotvars Parameter for the parallel coordinate plot. A vector giving the variables to be
plotted. If NULL (the default), all variables are plotted.
col a vector of length two giving the colors to be used in the plot. The second color
will be used for highlighting.
alpha a numeric value between 0 and 1 giving the level of transparency of the colors,
or NULL. This can be used to prevent overplotting.
lty a vector of length two giving the line types. The second line type will be used
for the highlighted observations. If a single value is supplied, it will be used for
both non-highlighted and highlighted observations.
xaxt the x-axis type (see par).
xaxlabels a character vector containing the labels for the x-axis. If NULL, the column
names of x will be used.
las the style of axis labels (see par).
interactive a logical indicating whether the variables to be used for highlighting can be
selected interactively (see ‘Details’).
pch a vector of length two giving the symbol of the plotting points. The symbol will
be used for the highlighted observations. If a single value is supplied, it will be
used for both non-highlighted and highlighted observations.
ask logical; if TRUE, the user is asked before each plot, see par(ask=).
center logical; indicates if the data should be centered prior plotting the ternary plot.
scale logical, indicates if the data should be centered prior plotting the ternary plot.
id reads the position of the graphics pointer when the (first) mouse button is pressed
and returns the corresponding index of the observation. (only used by the ternary
plot)
seg1 length of the plotting symbol (spikes) for the ternary plot.
segl if TRUE, the spikes of the plotting symbol are justified.
Details

The first plot (which == 1) is a multiple scatterplot where for the imputed values another plot symbol and color is used in order to highlight them. Currently, the ggpairs functions from the GGally package is used.

Plot 2 is a parallel coordinate plot in which imputed values in certain variables are highlighted. In parallel coordinate plots, the variables are represented by parallel axes. Each observation of the scaled data is shown as a line. If interactive is TRUE, the variables to be used for highlighting can be selected interactively. Observations which includes imputed values in any of the selected variables will be highlighted. A variable can be added to the selection by clicking on a coordinate axis. If a variable is already selected, clicking on its coordinate axis will remove it from the selection. Clicking anywhere outside the plot region quits the interactive session.

Plot 3 shows a ternary diagram in which imputed values are highlighted, i.e. those spikes of the chosen plotting symbol are colored in red for which of the values are missing in the unimputed data set.

Value

None (invisible NULL).

Author(s)

Matthias Templ

References


See Also

impCoda, impKNNa

Examples

data(expenditures)
expenditures[1,3]
expenditures[1,3] <- NA
xi <- impKNNa(expenditures)
xi
summary(xi)
## Not run: plot(xi, which=1)
plot(xi, which=2)
plot(xi, which=3)
plot(xi, which=3, seg1=FALSE)
Description

Provides a screeplot for (robust) compositional principal components analysis.

Usage

```r
## S3 method for class 'pcaCoDa'
plot(x, y, ...)
```

Arguments

- `x`: object of class ‘pcaCoDa’
- `y`: ...
- `...`: ...

Value

The robust compositional screeplot.

Author(s)

M. Templ, K. Hron

References


See Also

`pcaCoDa, biplot.pcaCoDa`

Examples

```r
data(coffee)
p1 <- pcaCoDa(coffee[,-1])
plot(p1)
plot(p1, type="lines")
```
precipitation

<table>
<thead>
<tr>
<th>precipitation</th>
<th>24-hour precipitation</th>
</tr>
</thead>
</table>

Description

table containing counts for 24-hour precipitation for season at the rain-gouge.

Usage

data(precipitation)

Format

A table with 4 rows and 6 columns

Details

- spring numeric vector on counts for different level of precipitation
- summer numeric vector on counts for different level of precipitation
- autumn numeric vector on counts for different level of precipitation
- winter numeric vector on counts for different level of precipitation

Author(s)

Matthias Templ <matthias.templ@tuwien.ac.at>

References


Examples

data(precipitation)
precipitation
str(precipitation)
print.imp  

Print method for objects of class imp

Description

The function returns a few information about how many missing values are imputed and possible other information about the amount of iterations, for example.

Usage

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

Arguments

- `x` an object of class ‘imp’
- `...` additional arguments passed trough

Value

None (invisible NULL).

Author(s)

Matthias Templ

See Also

impCoda, impKNNa

Examples

```r
data(expenditures)
expenditures[1,3] = NA
## Not run:
x1 <- impCoda(expenditures)
x1
summary(x1)
plot(x1, which=1:2)
## End(Not run)
```
production

production splitted by nationality on enterprise level

Description

- nace  NACE classification, 2 digits
- oenace_2008  Corresponding Austrian NACE classification (in German)
- year  year
- month  month
- enterprise  enterprise ID
- total  total ...
- home  home ...
- EU  EU ...
- non-EU  non-EU ...

