

Package: nexus (via r-universe)

October 23, 2024

Title Sourcing Archaeological Materials by Chemical Composition

Version 0.3.0.9000

Maintainer Nicolas Frerebeau

<nicolas.frerebeau@u-bordeaux-montaigne.fr>

Description Exploration and analysis of compositional data in the framework of Aitchison (1986, ISBN: 978-94-010-8324-9). This package provides tools for chemical fingerprinting and source tracking of ancient materials.

License GPL (>= 3)

URL <https://packages.tesselle.org/nexus/>,
<https://github.com/tesselle/nexus>

BugReports <https://github.com/tesselle/nexus/issues>

Depends R (>= 3.5), dimensio (>= 0.9.0)

Imports arkhe (>= 1.7.0), graphics, grDevices, isopleuros (>= 1.2.0), khroma (>= 1.14.0), methods, MASS, stats, utils

Suggests cluster, folio (>= 1.5.0), igraph, knitr, markdown, rsvg, svglite, tinysnapshot, tinytest

VignetteBuilder knitr

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Collate 'AllClasses.R' 'AllGenerics.R' 'aggregate.R' 'barplot.R'
'chemistry.R' 'coerce.R' 'condense.R' 'covariance.R' 'data.R'
'describe.R' 'dist.R' 'graph.R' 'group.R' 'hist.R' 'margin.R'
'mean.R' 'mix.R' 'mutators.R' 'nexus-internal.R'
'nexus-package.R' 'outliers.R' 'pca.R' 'plot.R' 'quantile.R'
'reexport.R' 'replace.R' 'scale.R' 'show.R' 'simplex.R'
'split.R' 'subset.R' 'summary.R' 'transform_alr.R'
'transform_clr.R' 'transform_ilr.R' 'transform_inverse.R'

'transform_lr.R' 'transform_plr.R' 'validate.R' 'variance.R'
 'variation.R' 'zzz.R'

Repository <https://tesselle.r-universe.dev>

RemoteUrl <https://github.com/tesselle/nexus>

RemoteRef HEAD

RemoteSha 7d0e4e08fa8c84b62db811f7628beed8e6d23040

Contents

aggregate	3
arctic	4
arithmetic	5
as_amounts	6
as_composition	7
as_graph	8
barplot	10
bind	12
boxite	13
chemistry	14
closure	15
condense	16
covariance	17
coxite	19
describe	20
detect_outlier	20
dist	23
extract	24
groups	25
hist	27
hongite	28
kongite	29
lava	30
mahalanobis	30
margin	31
mean	32
missing	34
mix	35
mutators	36
pca	37
perturbation	39
pip	40
plot	41
plot_logratio	43
plot_outlier	44
powering	47
predator	48

quantile	49
replace_NA	50
replace_zero	51
scalar	52
scale	53
slides	54
split	55
subset	56
totals	58
transform_alr	59
transform_clr	61
transform_ilm	62
transform_inverse	63
transform_lr	65
transform_plr	66
variance	67
variance_total	69
variation	71
Index	73

 aggregate

Compute Summary Statistics of Data Subsets

Description

Splits the data into subsets, computes summary statistics for each, and returns the result.

Usage

```
## S4 method for signature 'CompositionMatrix'
aggregate(x, by, FUN, ..., simplify = TRUE, drop = TRUE)
```

Arguments

x	A CompositionMatrix object.
by	A vector or a list of grouping elements, each as long as the variables in x. The elements are coerced to factors before use (in the sense that interaction(by) defines the grouping).
FUN	A function to compute the summary statistics.
...	Further arguments to be passed to FUN.
simplify	A logical scalar: should the results be simplified to a matrix if possible?
drop	A logical scalar indicating whether to drop unused combinations of grouping values.

Value

A [matrix](#).

Author(s)

N. Frerebeau

See Also

Other statistics: [condense\(\)](#), [covariance\(\)](#), [dist](#), [mahalanobis\(\)](#), [margin\(\)](#), [mean\(\)](#), [pip\(\)](#), [quantile\(\)](#), [scale\(\)](#), [variance\(\)](#), [variance_total\(\)](#), [variation\(\)](#)

Examples

```
## Data from Aitchison 1986
data("slides")

## Coerce to a compositional matrix
coda <- as_composition(slides)

## Compositional mean by slide
aggregate(coda, by = slides$slide, FUN = mean)

## Metric variance by slide
aggregate(coda, by = slides$slide, FUN = variance_total)
```

arctic

Arctic Lake

Description

Sand, silt, clay compositions of 39 sediment samples at different water depths in an Arctic lake.

Usage

```
arctic
```

Format

A [data.frame](#) with 4 variables:

sand Sand content (percent).

silt Silt content (percent).

clay Clay content (percent).

depth Water depth (m).

References

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. London: Chapman and Hall.

See Also

Other datasets: [boxite](#), [coxite](#), [hongite](#), [kongite](#), [lava](#), [predator](#), [slides](#)

arithmetic

Operations in the Simplex

Description

Operators performing operations in the simplex.

Usage

```
x %perturbe% y
```

```
x %power% y
```

```
## S4 method for signature 'CompositionMatrix,CompositionMatrix'
```

```
x %perturbe% y
```

```
## S4 method for signature 'CompositionMatrix,numeric'
```

```
x %power% y
```

```
## S4 method for signature 'numeric,CompositionMatrix'
```

```
x %power% y
```

Arguments

x A [CompositionMatrix](#) object.

y A [CompositionMatrix](#) object or a [numeric](#) vector.

Details

%perturbe% [Perturbation operation](#).

%power% [Powering operation](#).

Value

A [CompositionMatrix](#) object or a [numeric](#) vector (same as x).

Author(s)

N. Frerebeau

See Also

Other operations in the simplex: [closure\(\)](#), [perturbation\(\)](#), [powering\(\)](#), [scalar\(\)](#)

Examples

```
x <- as_composition(c(1, 2, 3))
y <- as_composition(c(1, 2, 1))

## Perturbation
perturbation(x, y)
x + y

## Powering
powering(y, 2)
y * 2

## Scalar product
scalar(x, y)
```

as_amounts

Coerce to Amounts

Description

Coerce to Amounts

Usage

```
as_amounts(from, ...)
```

S4 method for signature 'CompositionMatrix'
as_amounts(from)

Arguments

from A [CompositionMatrix](#) object.
... Currently not used.

Value

A [numeric matrix](#).

Author(s)

N. Frerebeau

See Also

Other compositional data tools: [as_composition\(\)](#)

Examples

```
## Create a count matrix
A1 <- matrix(data = sample(1:100, 100, TRUE), nrow = 20)

## Coerce to compositions
B <- as_composition(A1)

## Row sums are internally stored before coercing to relative frequencies
totals(B)

## This allows to restore the source data
A2 <- as_amounts(B)

## Coerce to a data.frame
X <- data.frame(B)
head(X)
```

as_composition	<i>Coerce to a Closed Compositional Matrix</i>
----------------	--

Description

Coerces an object to a `CompositionMatrix` object.

Usage

```
as_composition(from, ...)

## S4 method for signature 'numeric'
as_composition(from)

## S4 method for signature 'matrix'
as_composition(from)

## S4 method for signature 'data.frame'
as_composition(
  from,
  parts = NULL,
  groups = NULL,
  verbose = getOption("nexus.verbose")
)
```

Arguments

from	A matrix or data.frame to be coerced.
...	Currently not used.
parts	A vector giving the index of the column to be used a compositional parts. If NULL (the default), all double columns will be used.

groups An [integer](#) giving the index of the column to be used to group the samples. If NULL (the default), no grouping is stored.

verbose A [logical](#) scalar: should R report extra information on progress?

Details

See `vignette("nexus")`.

Value

A [CompositionMatrix](#) object.

Author(s)

N. Frerebeau

See Also

Other compositional data tools: [as_amounts\(\)](#)

Examples

```
## Create a count matrix
A1 <- matrix(data = sample(1:100, 100, TRUE), nrow = 20)

## Coerce to compositions
B <- as_composition(A1)

## Row sums are internally stored before coercing to relative frequencies
totals(B)

## This allows to restore the source data
A2 <- as_amounts(B)

## Coerce to a data.frame
X <- data.frame(B)
head(X)
```

as_graph

Graph of Log-ratios

Description

Produces a graph of log-ratios.

Usage

```
as_graph(object, ...)  
  
## S4 method for signature 'LR'  
as_graph(object)  
  
## S4 method for signature 'ALR'  
as_graph(object)  
  
## S4 method for signature 'ILR'  
as_graph(object)
```

Arguments

object	A LogRatio object.
...	Currently not used.

Value

An **igraph** graph object.