Usage

data(production)

Format

A data frame with 535 rows and 9 variables

Author(s)

Matthias Templ <matthias.templ@tuwien.ac.at>

Source

statCube data base at www.statistik.ac.at. The product and all material contained therein are protected by copyright with all rights reserved by the Bundesanstalt Statistik Oesterreich (STATISTICS AUSTRIA). It is permitted to reproduce, distribute, make publicly available and process the content for non-commercial purposes. Prior to any use for commercial purposes a written consent of STATISTICS AUSTRIA must be obtained. Any use of the contained material must be correctly reproduced and clearly cite the source STATISTICS AUSTRIA. If tables published by STATISTICS AUSTRIA are partially used, displayed or otherwise changed, a note must be added at an adequate position to show data was extracted or adapted.

Examples

data(production)
str(production)
summary(production)
pTab

Propability table

Description
Calculates the propability table using different methods

Usage
pTab(x, method = "dirichlet", alpha = 1/length(as.numeric(x)))

Arguments
x an object of class table
method default is ‘dirichlet’. Other available methods: ‘classical’ that is function prop.table() from package base or method “half” that add 1/2 to each cell to avoid zero problems.
alpha constant used for method ‘dirichlet’

Value
The probability table

Author(s)
Matthias Templ

References

Examples
data(precipitation)
pTab(precipitation)
pTab(precipitation, method = "dirichlet")
rcodes  codes for UNIDO tables

Description

- ISOCNISOCN codes
- OPERATOROperator
- ADESCCountry
- CDESCCountry code
- CDESCCountry destination
- ADESCCountry destination code

Usage

data(rcodes)

Format

A data.frame with 2717 rows and 6 columns.

Examples

data(rcodes)
str(rcodes)

rdcm  relative difference between covariance matrices

Description

The sample covariance matrices are computed with the same isometric transformed observations.

Usage

rdcm(x, y)

Arguments

x  matrix or data frame
y  matrix or data frame of the same size as x.

Details

The difference in covariance structure is based on the Euclidean distance between both covariance estimations.
**Value**

the error measures value

**Author(s)**

Matthias Templ

**References**


**See Also**

rdcm

**Examples**

```r
data(expenditures)
x <- expenditures
x[1,3] <- NA
xi <- impKNNa(x)$xImp
rdcm(expenditures, xi)
```

---

<table>
<thead>
<tr>
<th>rSDev</th>
<th>Relative simplicial deviance</th>
</tr>
</thead>
</table>

**Description**

Relative simplicial deviance

**Usage**

`rSDev(x, y)`

**Arguments**

- `x`: a probability table
- `y`: an interaction table

**Value**

The relative simplicial deviance
Author(s)

Matthias Templ

References


Examples

data(precipitation)
tabprob <- prop.table(precipitation)
tabind <- indTab(precipitation)
tabint <- intTab(tabprob, tabind)
rsdev(tabprob, tabint$intTab)

rsDev.test  Relative simplicial deviance tests

Description

Monte Carlo based contingency table tests considering the compositional approach to contingency tables.

Usage

rsDev.test(x, R = 999, method = "rmultinom")

Arguments

x  matrix, data.frame or table
R  an integer specifying the number of replicates used in the Monte Carlo test.
method  either “rmultinom” (default) or “permutation”.

Details

Method “rmultinom” generate multinomially distributed samples from the independent probability table, which is estimated from x using geometric mean marginals. The relative simplicial deviance of the original data are then compared to the generated ones.

Method “permutation” permutes the entries of x and compares the relative simplicial deviance estimated from the original data to the ones of the permuted data (the independence table is unchanged and originates on x).

Method “rmultinom” should be preferred, while method “permutation” can be used for comparisons.
**Value**

A list with class “htest” containing the following components:

- statistic: the value of the relative simplicial deviance (test statistic).
- method: a character string indicating what type of rSDev.test was performed.
- p.value: the p-value for the test.

**Author(s)**

Matthias Templ, Karel Hron

**References**


**See Also**

rSDev

**Examples**

```r
data(precipitation)
rSDev.test(precipitation)
```

---

**Description**

Simplicial deviance

**Usage**

```r
SDev(x)
```

**Arguments**

- `x`: a probability table

**Value**

The simplicial deviance

**Author(s)**

Matthias Templ
References


Examples

```r
data(precipitation)
tablprob <- prop.table(precipitation)
SDev(tablprob)
```

```
skyeLavas
aphyric skye lavas data
```

Description

AFM compositions of 23 aphyric Skye lavas. This data set can be found on page 360 of the Aitchison book (see reference).

Usage

data(skyeLavas)

Format

A data frame with 23 observations on the following 3 variables.

Details

- sodium-potassium a numeric vector of percentages of Na2O+K2O
- iron a numeric vector of percentages of Fe2O3
- magnesium a numeric vector of percentages of MgO

Author(s)

Matthias Templ <matthiastempl@tuwien.ac.at>

References


Examples

```r
data(skyeLavas)
str(skyeLavas)
summary(skyeLavas)
rowSums(skyeLavas)
```
Description
Social expenditures according to source (public or private) and three important branches (health, old age, incapacity related) in selected OECD countries in 2010. Expenditures are always provided in the respective currency.

Usage
data(socExp)

Format
A data frame with 20 observations on the following 8 variables (country + currency + row-wise sorted cells of 2x3 compositional table).

Details
• country  Country of origin
• currency  Currency unit (in Million)
• health-public  Health from the public
• old-public  Old age expenditures from the public
• incap-public  Incapacity related expenditures from the public
• health-private  Health from private sources
• old-private  Old age expenditures from private sources
• incap-private  Incapacity related expenditures from private sources

Author(s)
conversion to R by Karel Hron Karel Hron and modifications by Matthias Templ <matthias.templ@tuwien.ac.at>

References
OECD, http://www.oecd.org

Examples
data(socExp)
str(socExp)
rowSums(socExp[, 3:ncol(socExp)])
stats

Classical estimates for tables

Description

Some standard/classical (non-compositional) statistics

Usage

stats(x, margins = NULL, statistics = c("phi", "cramer", "chisq", "yates"), maggr = mean)

Arguments

x a data.frame, matrix or table
margins margins
statistics statistics of interest
maggr a function for calculating the mean margins of a table, default is the arithmetic mean

Details

statistics ‘phi’ is the values of the table divided by the product of margins. ‘cramer’ normalize these values according to the dimension of the table. ‘chisq’ are the expected values according to Pearson while ‘yates’ according to Yates.

For the maggr function argument, arithmetic means (mean) should be chosen to obtain the classical results. Any other user-provided functions should be take with care since the classical estimations relies on the arithmetic mean.

Value

List containing all statistics

Author(s)

Matthias Templ

References

Examples

data(precipitation)
tab1 <- indTab(precipitation)
stats(precipitation)
stats(precipitation, statistics = "cramer")
stats(precipitation, statistics = "chisq")
stats(precipitation, statistics = "yates")

## take with care
## (the provided statistics are not designed for that case):
stats(precipitation, statistics = "chisq", maggr = gmean)

summary.imp  Summary method for objects of class imp

Description

A short comparison of the original data and the imputed data is given.

Usage

## S3 method for class 'imp'
summary(object, ...)

Arguments

object an object of class ‘imp’
... additional arguments passed trough

Details

Note that this function will be enhanced with more sophisticated methods in future versions of the package. It is very rudimental in its present form.

Value

None (invisible NULL).

Author(s)

Matthias Templ

See Also

impCoda, impKNNa
Examples

data(expenditures)
expenditures[1,3]
expenditures[1,3] <- NA
xi <- impKNNa(expenditures)
xi
summary(xi)
# plot(xi, which=1:2)

---

Description

Teaching stuff in selected countries

Format

A (tidy) data frame with 1216 observations on the following 4 variables.

- country  Country of origin
- subject  school type: primary, lower secondary, higher secondary and tertiary
- year  Year
- value  Number of stuff

Details

Teaching staff include professional personnel directly involved in teaching students, including classroom teachers, special education teachers and other teachers who work with students as a whole class, in small groups, or in one-to-one teaching. Teaching staff also include department chairs of whose duties include some teaching, but it does not include non-professional personnel who support teachers in providing instruction to students, such as teachers’ aides and other paraprofessional personnel. Academic staff include personnel whose primary assignment is instruction, research or public service, holding an academic rank with such titles as professor, associate professor, assistant professor, instructor, lecturer, or the equivalent of any of these academic ranks. The category includes personnel with other titles (e.g. dean, director, associate dean, assistant dean, chair or head of department), if their principal activity is instruction or research.

Author(s)

translated from https://data.oecd.org/ and restructured by Matthias Templ

Source

OECD: https://data.oecd.org/
ternaryDiag

References


Examples

data(teachingStuff)
str(teachingStuff)

ternaryDiag

Ternary diagram

Description

This plot shows the relative proportions of three variables (compositional parts) in one diagramm. Before plotting, the data are scaled.

Usage

ternaryDiag(x, name = colnames(x), text = NULL, grid = TRUE, gridCol = grey(0.6), mcex = 1.2, line = “none”, robust = TRUE, group = NULL, tol = 0.975, ...)

Arguments

x matrix or data.frame with 3 columns
name names of the variables
text default NULL, text for each point can be provided
grid if TRUE a grid is plotted additionally in the ternary diagram
gridCol color for the grid lines
mcex label size
line may be set to “none”, “pca”, “regression”, “regressionconf”, “regressionpred”, “ellipse”, “lda”
robust if line equals TRUE, it dedicates if a robust estimation is applied or not.
group if line equals “da”, it determines the grouping variable
tol if line equals “ellipse”, it determines the parameter for the tolerance ellipse
... further parameters, see, e.g., par()

Details

The relative proportions of each variable are plotted.

Author(s)

Peter Filzmoser <P.Filzmoser@tuwien.ac.at>, Matthias Templ
ternaryDiagAbline

Add a line to a ternary diagram.

description

A low-level plot function which adds a line to a high-level ternary diagram.