Author(s)

N. Frerebeau

See Also

Other plot methods: [barplot\(\)](#), [hist\(\)](#), [plot\(\)](#), [plot_logratio](#)

Examples

```
if (requireNamespace("igraph", quietly = TRUE)) {  
  
  library(igraph)  
  
  ## Data from Aitchison 1986  
  data("hongite")  
  
  ## Coerce to compositional data  
  coda <- as_composition(hongite)  
  
  ## Pairwise log-ratio  
  lr <- transform_lr(coda)  
  lr_graph <- as_graph(lr)  
  plot(lr_graph)  
  
  ## Additive log-ratio  
  alr <- transform_alr(coda)  
  alr_graph <- as_graph(alr)  
  plot(alr_graph)
```

```
## Isometric log-ratio
ilr <- transform_ilr(coda)
ilr_graph <- as_graph(ilr)
plot(ilr_graph)

plr <- transform_plr(coda)
plr_graph <- as_graph(plr)
plot(plr_graph)

}
```

barplot

Barplot of Compositional Data

Description

Displays a compositional bar chart.

Usage

```
## S4 method for signature 'CompositionMatrix'
barplot(
  height,
  ...,
  by = groups(height),
  order_columns = FALSE,
  order_rows = NULL,
  decreasing = TRUE,
  space = 0.2,
  offset = 0.025,
  color = palette_color_discrete(),
  border = NA,
  axes = TRUE,
  legend = TRUE
)
```

Arguments

height	A CompositionMatrix object.
...	Further graphical parameters.
by	A vector of grouping elements, as long as the variables in height.
order_columns	A logical scalar: should columns be reordered?
order_rows	An integer vector giving the index of the column to be used for the ordering of the data.
decreasing	A logical scalar: should the sort order of rows be increasing or decreasing?

space	A length-one numeric vector giving the the amount of space (as a fraction of the width of a bar) left between each bar (defaults to 0.2).
offset	A length-one numeric vector giving the the amount of space (as a fraction) left between groups (defaults to 0.025). Only used if groups is not NULL.
color	A palette function that when called with a single argument returns a character vector of colors.
border	The color to draw the borders.
axes	A logical scalar: should axes be drawn on the plot?
legend	A logical scalar: should the legend be displayed?

Value

barplot() is called for its side-effects: is results in a graphic being displayed (invisibly return height).

Author(s)

N. Frerebeau

See Also

Other plot methods: [as_graph\(\)](#), [hist\(\)](#), [plot\(\)](#), [plot_logratio](#)

Examples

```
## Data from Aitchison 1986
data("hongite")

## Coerce to compositional data
coda <- as_composition(hongite)

## Bar plot
barplot(coda)

## Data from Day et al. 2011
data("kommos", package = "folio")
kommos <- remove_NA(kommos, margin = 1) # Remove cases with missing values
coda <- as_composition(kommos, groups = 1) # Coerce to compositional data

## Use ceramic types for grouping
barplot(coda, order_columns = TRUE)

## Display only minor elements
minor <- coda[, is_element_minor(coda)]
barplot(minor, order_columns = TRUE)
```

bind*Combine Two Composition Matrices*

Description

Combine Two Composition Matrices

Usage

```
## S4 method for signature 'CompositionMatrix,CompositionMatrix'  
rbind2(x, y)
```

Arguments

x, y A [CompositionMatrix](#) object.

Details

rbind2() combine by rows.

Value

A [CompositionMatrix](#) objects.

Author(s)

N. Frerebeau

See Also

Other subsetting methods: [extract\(\)](#), [split\(\)](#), [subset\(\)](#)

Examples

```
## Create a data.frame  
X <- data.frame(  
  samples = c("A", "A", "A", "B", "B", "B", "C", "C", "C"),  
  groups = c("X", "X", "X", "X", NA, NA, "Y", "Y", "Y"),  
  Ca = c(7.72, 7.32, 3.11, 7.19, 7.41, 5, 4.18, 1, 4.51),  
  Fe = c(6.12, 5.88, 5.12, 6.18, 6.02, 7.14, 5.25, 5.28, 5.72),  
  Na = c(0.97, 1.59, 1.25, 0.86, 0.76, 0.51, 0.75, 0.52, 0.56)  
)  
  
## Coerce to a compositional matrix  
Y <- as_composition(X)  
  
## Split by group  
## !/\ Unassigned samples are discarded ! !/  
(s1 <- split(Y, f = X$groups))
```

```
## Split by group
## Keep unassigned samples, see help(factor)
(s2 <- split(Y, f = factor(X$groups, exclude = NULL)))

## Bind by rows
do.call(rbind, s2)
```

boxite

Boxite Mineralogy

Description

Mineral compositions and depths of 25 specimens of boxite.

Usage

boxite

Format

A [data.frame](#) with 5 variables (minerals):

- A Albite (weight percent).
 - B Blandite (weight percent).
 - C Cornite (weight percent).
 - D Daubite (weight percent).
 - E Endite (weight percent).
- depth Depth (meter).

References

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. London: Chapman and Hall.

See Also

Other datasets: [arctic](#), [coxite](#), [hongite](#), [kongite](#), [lava](#), [predator](#), [slides](#)

Description

Identify oxides and major, minor and traces elements in a compositional data matrix.

Usage

```
is_element_major(object, ...)  
  
is_element_minor(object, ...)  
  
is_element_trace(object, ...)  
  
is_oxide(object, ...)  
  
## S4 method for signature 'character'  
is_oxide(object)  
  
## S4 method for signature 'CompositionMatrix'  
is_oxide(object)  
  
## S4 method for signature 'CompositionMatrix'  
is_element_major(object, min = 1/100, max = Inf)  
  
## S4 method for signature 'CompositionMatrix'  
is_element_minor(object, min = 0.1/100, max = 1/100)  
  
## S4 method for signature 'CompositionMatrix'  
is_element_trace(object, min = -Inf, max = 0.1/100)
```

Arguments

object	A CompositionMatrix object.
...	Currently not used.
min	A length-one numeric vector specifying the lower bound for element identification.
max	A length-one numeric vector specifying the upper bound for element identification.

Details

There is no definite classification of what are the major, minor and trace elements are. By default, the following rule of thumb is used:

major elements The major elements are those that define the material under study. Major elements usually have concentrations of above 1%.

minor elements Minor elements usually have concentrations between 1% and 0.1%

trace elements Trace elements usually have concentrations of less than 0.1%.

Value

A [logical](#) vector.

Note

`is_oxide()` uses a regular expression (it does not check if elements exist or if stoichiometry is valid).

Author(s)

N. Frerebeau

Examples

```
## Data from Day et al. 2011
data("kommos", package = "folio") # Coerce to compositional data
kommos <- remove_NA(kommos, margin = 1) # Remove cases with missing values
coda <- as_composition(kommos, groups = 1) # Use ceramic types for grouping

is_element_major(coda)
is_element_minor(coda)
is_element_trace(coda)

is_oxide(coda)
```

closure

Closure Operation

Description

Closes compositions to sum to 1.

Usage

```
closure(x, ...)
```

```
## S4 method for signature 'numeric'
closure(x, total = 1, na.rm = TRUE)
```

```
## S4 method for signature 'matrix'
closure(x, total = 1, na.rm = TRUE)
```

Arguments

<code>x</code>	A numeric vector or matrix.
<code>...</code>	Currently not used.
<code>total</code>	A numeric vector specifying the total amount to which the compositions should be closed (defaults to 1).
<code>na.rm</code>	A logical scalar: should missing values be removed?

Value

A [numeric](#) vector or matrix (same as `x`).

Author(s)

N. Frerebeau

See Also

Other operations in the simplex: [arithmetic](#), [perturbation\(\)](#), [powering\(\)](#), [scalar\(\)](#)

Examples

```
x <- as_composition(c(1, 2, 3))
y <- as_composition(c(1, 2, 1))

## Perturbation
perturbation(x, y)
x + y

## Powering
powering(y, 2)
y * 2

## Scalar product
scalar(x, y)
```

condense

Compositional Mean of Data Subsets

Description

Splits the data into subsets and computes compositional mean for each.

Usage

```
condense(x, ...)
```

S4 method for signature 'CompositionMatrix'

```
condense(x, by = groups(x), ...)
```


Arguments

<code>x</code>	A <code>CompositionMatrix</code> object.
<code>...</code>	Further arguments to be passed to <code>mean()</code> .
<code>by</code>	A vector or a list of grouping elements, each as long as the variables in <code>x</code> . The elements are coerced to factors before use (in the sense that <code>interaction(by)</code> defines the grouping).

Value

A `CompositionMatrix` object.

Author(s)

N. Frerebeau

See Also

`mean()`, `aggregate()`

Other statistics: `aggregate()`, `covariance()`, `dist`, `mahalanobis()`, `margin()`, `mean()`, `pip()`, `quantile()`, `scale()`, `variance()`, `variance_total()`, `variation()`

Examples

```
## Data from Aitchison 1986
data("slides")

## Coerce to a compositional matrix
coda <- as_composition(slides, groups = 2)

## Compositional mean by group
condense(coda)
```

covariance

Covariance Matrix

Description

Computes the (centered) log-ratio covariance matrix (see below).