Usage

ternaryDiagAbline(x, ...)

Arguments

x Two-dimensional data set in isometric log-ratio transformed space.

... Additional graphical parameters passed through.

details

This is a small utility function which helps to add a line in a ternary plot from two given points in an isometric transformed space.
ternaryDiagEllipse

Value

no values are returned.

Author(s)

Matthias Templ

See Also

ternaryDiag

Examples

data(coffee)
x <- coffee[,2:4]
ternaryDiag(x, grid=FALSE)
ternaryDiagAbline(data.frame(z1=c(0.01,0.5), z2=c(0.4,0.8)), col="red")

Description

Low-level plot function which add tolerance ellipses to a high-level plot of a ternary diagram.

Usage

ternaryDiagEllipse(x, tolerance = c(0.9, 0.95, 0.975),
locscatt = "MCD", ...)

Arguments

x

tolerance

tlocscatt

... Additional arguments passed trough.

Value

no values are returned.

Author(s)

Peter Filzmoser, Matthias Templ
ternaryDiagPoints

add points or lines to a given ternary diagram.

Description

Low-level plot function to add points or lines to a ternary high-level plot.

Usage

ternaryDiagPoints(x, ...)

Arguments

x 
Three-dimensional composition given as an object of class "matrix" or "data.frame".

... 
Additional graphical parameters passed through.

Value

no values are returned.

Author(s)

Matthias Templ

References


See Also

ternaryDiag
Examples

data(coffee)
x <- coffee[,2:4]
ternaryDiag(x, grid=FALSE)
ternaryDiagPoints(x+1, col="red", pch=2)

trondelagC  regional geochemical survey of soil C in Norway

Description
A regional-scale geochemical survey of C horizon samples in Nord-Trondelag, Central Norway

Usage

data(trondelagC)

Format
A data frame with 754 observations and 70 variables

Details
- `X.S_ID` ID
- `X.Loc_ID` ID
- `longitude` longitude in WGS84
- `latitude` latitude in WGS84
- `E32wgs` UTM zone east
- `N32wgs` UTM zone north
- `X.Medium`
- `Ag` Concentration of silver (in mg/kg)
- `Al` Concentration of aluminum (in mg/kg)
- `As` Concentration of arsenic (in mg/kg)
- `Au` Concentration of gold (in mg/kg)
- `B` Concentration of boron (in mg/kg)
- `Ba` Concentration of barium (in mg/kg)
- `Be` Concentration of beryllium (in mg/kg)
- `Bi` Concentration of bismuth (in mg/kg)
- `Ca` Concentration of calzium (in mg/kg)
- `Cd` Concentration of cadmium (in mg/kg)
• Ce  Concentration of cerium (in mg/kg)
• Co  Concentration of cobalt (in mg/kg)
• Cr  Concentration of chromium (in mg/kg)
• Cs  Concentration of cesium (in mg/kg)
• Cu  Concentration of copper (in mg/kg)
• Fe  Concentration of iron (in mg/kg)
• Ga  Concentration of gallium (in mg/kg)
• Ge  Concentration of germanium (in mg/kg)
• Hf  Concentration of hafnium (in mg/kg)
• Hg  Concentration of mercury (in mg/kg)
• In  Concentration of indium (in mg/kg)
• K  Concentration of potassium (in mg/kg)
• La  Concentration of lanthanum (in mg/kg)
• Li  Concentration of lithium (in mg/kg)
• Mg  Concentration of magnesium (in mg/kg)
• Mn  Concentration of manganese (in mg/kg)
• Mo  Concentration of molybdenum (in mg/kg)
• Na  Concentration of sodium (in mg/kg)
• Nb  Concentration of niobium (in mg/kg)
• Ni  Concentration of nickel (in mg/kg)
• P  Concentration of phosphorus (in mg/kg)
• Pb  Concentration of lead (in mg/kg)
• Pb204 Concentration of lead, 204 neutrons (in mg/kg)
• Pb206 Concentration of lead, 206 neutrons (in mg/kg)
• Pb207 Concentration of lead, 207 neutrons (in mg/kg)
• Pb208 Concentration of lead, 208 neutrons (in mg/kg)
• X6_7Pb Concentration of lead (in mg/kg)
• X7_8Pb Concentration of lead (in mg/kg)
• X6_4Pb Concentration of lead (in mg/kg)
• X7_4Pb Concentration of lead (in mg/kg)
• X8_4Pb Concentration of lead (in mg/kg)
• Pd Concentration of palladium (in mg/kg)
• Pt  Concentration of platinum (in mg/kg)
• Rb  Concentration of rubidium (in mg/kg)
• Re  Concentration of rhenium (in mg/kg)
• S  Concentration of sulfur (in mg/kg)
• Sb  Concentration of antimony (in mg/kg)
- Sc  Concentration of scandium (in mg/kg)
- Se  Concentration of selenium (in mg/kg)
- Sn  Concentration of tin (in mg/kg)
- Sr  Concentration of strontium (in mg/kg)
- Ta  Concentration of tantalum (in mg/kg)
- Te  Concentration of tellurium (in mg/kg)
- Th  Concentration of thorium (in mg/kg)
- Ti  Concentration of titanium (in mg/kg)
- Tl  Concentration of thalium (in mg/kg)
- U   Concentration of uranium (in mg/kg)
- V   Concentration of vanadium (in mg/kg)
- W   Concentration of tungsten (in mg/kg)
- Y   Concentration of yttrium (in mg/kg)
- Zn  Concentration of zinc (in mg/kg)
- Zr  Concentration of zirconium (in mg/kg)