Usage

```
covariance(x, ...)
```

S4 method for signature 'CompositionMatrix'
 covariance(x, center = TRUE, method = "pearson")

S4 method for signature 'ALR'

```
covariance(x, method = "pearson")

## S4 method for signature 'CLR'
covariance(x, method = "pearson")
```

Arguments

x	A CompositionMatrix object.
...	Currently not used.
center	A logical scalar: should the <i>centered</i> log-ratio covariance matrix be computed?
method	A character string indicating which covariance is to be computed (see stats::cov()).

Value

A [matrix](#).

Methods (by class)

- `covariance(ALR)`: Computes the log-ratio covariance matrix (Aitchison 1986, definition 4.5).
- `covariance(CLR)`: Computes the centered log-ratio covariance matrix (Aitchison 1986, definition 4.6).

Author(s)

N. Frerebeau

References

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. London: Chapman and Hall, p. 64-91.

Greenacre, M. J. (2019). *Compositional Data Analysis in Practice*. Boca Raton: CRC Press.

See Also

Other statistics: [aggregate\(\)](#), [condense\(\)](#), [dist](#), [mahalanobis\(\)](#), [margin\(\)](#), [mean\(\)](#), [pip\(\)](#), [quantile\(\)](#), [scale\(\)](#), [variance\(\)](#), [variance_total\(\)](#), [variation\(\)](#)

Examples

```
## Data from Aitchison 1986
data("hongite")

## Coerce to compositional data
coda <- as_composition(hongite)

## Log-ratio covariance matrix
## (Aitchison 1986, definition 4.5)
covariance(coda, center = FALSE)
```

```
## Centered log-ratio covariance matrix
## (Aitchison 1986, definition 4.6)
covariance(coda, center = TRUE)
```

coxite	<i>Coxite Mineralogy</i>
--------	--------------------------

Description

Mineral compositions, depths and porosity of 25 specimens of coxite.

Usage

```
coxite
```

Format

A [data.frame](#) with 5 variables (minerals):

A Albite (weight percent).

B Blandite (weight percent).

C Cornite (weight percent).

D Daubite (weight percent).

E Endite (weight percent).

depth Depth (meter).

porosity Porosity (percent).

References

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. London: Chapman and Hall.

See Also

Other datasets: [arctic](#), [boxite](#), [hongite](#), [kongite](#), [lava](#), [predator](#), [slides](#)

describe	<i>Data Description</i>
----------	-------------------------

Description

Describes an object.

Usage

```
## S4 method for signature 'CompositionMatrix'  
describe(x)
```

Arguments

x A *CompositionMatrix* object.

Value

describe() is called for its side-effects. Invisibly returns x.

Author(s)

N. Frerebeau

Examples

```
## Data from Aitchison 1986  
data("slides")  
  
## Coerce to compositional data  
coda <- as_composition(slides, groups = 2)  
  
## Quick description  
describe(coda)
```

detect_outlier	<i>Outlier Detection</i>
----------------	--------------------------

Description

Outlier Detection

Usage

```

detect_outlier(object, reference, ...)

is_outlier(object, ...)

## S4 method for signature 'CompositionMatrix,missing'
detect_outlier(
  object,
  ...,
  robust = TRUE,
  method = c("mve", "mcd"),
  quantile = 0.975
)

## S4 method for signature 'CompositionMatrix,CompositionMatrix'
detect_outlier(
  object,
  reference,
  ...,
  robust = TRUE,
  method = c("mve", "mcd"),
  quantile = 0.975
)

## S4 method for signature 'OutlierIndex'
is_outlier(object, robust = TRUE)

```

Arguments

object	A CompositionMatrix .
reference	A CompositionMatrix . If missing, object is used.
...	Further parameters to be passed to MASS::cov.rob() .
robust	A logical scalar: should robust estimators be used?
method	A character string specifying the method to be used. It must be one of "mve" (minimum volume ellipsoid) or "mcd" (minimum covariance determinant; see MASS::cov.rob()). Only used if robust is TRUE.
quantile	A length-one numeric vector giving the significance level. quantile is used as a cut-off value for outlier detection: observations with larger (squared) Mahalanobis distance are considered as potential outliers.

Details

An outlier can be defined as having a very large Mahalanobis distance from all observations. In this way, a certain proportion of the observations can be identified, e.g. the top 2% of values (i.e. values above the 0.98th percentile of the Chi-2 distribution).

On the one hand, the Mahalanobis distance is likely to be strongly affected by the presence of outliers. Rousseeuw and van Zomeren (1990) thus recommend using robust methods (which are not excessively affected by the presence of outliers).

On the other hand, the choice of the threshold for classifying an observation as an outlier should be discussed. There is no apparent reason why a particular threshold should be applicable to all data sets (Filzmoser, Garrett, and Reimann 2005).

Value

- `detect_outlier()` returns an `OutlierIndex` object.
- `is_outlier()` returns a `logical` vector.

Author(s)

N. Frerebeau

References

Filzmoser, P., Garrett, R. G. & Reimann, C. (2005). Multivariate outlier detection in exploration geochemistry. *Computers & Geosciences*, 31(5), 579-587. doi:10.1016/j.cageo.2004.11.013.

Filzmoser, P. & Hron, K. (2008). Outlier Detection for Compositional Data Using Robust Methods. *Mathematical Geosciences*, 40(3), 233-248. doi:10.1007/s1100400791415.

Filzmoser, P., Hron, K. & Reimann, C. (2012). Interpretation of multivariate outliers for compositional data. *Computers & Geosciences*, 39, 77-85. doi:10.1016/j.cageo.2011.06.014.

Rousseeuw, P. J. & van Zomeren, B. C. (1990). Unmasking Multivariate Outliers and Leverage Points. *Journal of the American Statistical Association*, 85(411): 633-639. doi:10.1080/01621459.1990.10474920.

Santos, F. (2020). Modern methods for old data: An overview of some robust methods for outliers detection with applications in osteology. *Journal of Archaeological Science: Reports*, 32, 102423. doi:10.1016/j.jasrep.2020.102423.

See Also

Other outlier detection methods: [plot_outlier](#)

Examples

```
## Data from Day et al. 2011
data("kommos", package = "folio") # Coerce to compositional data
kommos <- remove_NA(kommos, margin = 1) # Remove cases with missing values
coda <- as_composition(kommos, parts = 3:17, groups = 1)

## Detect outliers
out <- detect_outlier(coda)

plot(out, type = "dotchart")
plot(out, type = "distance")

## Detect outliers according to CJ
```

```
ref <- extract(coda, "CJ")
out <- detect_outlier(coda, reference = ref, method = "mcd")
plot(out, type = "dotchart")
```

dist*Distances*

Description

Computes the distances between all rows of in *x*.

Usage

```
## S4 method for signature 'CompositionMatrix'
dist(x, method = "euclidean", diag = FALSE, upper = FALSE, p = 2)
```

Arguments

<i>x</i>	A CompositionMatrix object.
<i>method</i>	A character string specifying the distance measure to be used. See stats::dist() for the available distances.
<i>diag</i>	A logical scalar indicating whether the diagonal of the distance matrix should be printed.
<i>upper</i>	A logical scalar indicating whether the upper triangle of the distance matrix should be printed.
<i>p</i>	An integer giving the power of the Minkowski distance.

Details

Distances are computed on [CLR-transformed](#) data.

Value

A [stats::dist](#) object.

Author(s)

N. Frerebeau

References

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. London: Chapman and Hall, p. 64-91.

Greenacre, M. J. (2019). *Compositional Data Analysis in Practice*. Boca Raton: CRC Press.

See Also

[stats::dist\(\)](#)

Other statistics: [aggregate\(\)](#), [condense\(\)](#), [covariance\(\)](#), [mahalanobis\(\)](#), [margin\(\)](#), [mean\(\)](#), [pip\(\)](#), [quantile\(\)](#), [scale\(\)](#), [variance\(\)](#), [variance_total\(\)](#), [variation\(\)](#)

Examples

```
## Data from Aitchison 1986
data("hongite")

## Coerce to compositional data
coda <- as_composition(hongite)

## Aitchison distance
## (euclidean distance between CLR-transformed compositions)
d <- dist(coda)

## Cluster dendrogram
h <- hclust(d, method = "ward.D2")
plot(h)
```

extract

Group-based Subset

Description

Group-based Subset

Usage

```
extract(object, ...)
```

```
## S4 method for signature 'CompositionMatrix'
extract(object, name)
```

Arguments

object	A CompositionMatrix object.
...	Currently not used.
name	A character vector specifying the group of object to extract.

Value

A [CompositionMatrix](#) object.

Author(s)

N. Frerebeau

See Also

Other subsetting methods: [bind](#), [split\(\)](#), [subset\(\)](#)

Examples

```
## Data from Aitchison 1986
data("slides")
head(slides)

## Coerce to compositional data
coda <- as_composition(slides, groups = 2)

groups(coda)
```

groups

Working With Groups

Description

Retrieves or defines the (reference) groups to which the observations belong.