The samples were analysed using aqua regia extraction. Sampling was based on a 6.6km grid, i.e. 1 sample site/36 km².

**Author(s)**

NGU, [http://www.ngu.no](http://www.ngu.no), transferred to R by Matthias Templ <matthias.templ@tuwien.ac.at>

**References**


**Examples**

```r
data(trondelagO)
str(trondelagO)
```

**Description**

A regional-scale geochemical survey of O horizon samples in Nord-Trondelag, Central Norway

**Usage**

```r
data(trondelagO)
```
Format

A data frame with 754 observations and 70 variables

Details

- x.Nloc_id  ID
- litho  Rock type
- longitude  longitude in WGS84
- latitude  latitude in WGS84
- E32wgs  UTM zone east
- N32wgs  UTM zone north
- X.Medium  a numeric vector
- Alt_masl  a numeric vector
- LOI_480  Loss on ignition
- pH  Numeric scale used to specify the acidity or alkalinity of an aqueous solution
- Ag  Concentration of silver (in mg/kg)
- A1  Concentration of aluminum (in mg/kg)
- As  Concentration of arsenic (in mg/kg)
- Au  Concentration of gold (in mg/kg)
- B  Concentration of boron (in mg/kg)
- Ba  Concentration of barium (in mg/kg)
- Be  Concentration of beryllium (in mg/kg)
- Bi  Concentration of bismuth (in mg/kg)
- Ca  Concentration of calcium (in mg/kg)
- Cd  Concentration of cadmium (in mg/kg)
- Ce  Concentration of cerium (in mg/kg)
- Co  Concentration of cobalt (in mg/kg)
- Cr  Concentration of chromium (in mg/kg)
- Cs  Concentration of cesium (in mg/kg)
- Cu  Concentration of copper (in mg/kg)
- Fe  Concentration of iron (in mg/kg)
- Ga  Concentration of gallium (in mg/kg)
- Ge  Concentration of germanium (in mg/kg)
- Hf  Concentration of hafnium (in mg/kg)
- Hg  Concentration of mercury (in mg/kg)
- In  Concentration of indium (in mg/kg)
- K  Concentration of pottasium (in mg/kg)
- La  Concentration of lanthanum (in mg/kg)
• Li Concentration of lithium (in mg/kg)
• Mg Concentration of magnesium (in mg/kg)
• Mn Concentration of manganese (in mg/kg)
• Mo Concentration of molybdenum (in mg/kg)
• Na Concentration of sodium (in mg/kg)
• Nb Concentration of niobium (in mg/kg)
• Ni Concentration of nickel (in mg/kg)
• P Concentration of phosphorus (in mg/kg)
• Pb Concentration of lead (in mg/kg)
• Pb204 Concentration of lead, 204 neutrons (in mg/kg)
• Pb206 Concentration of lead, 206 neutrons (in mg/kg)
• Pb207 Concentration of lead, 207 neutrons (in mg/kg)
• Pb208 Concentration of lead, 208 neutrons (in mg/kg)
• X6_7Pb Concentration of lead (in mg/kg)
• X7_8Pb Concentration of lead (in mg/kg)
• X6_4Pb Concentration of lead (in mg/kg)
• X7_4Pb Concentration of lead (in mg/kg)
• X8_4Pb Concentration of lead (in mg/kg)
• Pd Concentration of palladium (in mg/kg)
• Pt Concentration of platium (in mg/kg)
• Rb Concentration of rubidium (in mg/kg)
• Re Concentration of rhenium (in mg/kg)
• S Concentration of sulfur (in mg/kg)
• Sb Concentration of antimony (in mg/kg)
• Sc Concentration of scandium (in mg/kg)
• Se Concentration of selenium (in mg/kg)
• Sn Concentration of tin (in mg/kg)
• Sr Concentration of strontium (in mg/kg)
• Ta Concentration of tantalum (in mg/kg)
• Te Concentration of tellurium (in mg/kg)
• Th Concentration of thorium (in mg/kg)
• Ti Concentration of titanium (in mg/kg)
• Tl Concentration of thalium (in mg/kg)
• U Concentration of uranium (in mg/kg)
• V Concentration of vanadium (in mg/kg)
• W Concentration of tungsten (in mg/kg)
• Y Concentration of yttrium (in mg/kg)
• Zn Concentration of zinc (in mg/kg)
• Zr Concentration of zirconium (in mg/kg)

The samples were analysed using aqua regia extraction. Sampling was based on a 6.6km grid, i.e. 1 sample site/36 km².
Author(s)

NGU, http://www.ngu.no, transferred to R by Matthias Templ <matthias.templ@tuwien.ac.at>

References


Examples

data(trondelago)
str(trondelago)

unemployed unemployed of young people

Description

Youth not in employment, education or training (NEET) in 43 countries from 1997 till 2015

Format

A (tidy) data frame with 1216 observations on the following 4 variables.