Usage

```
groups(object)

groups(object) <- value

any_assigned(object)

is_assigned(object)

## S4 method for signature 'CompositionMatrix'
is_assigned(object)

## S4 method for signature 'LogRatio'
is_assigned(object)

## S4 method for signature 'CompositionMatrix'
any_assigned(object)

## S4 method for signature 'LogRatio'
any_assigned(object)

## S4 method for signature 'CompositionMatrix'
groups(object)

## S4 method for signature 'LogRatio'
```

```
groups(object)

## S4 method for signature 'OutlierIndex'
groups(object)

## S4 replacement method for signature 'CompositionMatrix,ANY'
groups(object) <- value

## S4 replacement method for signature 'CompositionMatrix,list'
groups(object) <- value
```

Arguments

object	An object from which to get or set groups.
value	A possible value for the groups of x (typically, a character vector). If value is a list , interaction(value) defines the grouping.

Details

Missing values (NA) or empty strings ("") can be used to specify that a sample does not belong to any group.

Value

- `groups() <- value` returns an object of the same sort as x with the new group names assigned.
- `groups()` returns a [character](#) vector giving the group names of x.
- `any_assigned()` returns a [logical](#) scalar specifying whether or not x has groups.
- `is_assigned()` returns a [logical](#) vector specifying whether or not an observation belongs to a group.

Author(s)

N. Frerebeau

See Also

Other mutators: [mutators](#), [totals\(\)](#)

Examples

```
## Data from Aitchison 1986
data("slides")
head(slides)

## Coerce to compositional data
coda <- as_composition(slides, groups = 2)

groups(coda)
```

 hist *Histogram of Compositional Data*

Description

Produces an histogram of univariate ILR data (see Filzmoser *et al.*, 2009).

Usage

```
## S4 method for signature 'CompositionMatrix'
hist(
  x,
  ...,
  select = 1,
  breaks = "Sturges",
  freq = FALSE,
  labels = FALSE,
  main = NULL,
  sub = NULL,
  ann = graphics::par("ann"),
  axes = TRUE,
  frame.plot = axes
)
```

Arguments

x	A CompositionMatrix object.
...	Further graphical parameters.
select	A length-one vector of column indices.
breaks	An object specifying how to compute the breakpoints (see graphics::hist()).
freq	A logical scalar: should absolute frequencies (counts) be displayed? If FALSE (the default), relative frequencies (probabilities) are displayed (see graphics::hist()).
labels	A logical scalar: should labels be drawn on top of bars? If TRUE, draw the counts or rounded densities; if labels is a character vector, draw itself.
main	A character string giving a main title for the plot.
sub	A character string giving a subtitle for the plot.
ann	A logical scalar: should the default annotation (title and x and y axis labels) appear on the plot?
axes	A logical scalar: should axes be drawn on the plot?
frame.plot	A logical scalar: should a box be drawn around the plot?

Value

hist() is called for its side-effects: it results in a graphic being displayed (invisibly return x).

Author(s)

N. Frerebeau

References

Filzmoser, P., Hron, K. & Reimann, C. (2009). Univariate Statistical Analysis of Environmental (Compositional) Data: Problems and Possibilities. *Science of The Total Environment*, 407(23): 6100-6108. doi:10.1016/j.scitotenv.2009.08.008.

See Also

Other plot methods: [as_graph\(\)](#), [barplot\(\)](#), [plot\(\)](#), [plot_logratio](#)

Examples

```
## Data from Aitchison 1986
data("hongite")

## Coerce to compositional data
coda <- as_composition(hongite)

## Boxplot plot
hist(coda, select = "A")
hist(coda, select = "B")
```

hongite

Hongite Mineralogy

Description

Mineral compositions of 25 specimens of hongite.

Usage

hongite

Format

A [data.frame](#) with 5 variables (minerals):

- A Albite (weight percent).
- B Blandite (weight percent).
- C Cornite (weight percent).
- D Daubite (weight percent).
- E Endite (weight percent).

References

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. London: Chapman and Hall.

See Also

Other datasets: [arctic](#), [boxite](#), [coxite](#), [kongite](#), [lava](#), [predator](#), [slides](#)

kongite

Kongite Mineralogy

Description

Mineral compositions of 25 specimens of kongite.

Usage

kongite

Format

A [data.frame](#) with 5 variables (minerals):

- A Albite (weight percent).
- B Blandite (weight percent).
- C Cornite (weight percent).
- D Daubite (weight percent).
- E Endite (weight percent).

References

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. London: Chapman and Hall.

See Also

Other datasets: [arctic](#), [boxite](#), [coxite](#), [hongite](#), [lava](#), [predator](#), [slides](#)

lava

Skye Lavas Compositions

Description

Chemical compositions of 23 aphyric Skye lavas.

Usage

lava

Format

A `data.frame` with 3 variables (percent):

A Na₂O + K₂O.

F Fe₂O₃.

M MgO.

Source

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. London: Chapman and Hall.
[doi:10.1007/9789400941090](https://doi.org/10.1007/9789400941090).

See Also

Other datasets: [arctic](#), [boxite](#), [coxite](#), [hongite](#), [kongite](#), [predator](#), [slides](#)

mahalanobis*Mahalanobis Distance*

Description

Computes the squared Mahalanobis distance of all rows in `x`.

Usage

```
## S4 method for signature 'CompositionMatrix'  
mahalanobis(x, center, cov, ..., robust = TRUE, method = c("mve", "mcd"))
```

```
## S4 method for signature 'ILR'  
mahalanobis(x, center, cov, ..., robust = TRUE, method = c("mve", "mcd"))
```

Arguments

x	A CompositionMatrix or an ILR object.
center	A numeric vector giving the mean vector of the distribution. If missing, will be estimated from x.
cov	A numeric matrix giving the covariance of the distribution. If missing, will be estimated from x.
...	Extra parameters to be passed to MASS::cov.rob() . Only used if robust is TRUE.
robust	A logical scalar: should robust location and scatter estimation be used?
method	A character string specifying the method to be used. It must be one of "mve" (minimum volume ellipsoid) or "mcd" (minimum covariance determinant). Only used if robust is TRUE.

Value

A [numeric](#) vector.

Author(s)

N. Frerebeau

See Also

[stats::mahalanobis\(\)](#)

Other statistics: [aggregate\(\)](#), [condense\(\)](#), [covariance\(\)](#), [dist](#), [margin\(\)](#), [mean\(\)](#), [pip\(\)](#), [quantile\(\)](#), [scale\(\)](#), [variance\(\)](#), [variance_total\(\)](#), [variation\(\)](#)

Examples

```
## Data from Aitchison 1986
data("hongite")

## Coerce to compositional data
coda <- as_composition(hongite)

## Mahalanobis distance
mahalanobis(coda)
```

margin

Marginal Compositions

Description

Marginal Compositions

Usage

```
margin(x, ...)  
  
## S4 method for signature 'CompositionMatrix'  
margin(x, parts = c(1, 2), name = "*")
```

Arguments

x	A CompositionMatrix object.
...	Currently not used.
parts	An integer or a character vector specifying the columns to be selected.
name	A character string giving the name of the amalgamation column.

Value

A [CompositionMatrix](#) object.

Author(s)

N. Frerebeau

See Also

Other statistics: [aggregate\(\)](#), [condense\(\)](#), [covariance\(\)](#), [dist](#), [mahalanobis\(\)](#), [mean\(\)](#), [pip\(\)](#), [quantile\(\)](#), [scale\(\)](#), [variance\(\)](#), [variance_total\(\)](#), [variation\(\)](#)

Examples

```
## Data from Aitchison 1986  
data("hongite")  
  
## Coerce to compositional data  
coda <- as_composition(hongite)  
  
## Marginal compositions  
mar <- margin(coda, parts = c("B", "D"))  
head(mar)
```

mean

Compositional Mean

Description

Compositional Mean

Usage

```
## S4 method for signature 'CompositionMatrix'  
mean(x, ...)
```

Arguments

x A [CompositionMatrix](#) object.
... Further arguments to be passed to internal methods.

Details

Closed vector of the columns geometric means.

Value

A [numeric](#) vector.

Author(s)

N. Frerebeau

References

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. London: Chapman and Hall, p. 64-91.

See Also

Other statistics: [aggregate\(\)](#), [condense\(\)](#), [covariance\(\)](#), [dist](#), [mahalanobis\(\)](#), [margin\(\)](#), [pip\(\)](#), [quantile\(\)](#), [scale\(\)](#), [variance\(\)](#), [variance_total\(\)](#), [variation\(\)](#)

Examples

```
## Data from Aitchison 1986  
data("hongite")  
  
## Coerce to compositional data  
coda <- as_composition(hongite)  
  
## Mean  
mean(coda)  
  
## Quantile  
quantile(coda)
```

missing

Missing Values Policy

Description

Missing Values Policy

Details

Compositional data are quantitative positive descriptions of the parts of some whole, carrying relative, rather than absolute, information (ie. only relative changes are relevant; Aitchison 1986).

Basically, three situations can be outlined regarding missing values in compositions:

- Unobserved quantities.
- Amounts observed, but which happen to be below the detection limit (thus interpreted as small unknown values).
- Absolutely zero quantities.

These situations can be represented in several ways:

- The presence of zeros.
- The presence of missing values (NA).

When creating a [CompositionMatrix](#) object, the presence of zero and NA values is allowed: this makes it possible to explore and visualize the data while preserving the missing structure. However, **the user must deal with these missing values before proceeding further** (e.g. by removing incomplete cases or replacing the values concerned): log-ratio transformations cannot be computed in the presence of zeros or missing values.

Note

If you need more advanced features (e.g. imputation of missing values), you should consider the [compositions](#) or [robCompositions](#) package.

References

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. London: Chapman and Hall.

See Also

Other imputation methods: [replace_NA\(\)](#), [replace_zero\(\)](#)

mix *Mixed-Mode Analysis*

Description

Mixes chemical and petrographic matrices.

Usage

```
mix(x, y, ...)  
  
## S4 method for signature 'matrix,matrix'  
mix(x, y, lambda = 1, ...)  
  
## S4 method for signature 'dist,dist'  
mix(x, y, mu = 0.5)
```

Arguments

x	A matrix of chemical compositional data or a dissimilarity matrix for these chemical compositional data.
y	A matrix of coded mineralogical binary data or a dissimilarity matrix for these mineralogical data.
...	Extra parameters to be passed to cluster::daisy() .
lambda	A length-one numeric vector giving a weighting factor.
mu	A length-one numeric vector that lies between 0 and 1 giving the mixing parameter.

Value

A [stats::dist](#) object.

Methods (by class)

- `mix(x = matrix, y = matrix)`: First approach of mixed-mode analysis.
- `mix(x = dist, y = dist)`: Second approach of mixed-mode analysis.

Note

Experimental.

Author(s)

N. Frerebeau

References

Baxter, M. J., Beardah, C. C., Papageorgiou, I., Cau, M. A., Day, P. M. & Kilikoglou, V. (2008). On Statistical Approaches to the Study of Ceramic Artefacts Using Geochemical and Petrographic Data. *Archaeometry*, 50(1): 142-157. doi:10.1111/j.14754754.2007.00359.x.

Beardah, C. C., Baxter, M. J., Papageorgiou, I. & Cau, M. A. (2003). "Mixed-Mode" Approaches to the Grouping of Ceramic Artefacts Using S-Plus. In M. Doerr and A. Sarris, *The Digital Heritage of Archaeology*, p. 261-266. Athens: Archive of Monuments and Publications, Hellenic Ministry of Culture.

Gower, J. C. (1971). A general coefficient of similarity and some of its properties. *Biometrics*, 27(4):857-874. doi:10.2307/2528823.

Examples

```
## Can Sora datasets
## Data from Cau (1999) and Cau et al. (2007)
path_chem <- system.file("extdata", "cansora_chemistry.csv", package = "nexus")
chemistry <- read.csv(path_chem, header = TRUE, row.names = 1)
path_petro <- system.file("extdata", "cansora_petrography.csv", package = "nexus")
petrography <- read.csv(path_petro, header = TRUE, row.names = 1)

## Prepare chemical data
major <- c("Fe2O3", "Al2O3", "MnO", "P2O5", "TiO2",
          "MgO", "CaO", "Na2O", "K2O", "SiO2")
chem <- chemistry[-1, major]

## Prepare petrographic data
petro <- petrography[-c(7, 8), -1]
petro <- cdt(petro) # Get the complete disjunctive table

## First approach
mix1 <- mix(as.matrix(chem), as.matrix(petro), lambda = 2)
mds1 <- stats::cmdscale(mix1) # Multi-Dimensional Scaling
plot(mds1)
```

mutators

Get or Set Parts of an Object

Description

Getters and setters to retrieve or set parts of an object.

Usage

```
## S4 method for signature 'CompositionMatrix'
labels(object, ...)

## S4 method for signature 'LogRatio'
```

```
labels(object, ...)  
  
## S4 method for signature 'ALR'  
weights(object, ...)  
  
## S4 method for signature 'LR'  
weights(object, ...)  
  
## S4 method for signature 'LogRatio'  
weights(object, ...)
```

Arguments

object	An object from which to get or set element(s).
...	Currently not used.

Author(s)

N. Frerebeau

See Also

Other mutators: [groups\(\)](#), [totals\(\)](#)

pca

Principal Components Analysis

Description

Computes a principal components analysis based on the singular value decomposition.

Usage

```
## S4 method for signature 'CompositionMatrix'  
pca(  
  object,  
  center = TRUE,  
  scale = FALSE,  
  rank = NULL,  
  sup_row = NULL,  
  sup_col = NULL,  
  weight_row = NULL,  
  weight_col = NULL  
)  
  
## S4 method for signature 'LogRatio'  
pca(  
  object,  
  center = TRUE,  
  scale = FALSE,  
  rank = NULL,  
  sup_row = NULL,  
  sup_col = NULL,  
  weight_row = NULL,  
  weight_col = NULL  
)
```

```

object,
center = TRUE,
scale = FALSE,
rank = NULL,
sup_row = NULL,
sup_col = NULL,
weight_row = NULL,
weight_col = NULL
)

```

Arguments

object	A CompositionMatrix or LogRatio object.
center	A logical scalar: should the variables be shifted to be zero centered?
scale	A logical scalar: should the variables be scaled to unit variance?
rank	An integer value specifying the maximal number of components to be kept in the results. If NULL (the default), $p - 1$ components will be returned.
sup_row	A vector specifying the indices of the supplementary rows.
sup_col	A vector specifying the indices of the supplementary columns.
weight_row	A numeric vector specifying the active row (individual) weights. If NULL (the default), uniform weights are used. Row weights are internally normalized to sum 1
weight_col	A numeric vector specifying the active column (variable) weights. If NULL (the default), uniform weights (1) are used.

Value

A [dimensio::PCA](#) object. See [dimensio::pca\(\)](#) for details.

Methods (by class)

- `pca(CompositionMatrix)`: PCA of centered log-ratio, i.e. log-ratio analysis (LRA).

Author(s)

N. Frerebeau

References

- Aitchison, J. and Greenacre, M. (2002). Biplots of compositional data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 51: 375-392. doi:10.1111/14679876.00275.
- Filzmoser, P., Hron, K. and Reimann, C. (2009). Principal component analysis for compositional data with outliers. *Environmetrics*, 20: 621-632. doi:10.1002/env.966.

See Also

[dimensio::pca\(\)](#), [dimensio::biplot\(\)](#), [dimensio::screeplot\(\)](#), [dimensio::viz_individuals\(\)](#), [dimensio::viz_variables\(\)](#)

Examples

```
## Data from Day et al. 2011
data("kommos", package = "folio") # Coerce to compositional data
kommos <- remove_NA(kommos, margin = 1) # Remove cases with missing values
coda <- as_composition(kommos, groups = 1) # Use ceramic types for grouping

## Log-Ratio Analysis
X <- pca(coda)

## Biplot
biplot(X)

## Explore results
viz_individuals(X)
viz_variables(X)
```

perturbation	<i>Perturbation Operation</i>
--------------	-------------------------------

Description

Perturbation of two compositions.

Usage

```
perturbation(x, y, ...)
```

```
## S4 method for signature 'numeric,numeric'
perturbation(x, y)
```

```
## S4 method for signature 'CompositionMatrix,numeric'
perturbation(x, y)
```

```
## S4 method for signature 'CompositionMatrix,matrix'
perturbation(x, y)
```

Arguments

x, y	A numeric vector of compositional data or a CompositionMatrix object.
...	Currently not used.

Details

In compositional geometry, perturbation plays the role of sum (translation). It is the closed component-wise product of two compositions.

Value

A [numeric](#) vector.

Author(s)

N. Frerebeau

See Also

Other operations in the simplex: [arithmetic](#), [closure\(\)](#), [powering\(\)](#), [scalar\(\)](#)

Examples

```
x <- as_composition(c(1, 2, 3))
y <- as_composition(c(1, 2, 1))

## Perturbation
perturbation(x, y)
x + y

## Powering
powering(y, 2)
y * 2

## Scalar product
scalar(x, y)
```

pip

Proportionality Index of Parts (PIP)

Description

Computes an index of association between parts.

Usage

```
pip(x, ...)
```

S4 method for signature 'CompositionMatrix'

```
pip(x)
```

Arguments

x A [CompositionMatrix](#) object.

... Currently not used.

Details

The proportionality index of parts (PIP) is based on the [variation matrix](#), but maintains the range of values within (0, 1).