- country Country of origin
- age age group
- year Year
- value percentage of unemployed

Details

This indicator presents the share of young people who are not in employment, education or training (NEET), as a percentage of the total number of young people in the corresponding age group, by gender. Young people in education include those attending part-time or full-time education, but exclude those in non-formal education and in educational activities of very short duration. Employment is defined according to the OECD/ILO Guidelines and covers all those who have been in paid work for at least one hour in the reference week of the survey or were temporarily absent from such work. Therefore NEET youth can be either unemployed or inactive and not involved in education or training. Young people who are neither in employment nor in education or training are at risk of becoming socially excluded - individuals with income below the poverty-line and lacking the skills to improve their economic situation.

Author(s)

translated from https://data.oecd.org/ and restructured by Matthias Templ
variation

Source

OECD: https://data.oecd.org/

References


Examples

data(unemployed)
str(unemployed)

```
variation           Robust and classical variation matrix
```

Description

Estimates the variation matrix with robust methods.

Usage

variation(x, robust = TRUE)

Arguments

x data frame or matrix with positive entries
robust if FALSE, standard measures are used.

Details

The variation matrix is estimated for a given compositional data set. Instead of using the classical standard deviations the mad is used when parameter robust is set to TRUE.

Value

The (robust) variation matrix.

Author(s)

Matthias Templ

References

Examples

data(expenditures)
variation(expenditures)
variation(expenditures, robust=FALSE)

zeroOut  Detection of outliers of zero-inflated data

Description

detects outliers in compositional zero-inflated data

Usage

zeroOut(x, impute = "knn")

Arguments

x  a data frame
impute  imputation method internally used

Details

XXX

Value

XXX

Author(s)

Matthias Templ

Examples

### Installing and loading required packages
data(expenditures)
Index

*Topic **aplot**
  - biplot.factanal, 22
  - biplot.pcaCoDa, 23
  - plot.imp, 102
  - plot.pcaCoDa, 105
  - ternaryDiag, 119
  - ternaryDiagAbleine, 120
  - ternaryDiagEllipse, 121
  - ternaryDiagPoints, 122

*Topic **arith**
  - adist, 9
  - perturbation, 96

*Topic **datasets**
  - cancer, 25
  - cancerMN, 26
  - chorizonDL, 30
  - expendituresEU, 50
  - gjovik, 54
  - goverxp, 58
  - haplogroups, 59
  - laborForce, 79
  - lifeExpGdp, 80
  - phd, 99
  - phd_totals, 100
  - teachingStuff, 118
  - unemployed, 128

*Topic **data**
  - ageCatWorld, 15
  - alcohol, 16
  - alcoholreg, 17
  - arcticLake, 17
  - coffee, 35
  - economy, 44
  - educFM, 45
  - election, 46
  - electionATbp, 47
  - employment, 48
  - employment.df, 48
  - expenditures, 49
  - GDPsatis, 51
  - gemas, 52
  - instw, 75
  - isic32, 79
  - machineOperators, 82
  - mcad, 83
  - mortality, 85
  - mortality_tab, 86
  - nutrients, 87
  - nutrients_branded, 89
  - payments, 93
  - precipitation, 106
  - production, 108
  - rcodes, 110
  - skyelavas, 114
  - socExp, 115
  - trondelagC, 123
  - trondelagD, 125