Value

A [matrix](#).

Author(s)

N. Frerebeau

References

Egozcue, J. J.. & Pawlowsky-Glahn, V. (2023). Subcompositional Coherence and a Novel Proportionality Index of Parts. *SORT*, 47(2): 229-244. [doi:10.57645/20.8080.02.7](https://doi.org/10.57645/20.8080.02.7).

See Also

Other statistics: [aggregate\(\)](#), [condense\(\)](#), [covariance\(\)](#), [dist](#), [mahalanobis\(\)](#), [margin\(\)](#), [mean\(\)](#), [quantile\(\)](#), [scale\(\)](#), [variance\(\)](#), [variance_total\(\)](#), [variation\(\)](#)

Examples

```
## Data from Aitchison 1986
data("hongite")

## Coerce to compositional data
coda <- as_composition(hongite)

## Variation matrix
## (Aitchison 1986, definition 4.4)
(varia <- variation(coda))

## Cluster dendrogram
d <- as.dist(varia)
h <- hclust(d, method = "ward.D2")
plot(h)

## Heatmap
stats::heatmap(
  varia,
  distfun = stats::as.dist,
  hclustfun = function(x) stats::hclust(x, method = "ward.D2"),
  symm = TRUE,
  scale = "none"
)
```

plot

Plot Compositional Data

Description

Displays a matrix of ternary plots.

Usage

```
## S4 method for signature 'CompositionMatrix,missing'
plot(
  x,
  ...,
  by = groups(x),
  margin = NULL,
  color = palette_color_discrete(),
  symbol = palette_shape()
)
```

Arguments

x	A CompositionMatrix object.
...	Further graphical parameters .
by	A vector of grouping elements, as long as the variables in x.
margin	A character string or an integer giving the index of the column to be used as the third part of the ternary plots. If NULL (the default), marginal compositions will be used (i.e. the geometric mean of the non-selected parts).
color	A palette function that when called with a single argument returns a character vector of colors.
symbol	A palette function that when called with a single argument returns a vector of symbols.

Value

plot() is called for its side-effects: it results in a graphic being displayed (invisibly return x).

Author(s)

N. Frerebeau

See Also

[isopleuros::ternary_pairs\(\)](#), [isopleuros::ternary_plot\(\)](#)
 Other plot methods: [as_graph\(\)](#), [barplot\(\)](#), [hist\(\)](#), [plot_logratio](#)

Examples

```
## Data from Day et al. 2011
data("kommos", package = "folio") # Coerce to compositional data
kommos <- remove_NA(kommos, margin = 1) # Remove cases with missing values
coda <- as_composition(kommos, parts = 3:8, groups = 1)

## Use ceramic types for grouping
plot(coda)

## Center and scale ternary plots
plot(coda, by = NULL, center = TRUE, scale = TRUE)
```

plot_logratio	<i>Plot Log-Ratios</i>
---------------	------------------------

Description

Displays a density plot.

Usage

```
## S4 method for signature 'LogRatio,missing'
plot(
  x,
  ...,
  by = groups(x),
  color = palette_color_discrete(),
  rug = TRUE,
  ticksize = 0.05,
  ncol = NULL,
  flip = FALSE,
  xlab = NULL,
  ylab = NULL,
  main = NULL,
  ann = graphics::par("ann"),
  axes = TRUE,
  frame.plot = axes,
  legend = list(x = "top")
)
```

Arguments

x	A LogRatio object.
...	Further graphical parameters , particularly, border and col.
by	A vector of grouping elements, as long as the variables in x. If set, a matrix of panels defined by groups will be drawn.
color	A palette function that when called with a single argument returns a character vector of colors.
rug	A logical scalar: should a <i>rug</i> representation (1-d plot) of the data be added to the plot?
ticksize	A length-one numeric vector giving the length of the ticks making up the <i>rug</i> . Positive lengths give inwards ticks. Only used if rug is TRUE.
ncol	An integer specifying the number of columns to use. Defaults to 1 for up to 4 groups, otherwise to 2.
flip	A logical scalar: should the y-axis (ticks and numbering) be flipped from side 2 (left) to 4 (right) from variable to variable?
xlab, ylab	A character vector giving the x and y axis labels.

main	A character string giving a main title for the plot.
ann	A logical scalar: should the default annotation (title and x and y axis labels) appear on the plot?
axes	A logical scalar: should axes be drawn on the plot?
frame.plot	A logical scalar: should a box be drawn around the plot?
legend	A list of additional arguments to be passed to <code>graphics::legend()</code> ; names of the list are used as argument names. If NULL, no legend is displayed.

Value

`plot()` is called for its side-effects: it results in a graphic being displayed (invisibly return x).

Author(s)

N. Frerebeau

See Also

Other plot methods: [as_graph\(\)](#), [barplot\(\)](#), [hist\(\)](#), [plot\(\)](#)

Examples

```
## Data from Day et al. 2011
data("kommos", package = "folio") # Coerce to compositional data
kommos <- remove_NA(kommos, margin = 1) # Remove cases with missing values
coda <- as_composition(kommos, groups = 1)

## Log ratio
clr <- transform_clr(coda)

## Density plot
plot(clr, by = NULL, flip = TRUE)

## Use ceramic types for grouping
plot(clr, flip = TRUE)
```

plot_outlier

Plot Outliers

Description

Plot Outliers

Usage

```
## S4 method for signature 'OutlierIndex,missing'
plot(
  x,
  ...,
  type = c("dotchart", "distance"),
  robust = TRUE,
  colors = color("discreterainbow"),
  symbols = c(16, 1, 3),
  xlim = NULL,
  ylim = NULL,
  xlab = NULL,
  ylab = NULL,
  main = NULL,
  sub = NULL,
  ann = graphics::par("ann"),
  axes = TRUE,
  frame.plot = axes,
  panel.first = NULL,
  panel.last = NULL,
  legend = list(x = "topleft")
)
```

Arguments

x	An OutlierIndex object.
...	Further graphical parameters .
type	A character string specifying the type of plot that should be made. It must be one of "dotchart" or "distance". Any unambiguous substring can be given.
robust	A logical scalar: should robust Mahalanobis distances be displayed? Only used if type is "dotchart".
colors	A vector of colors or a function that when called with a single argument (an integer specifying the number of colors) returns a vector of colors. Will be mapped to the group names.
symbols	A length-three vector of symbol specification for non-outliers and outliers (resp.).
xlim	A length-two numeric vector giving the x limits of the plot. The default value, NULL, indicates that the range of the finite values to be plotted should be used.
ylim	A length-two numeric vector giving the y limits of the plot. The default value, NULL, indicates that the range of the finite values to be plotted should be used.
xlab, ylab	A character vector giving the x and y axis labels.
main	A character string giving a main title for the plot.
sub	A character string giving a subtitle for the plot.
ann	A logical scalar: should the default annotation (title and x and y axis labels) appear on the plot?
axes	A logical scalar: should axes be drawn on the plot?

frame.plot	A logical scalar: should a box be drawn around the plot?
panel.first	An an expression to be evaluated after the plot axes are set up but before any plotting takes place. This can be useful for drawing background grids.
panel.last	An expression to be evaluated after plotting has taken place but before the axes, title and box are added.
legend	A list of additional arguments to be passed to graphics::legend() ; names of the list are used as argument names. If NULL, no legend is displayed.

Value

plot() is called for its side-effects: is results in a graphic being displayed (invisibly return x).

Author(s)

N. Frerebeau

References

- Filzmoser, P., Garrett, R. G. & Reimann, C. (2005). Multivariate outlier detection in exploration geochemistry. *Computers & Geosciences*, 31(5), 579-587. doi:10.1016/j.cageo.2004.11.013.
- Filzmoser, P. & Hron, K. (2008). Outlier Detection for Compositional Data Using Robust Methods. *Mathematical Geosciences*, 40(3), 233-248. doi:10.1007/s1100400791415.
- Filzmoser, P., Hron, K. & Reimann, C. (2012). Interpretation of multivariate outliers for compositional data. *Computers & Geosciences*, 39, 77-85. doi:10.1016/j.cageo.2011.06.014.

See Also

Other outlier detection methods: [detect_outlier\(\)](#)

Examples

```
## Data from Day et al. 2011
data("kommos", package = "folio") # Coerce to compositional data
kommos <- remove_NA(kommos, margin = 1) # Remove cases with missing values
coda <- as_composition(kommos, parts = 3:17, groups = 1)

## Detect outliers
out <- detect_outlier(coda)

plot(out, type = "dotchart")
plot(out, type = "distance")

## Detect outliers according to CJ
ref <- extract(coda, "CJ")
out <- detect_outlier(coda, reference = ref, method = "mcd")
plot(out, type = "dotchart")
```

powering	<i>Powering Operation</i>
----------	---------------------------

Description

Perturbation of two compositions.

Usage

```
powering(x, a, ...)  
  
## S4 method for signature 'numeric,numeric'  
powering(x, a)  
  
## S4 method for signature 'CompositionMatrix,numeric'  
powering(x, a)
```

Arguments

x	A numeric vector of compositional data or a CompositionMatrix object.
a	A numeric constant.
...	Currently not used.

Details

In compositional geometry, powering replaces the product of a vector by a scalar (scaling) and is defined as the closed powering of the components by a given scalar.

Value

A [numeric](#) vector.

Author(s)

N. Frerebeau

See Also

Other operations in the simplex: [arithmetic](#), [closure\(\)](#), [perturbation\(\)](#), [scalar\(\)](#)

Examples

```
x <- as_composition(c(1, 2, 3))  
y <- as_composition(c(1, 2, 1))  
  
## Perturbation  
perturbation(x, y)  
x + y
```

```
## Powering
powering(y, 2)
y * 2

## Scalar product
scalar(x, y)
```

predator

Predator-Prey Compositions

Description

Predator-prey compositions at 25 different sites.

Usage

predator

Format

A `data.frame` with 3 variables (proportions):

P Predator.

Q Prey of species Q.

R Prey of species R.

Source

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. London: Chapman and Hall.
[doi:10.1007/9789400941090](https://doi.org/10.1007/9789400941090).

See Also

Other datasets: [arctic](#), [boxite](#), [coxite](#), [hongite](#), [kongite](#), [lava](#), [slides](#)

quantile	<i>Sample Quantiles</i>
----------	-------------------------

Description

Sample Quantiles

Usage

```
## S4 method for signature 'CompositionMatrix'  
quantile(x, ..., probs = seq(0, 1, 0.25), na.rm = FALSE, names = TRUE)
```

Arguments

x	A CompositionMatrix object.
...	Currently not used.
probs	A numeric vector of probabilities with values in [0, 1].
na.rm	A logical scalar: should missing values be removed?
names	A logical scalar: should results be named?

Value

A [numeric](#) matrix.

Author(s)

N. Frerebeau

References

Filzmoser, P., Hron, K. & Reimann, C. (2009). Univariate Statistical Analysis of Environmental (Compositional) Data: Problems and Possibilities. *Science of The Total Environment*, 407(23): 6100-6108. doi:[10.1016/j.scitotenv.2009.08.008](https://doi.org/10.1016/j.scitotenv.2009.08.008).

See Also

Other statistics: [aggregate\(\)](#), [condense\(\)](#), [covariance\(\)](#), [dist](#), [mahalanobis\(\)](#), [margin\(\)](#), [mean\(\)](#), [pip\(\)](#), [scale\(\)](#), [variance\(\)](#), [variance_total\(\)](#), [variation\(\)](#)

Examples

```
## Data from Aitchison 1986  
data("hongite")  
  
## Coerce to compositional data  
coda <- as_composition(hongite)
```

```
## Mean
mean(coda)

## Quantile
quantile(coda)
```

replace_NA

Missing Values Replacement

Description

Multiplicative replacement of missing values.

Usage

```
## S4 method for signature 'CompositionMatrix'
replace_NA(x, value)
```

Arguments

x A [CompositionMatrix](#) object.
value A [numeric](#) vector giving the replacement values.

Value

An [CompositionMatrix](#) object, where all missing values have been replaced.

Author(s)

N. Frerebeau

References

Martín-Fernández, J. A., Barceló-Vidal, C. & Pawłowsky-Glahn, V. (2003). Dealing with Zeros and Missing Values in Compositional Data Sets Using Nonparametric Imputation. *Mathematical Geology*, 35(3): 253-278. doi:[10.1023/A:1023866030544](https://doi.org/10.1023/A:1023866030544).

See Also

Other imputation methods: [missing](#), [replace_zero\(\)](#)

Examples

```
## Data from Martín-Fernández et al. 2003
X <- data.frame(
  X1 = c(0.0000, 0.1304, 0.1963),
  X2 = c(0.1250, 0.3151, NA),
  X3 = c(0.1237, NA, NA),
  X4 = c(0.7253, 0.2002, 0.0819),
  X5 = c(0.0260, 0.3543, 0.0114)
)

## Coerce to a compositional matrix
Y <- as_composition(X)

## Replace zeros
Z <- replace_NA(Y, value = 0.2)
Z
```

replace_zero

Zero-Replacement

Description

Multiplicative replacement of zeros.

Usage

```
## S4 method for signature 'CompositionMatrix'
replace_zero(x, value, delta = 2/3)
```

Arguments

x A `CompositionMatrix` object.

value A `numeric` vector giving the detection limits of each part (in $(0, 1)$).

delta A `numeric` vector specifying the fraction of the detection limit to be used in replacement.

Value

An `CompositionMatrix` object, where all zero values have been replaced.

Author(s)

N. Frerebeau

References

- Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. London: Chapman and Hall.
- Martín-Fernández, J. A., Barceló-Vidal, C. & Pawłowsky-Glahn, V. (2003). Dealing with Zeros and Missing Values in Compositional Data Sets Using Nonparametric Imputation. *Mathematical Geology*, 35(3): 253-278. doi:10.1023/A:1023866030544.

See Also

Other imputation methods: [missing](#), [replace_NA\(\)](#)

Examples

```
## Data from Martín-Fernández et al. 2003
X <- data.frame(
  X1 = c(0.0000, 0.1304, 0.1963),
  X2 = c(0.1250, 0.3151, NA),
  X3 = c(0.1237, NA, NA),
  X4 = c(0.7253, 0.2002, 0.0819),
  X5 = c(0.0260, 0.3543, 0.0114)
)

## Coerce to a compositional matrix
Y <- as_composition(X)

## Replace zeros
Z <- replace_zero(Y, value = 0.02, delta = 2/3)
Z
```

scalar

Scalar Product

Description

Computes the Aitchison scalar product of two compositions.

Usage

```
scalar(x, y, ...)
```

S4 method for signature 'numeric,numeric'

```
scalar(x, y)
```

S4 method for signature 'CompositionMatrix,CompositionMatrix'

```
scalar(x, y)
```

Arguments

`x, y` A `CompositionMatrix` object.

`...` Currently not used.

Value

A [numeric](#) vector.

Author(s)

N. Frerebeau

See Also

Other operations in the simplex: [arithmetic](#), [closure\(\)](#), [perturbation\(\)](#), [powering\(\)](#)

Examples

```
x <- as_composition(c(1, 2, 3))
y <- as_composition(c(1, 2, 1))

## Perturbation
perturbation(x, y)
x + y

## Powering
powering(y, 2)
y * 2

## Scalar product
scalar(x, y)
```

scale

Scaling and Centering of Compositional Data

Description

Scaling and Centering of Compositional Data

Usage

```
## S4 method for signature 'CompositionMatrix'
scale(x, center = TRUE, scale = TRUE)
```

Arguments

x A [CompositionMatrix](#) object.

center A [logical](#) scalar or a [numeric](#) vector giving the center to be subtracted.

scale A [logical](#) scalar or a length-one [numeric](#) vector giving a scaling factor for multiplication.

Value

A [CompositionMatrix](#) object.

Author(s)

N. Frerebeau

References

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. London: Chapman and Hall, p. 64-91.

Boogaart, K. G. van den & Tolosana-Delgado, R. (2013). *Analyzing Compositional Data with R*. Berlin Heidelberg: Springer-Verlag. doi:10.1007/9783642368097.

See Also

Other statistics: [aggregate\(\)](#), [condense\(\)](#), [covariance\(\)](#), [dist](#), [mahalanobis\(\)](#), [margin\(\)](#), [mean\(\)](#), [pip\(\)](#), [quantile\(\)](#), [variance\(\)](#), [variance_total\(\)](#), [variation\(\)](#)

Examples

```
## Data from Aitchison 1986
data("hongite")

## Coerce to compositional data
coda <- as_composition(hongite)

## Center and scale
scaled <- scale(coda, center = TRUE, scale = TRUE)
mean(scaled)
head(scaled)
```

slides

Thin Sections Compositions

Description

Mineral compositions of five slides as reported by five analysts.

Usage

```
slides
```

Format

A [data.frame](#) with 9 variables:

analyst Analyst number.

slide Slide number.

quartz Quartz (percent).

microcline Microcline (percent).

plagioclass Plagioclass (percent).
 biotite Biotite (percent).
 muscovite Muscovite (percent).
 opaques Opaque minerals (percent).
 nonopaques Non-opaque minerals (percent).

References

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. London: Chapman and Hall.

See Also

Other datasets: [arctic](#), [boxite](#), [coxite](#), [hongite](#), [kongite](#), [lava](#), [predator](#)

split	<i>Divide into Groups</i>
-------	---------------------------

Description

Divides the compositional matrix `x` into the groups defined by `f`.