*Topic **hplot**
  - plot.imp, 102

*Topic **htest**
  - adtest, 12
  - adtestWrapper, 13
  - rSDev.test, 112

*Topic **iteration**
  - impCoda, 62

*Topic **manip**
  - addLR, 6
  - addLRinv, 8
  - adjust, 11
  - bootnComp, 24
  - ced, 27
  - cenLR, 28
  - cenLRinv, 29
  - constSum, 37
  - impAll, 61
  - impKNNa, 63
  - impRZalr, 65
  - impRZilr, 67
imputeBDLs, 68
imputeUDLs, 71
orthbasis, 90
rdcm, 110
*Topic math
aDist, 9
perturbation, 96
pivotCoord, 101
*Topic models
lmCoDaX, 81
*Topic multivariate
clustCoDa, 32
clustCoDa_qmode, 34
daCoDa, 40
dafisher, 41
impCoDa, 62
impKNNa, 63
impRZalr, 65
impRZilr, 67
imputeBDLs, 68
imputeUDLs, 71
missPatterns, 84
outCoDa, 91
pcaCoDa, 94
pfa, 97
ternaryDiag, 119
variation, 129
*Topic package
robCompositions-package, 4
*Topic print
print.imp, 107
summary.imp, 117
*Topic robust
impCoDa, 62
variation, 129
addLR, 6, 9, 29
addLRinv, 7, 8, 29
aDist, 9
adjust, 11
adjustImputed/imputeBDLs, 68
adtest, 12, 15
adtestWrapper, 13, 13
ageCatWorld, 15
aggr, 84
alcohol, 16
alcoholreg, 17
arcticLake, 17
balances, 18
balZav, 20
biomarker, 20
biplot.factanal, 22
biplot.pcaCoDa, 23, 95, 105
bootnComp, 24
cancer, 25
cancerMN, 26
ced, 27
cenLR, 9, 28, 29, 91
cenLRinv, 9, 29, 29
checkData/imputeBDLs, 68
chiorizon, 30, 32
chiorizonDL, 30, 30, 32
clustCoDa, 32
cor, 39
corCoDa, 39
daCoDa, 40
dafisher, 41
economy, 44
educFM, 45
election, 46
electionATbp, 47
employment, 48
employment_df, 48
expenditures, 49
expendituresEU, 50
GDPsatis, 51
gemas, 52
ggetEstimates/compareMahal, 36
gjovik, 54
gm, 56
gmean/gmean_sum, 57
Gmean_sum, 57
govexp, 58
haplogroups, 59
ilr.2x2, 60
ilrRegression/lmCoDaX, 81
impAll, 61
<table>
<thead>
<tr>
<th>Function/Module</th>
<th>Page Numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>impCoda, 11, 61, 62, 65, 104, 107, 117</td>
<td>113</td>
</tr>
<tr>
<td>impKNNa, 36, 63, 63, 104, 107, 117</td>
<td>113</td>
</tr>
<tr>
<td>impRZair, 65, 68</td>
<td>113</td>
</tr>
<tr>
<td>impRZilr, 25, 61, 66, 67</td>
<td>113</td>
</tr>
<tr>
<td>imputeBDLs, 68, 70, 73</td>
<td>113</td>
</tr>
<tr>
<td>imputeUDLs, 71</td>
<td>113</td>
</tr>
<tr>
<td>ind2x2, 73</td>
<td>113</td>
</tr>
<tr>
<td>indTab, 74</td>
<td>113</td>
</tr>
<tr>
<td>int2x2, 76</td>
<td>113</td>
</tr>
<tr>
<td>intArray, 77</td>
<td>113</td>
</tr>
<tr>
<td>intTab, 77, 78</td>
<td>113</td>
</tr>
<tr>
<td>iprod(dist), 9</td>
<td>113</td>
</tr>
<tr>
<td>isic32, 79</td>
<td>113</td>
</tr>
<tr>
<td>isomLR (pivotCoord), 101</td>
<td>113</td>
</tr>
<tr>
<td>isomLRinv (pivotCoord), 101</td>
<td>113</td>
</tr>
<tr>
<td>isomLRinvp (pivotCoord), 101</td>
<td>113</td>
</tr>
<tr>
<td>isomLRp (pivotCoord), 101</td>
<td>113</td>
</tr>
<tr>
<td>laborForce, 79</td>
<td>113</td>
</tr>
<tr>
<td>LdaClassic, 41</td>
<td>113</td>
</tr>
<tr>
<td>lifeExpGdp, 80</td>
<td>113</td>
</tr>
<tr>
<td>Linda, 41, 43</td>
<td>113</td>
</tr>
<tr>
<td>lm, 82</td>
<td>113</td>
</tr>
<tr>
<td>lmCoDaX, 81</td>
<td>113</td>
</tr>
<tr>
<td>machineOperators, 82</td>
<td>113</td>
</tr>
<tr>
<td>mad, 129</td>
<td>113</td>
</tr>
<tr>
<td>mcad, 83</td>
<td>113</td>
</tr>
<tr>
<td>missPatterns, 84</td>
<td>113</td>
</tr>
<tr>
<td>mortality, 85</td>
<td>113</td>
</tr>
<tr>
<td>mortality_tab, 86</td>
<td>113</td>
</tr>
<tr>
<td>nutrients, 87</td>
<td>113</td>
</tr>
<tr>
<td>nutrients_branded, 89</td>
<td>113</td>
</tr>
<tr>
<td>orthbasis, 90</td>
<td>113</td>
</tr>
<tr>
<td>outCoDa, 91</td>
<td>113</td>
</tr>
<tr>
<td>par, 103</td>
<td>113</td>
</tr>
<tr>
<td>payments, 93</td>
<td>113</td>
</tr>
<tr>
<td>pcaCoDa, 23, 24, 94, 105</td>
<td>113</td>
</tr>
<tr>
<td>perturbation, 96</td>
<td>113</td>
</tr>
<tr>
<td>pfa, 23, 97</td>
<td>113</td>
</tr>
<tr>
<td>phd, 99</td>
<td>113</td>
</tr>
<tr>
<td>phd_totals, 100</td>
<td>113</td>
</tr>
<tr>
<td>pivotCoord, 7, 10, 15, 29, 36, 63, 91, 92, 101</td>
<td>113</td>
</tr>
<tr>
<td>pivotCoordInv, 9, 29</td>
<td>113</td>
</tr>
<tr>
<td>pivotCoordInv (pivotCoord), 101</td>
<td>113</td>
</tr>
<tr>
<td>plot.biomarker, 21</td>
<td>113</td>
</tr>
<tr>
<td>plot.biomarker(biomarker), 20</td>
<td>113</td>
</tr>
<tr>
<td>plot.clustCoDa (clustCoDa), 32</td>
<td>113</td>
</tr>
<tr>
<td>plot.clustCoDa_qmode (clustCoDa_qmode), 34</td>
<td>113</td>
</tr>
<tr>
<td>plot.estimates (compareMahal), 36</td>
<td>113</td>
</tr>
<tr>
<td>plot.imp, 102</td>
<td>113</td>
</tr>
<tr>
<td>plot.mahal (compareMahal), 36</td>
<td>113</td>
</tr>
<tr>
<td>plot.outCoDa (outCoDa), 91</td>
<td>113</td>
</tr>
<tr>
<td>powering (perturbation), 96</td>
<td>113</td>
</tr>
<tr>
<td>precipitation, 106</td>
<td>113</td>
</tr>
<tr>
<td>predict.dafisher (dafisher), 41</td>
<td>113</td>
</tr>
<tr>
<td>print.adtestWrapper (adtestWrapper), 13</td>
<td>113</td>
</tr>
<tr>
<td>print.biomarker (biomarker), 20</td>
<td>113</td>
</tr>
<tr>
<td>print.coord (coord), 38</td>
<td>113</td>
</tr>
<tr>
<td>print.dafisher (dafisher), 41</td>
<td>113</td>
</tr>
<tr>
<td>print.estimates (compareMahal), 36</td>
<td>113</td>
</tr>
<tr>
<td>print.imp, 107</td>
<td>113</td>
</tr>
<tr>
<td>print.outCoDa (outCoDa), 91</td>
<td>113</td>
</tr>
<tr>
<td>print.pcaCoDa, 95</td>
<td>113</td>
</tr>
<tr>
<td>print.pcaCoDa (pcaCoDa), 94</td>
<td>113</td>
</tr>
<tr>
<td>print.replaced (imputeBDLs), 68</td>
<td>113</td>
</tr>
<tr>
<td>production, 108</td>
<td>113</td>
</tr>
<tr>
<td>pTab, 109</td>
<td>113</td>
</tr>
<tr>
<td>QdaClassic, 41</td>
<td>113</td>
</tr>
<tr>
<td>QdaCov, 41</td>
<td>113</td>
</tr>
<tr>
<td>rcodes, 110</td>
<td>113</td>
</tr>
<tr>
<td>rdcm, 27, 110, 111</td>
<td>113</td>
</tr>
<tr>
<td>robCompositions</td>
<td>113</td>
</tr>
<tr>
<td>(robCompositions-package), 4</td>
<td>113</td>
</tr>
<tr>
<td>robCompositions-package, 4</td>
<td>113</td>
</tr>
<tr>
<td>robilrregression (lmCoDaX), 81</td>
<td>113</td>
</tr>
<tr>
<td>RSDev, 111, 113</td>
<td>113</td>
</tr>
<tr>
<td>RSDev.test, 112</td>
<td>113</td>
</tr>
<tr>
<td>SDev, 113</td>
<td>113</td>
</tr>
<tr>
<td>skyeLavas, 114</td>
<td>113</td>
</tr>
<tr>
<td>socExp, 115</td>
<td>113</td>
</tr>
<tr>
<td>stats, 116</td>
<td>113</td>
</tr>
<tr>
<td>summary.adtestWrapper (adtestWrapper), 13</td>
<td>113</td>
</tr>
<tr>
<td>summary.biomarker (biomarker), 20</td>
<td>113</td>
</tr>
<tr>
<td>summary.dafisher (dafisher), 41</td>
<td>113</td>
</tr>
<tr>
<td>summary.imp, 117</td>
<td>113</td>
</tr>
<tr>
<td>summary.pcaCoDa, 95</td>
<td>113</td>
</tr>
<tr>
<td>summary.pcaCoDa (pcaCoDa), 94</td>
<td>113</td>
</tr>
</tbody>
</table>
teachingStuff, 118
ternary, 120
ternaryDiag, 119, 121, 122
ternaryDiagAbline, 120
ternaryDiagEllipse, 121
ternaryDiagLines (ternaryDiagPoints), 122
ternaryDiagPoints, 122
trondelagC, 123
trondelagO, 125
unemployed, 128
variation, 129
zeroOut, 130
zeroPatterns (missPatterns), 84