Usage

```
## S4 method for signature 'CompositionMatrix'
split(x, f, drop = FALSE, ...)
```

```
## S4 method for signature 'LogRatio'
split(x, f, drop = FALSE, ...)
```

Arguments

<code>x</code>	A CompositionMatrix object.
<code>f</code>	A 'factor' in the sense that <code>as.factor(f)</code> defines the grouping, or a list of such factors in which case their interaction is used for the grouping (see <code>base::split()</code>).
<code>drop</code>	A logical scalar: should levels that do not occur be dropped?
<code>...</code>	Currently not used.

Value

A list of [CompositionMatrix](#) objects.

Author(s)

N. Frerebeau

See Also

Other subsetting methods: [bind](#), [extract\(\)](#), [subset\(\)](#)

Examples

```
## Create a data.frame
X <- data.frame(
  samples = c("A", "A", "A", "B", "B", "B", "C", "C", "C"),
  groups = c("X", "X", "X", "X", NA, NA, "Y", "Y", "Y"),
  Ca = c(7.72, 7.32, 3.11, 7.19, 7.41, 5, 4.18, 1, 4.51),
  Fe = c(6.12, 5.88, 5.12, 6.18, 6.02, 7.14, 5.25, 5.28, 5.72),
  Na = c(0.97, 1.59, 1.25, 0.86, 0.76, 0.51, 0.75, 0.52, 0.56)
)

## Coerce to a compositional matrix
Y <- as_composition(X)

## Split by group
## !\ Unassigned samples are discarded ! !\
(s1 <- split(Y, f = X$groups))

## Split by group
## Keep unassigned samples, see help(factor)
(s2 <- split(Y, f = factor(X$groups, exclude = NULL)))

## Bind by rows
do.call(rbind, s2)
```

subset

Extract or Replace Parts of an Object

Description

Operators acting on objects to extract or replace parts.

Usage

```
## S4 method for signature 'CompositionMatrix,missing,missing,missing'
x[i, j, ..., drop = TRUE]

## S4 method for signature 'CompositionMatrix,missing,missing,logical'
x[i, j, ..., drop = TRUE]

## S4 method for signature 'CompositionMatrix,index,missing,missing'
x[i, j, ..., drop = TRUE]

## S4 method for signature 'CompositionMatrix,index,missing,logical'
x[i, j, ..., drop = TRUE]
```



```

## S4 method for signature 'CompositionMatrix,missing,index,missing'
x[i, j, ..., drop = TRUE]

## S4 method for signature 'CompositionMatrix,missing,index,logical'
x[i, j, ..., drop = TRUE]

## S4 method for signature 'CompositionMatrix,index,index,missing'
x[i, j, ..., drop = TRUE]

## S4 method for signature 'CompositionMatrix,index,index,logical'
x[i, j, ..., drop = TRUE]

## S4 replacement method for signature 'CompositionMatrix'
x[i, j, ...] <- value

## S4 replacement method for signature 'CompositionMatrix'
x[[i, j, ...]] <- value

```

Arguments

<code>x</code>	An object from which to extract element(s) or in which to replace element(s).
<code>i, j</code>	Indices specifying elements to extract or replace. Indices are numeric , integer or character vectors or empty (missing) or NULL. Numeric values are coerced to integer as by as.integer() . Character vectors will be matched to the name of the elements. An empty index (a comma separated blank) indicates that all entries in that dimension are selected.
<code>...</code>	Currently not used.
<code>drop</code>	A logical scalar: should the result be coerced to the lowest possible dimension? This only works for extracting elements, not for the replacement. Defaults to FALSE.
<code>value</code>	A possible value for the element(s) of <code>x</code> .

Value

A subsetted object of the same sort as `x`.

Author(s)

N. Frerebeau

See Also

Other subsetting methods: [bind](#), [extract\(\)](#), [split\(\)](#)

Examples

```
## Data from Aitchison 1986
data("hongite")

## Coerce to compositional data
coda <- as_composition(hongite)
head(coda)

## Subset
coda[[1, 1]] # Get the first value
coda[1] # Get the first value
coda[, ] # Get all values
coda[1, ] # Get the first row

## Subcomposition
subcoda <- coda[, 1:3] # Get the first three column
head(subcoda)
```

totals	<i>Row Sums</i>
--------	-----------------

Description

Retrieves or defines the row sums (before [closure](#)).

Usage

```
totals(object)

totals(object) <- value

## S4 method for signature 'CompositionMatrix'
totals(object)

## S4 method for signature 'LogRatio'
totals(object)

## S4 replacement method for signature 'CompositionMatrix'
totals(object) <- value
```

Arguments

object	An object from which to get or set totals.
value	A possible value for the totals of x.

Value

- `totals() <- value` returns an object of the same sort as x with the new row sums assigned.
- `totals()` returns the row sums of x.

Author(s)

N. Frerebeau

See Also

Other mutators: [groups\(\)](#), [mutators](#)

Examples

```
## Create a count matrix
A1 <- matrix(data = sample(1:100, 100, TRUE), nrow = 20)

## Coerce to compositions
B <- as_composition(A1)

## Row sums are internally stored before coercing to relative frequencies
totals(B)

## This allows to restore the source data
A2 <- as_amounts(B)

## Coerce to a data.frame
X <- data.frame(B)
head(X)
```

transform_alr

Additive Log-Ratios (ALR)

Description

Computes ALR transformation.

Usage

```
transform_alr(object, ...)
```

S4 method for signature 'CompositionMatrix'

```
transform_alr(object, j = ncol(object), weights = FALSE)
```

S4 method for signature 'CLR'

```
transform_alr(object, j = ncol(object))
```

Arguments

object A [CompositionMatrix](#) object.

... Currently not used.

j An [integer](#) giving the index of the rationing part (denominator).

weights A **logical** scalar: should varying weights (column means) be computed? If FALSE (the default), equally-weighted parts are used. Alternatively, a positive **numeric** vector of weights can be specified (will be rescaled to sum to 1). Weights will be used internally by other methods (e.g. `variance()`).

Details

The ALR transformation is the logratio of a pair of parts with respect to a fixed part.

Value

An **ALR** object.

Author(s)

N. Frerebeau

References

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. London: Chapman and Hall.

Greenacre, M. J. (2019). *Compositional Data Analysis in Practice*. Boca Raton: CRC Press.

Greenacre, M. J. (2021). Compositional Data Analysis. *Annual Review of Statistics and Its Application*, 8(1): 271-299. doi:10.1146/annurevstatistics042720124436.

See Also

Other log-ratio transformations: `transform_clr()`, `transform_ilr()`, `transform_inverse()`, `transform_lr()`, `transform_plr()`

Examples

```
## Data from Aitchison 1986
data("hongite")

## Coerce to compositional data
coda <- as_composition(hongite)

## Additive log-ratio
alr <- transform_alr(coda)

## Inverse transformation
inv_alr <- transform_inverse(alr)
all.equal(coda, inv_alr)
```

transform_clr	<i>Centered Log-Ratios (CLR)</i>
---------------	----------------------------------

Description

Computes CLR transformation.

Usage

```
transform_clr(object, ...)  
  
## S4 method for signature 'CompositionMatrix'  
transform_clr(object, weights = FALSE)  
  
## S4 method for signature 'ALR'  
transform_clr(object)
```

Arguments

object	A CompositionMatrix object.
...	Currently not used.
weights	A logical scalar: should varying weights (column means) be used? If FALSE (the default), equally-weighted parts are used. Alternatively, a positive numeric vector of weights can be specified (will be rescaled to sum to 1).

Details

The CLR transformation computes the log of each part relative to the geometric mean of all parts.

Value

A [CLR](#) object.

Author(s)

N. Frerebeau

References

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. London: Chapman and Hall.
Greenacre, M. J. (2019). *Compositional Data Analysis in Practice*. Boca Raton: CRC Press.
Greenacre, M. J. (2021). Compositional Data Analysis. *Annual Review of Statistics and Its Application*, 8(1): 271-299. doi:[10.1146/annurevstatistics042720124436](https://doi.org/10.1146/annurevstatistics042720124436).

See Also

Other log-ratio transformations: [transform_alr\(\)](#), [transform_ilr\(\)](#), [transform_inverse\(\)](#), [transform_lr\(\)](#), [transform_plr\(\)](#)

Examples

```
## Data from Aitchison 1986
data("hongite")

## Coerce to compositional data
coda <- as_composition(hongite)

## Centered log-ratio
clr <- transform_clr(coda)

## Inverse transformation
inv_clr <- transform_inverse(clr)
all.equal(coda, inv_clr)
```

transform_ilr	<i>Isometric Log-Ratios (ILR)</i>
---------------	-----------------------------------

Description

Computes ILR transformations.

Usage

```
transform_ilr(object, ...)
```

S4 method for signature 'CompositionMatrix'
transform_ilr(object)

S4 method for signature 'CLR'
transform_ilr(object)

S4 method for signature 'ALR'
transform_ilr(object)

Arguments

object	A CompositionMatrix object.
...	Currently not used.

Details

The ILR transformation provides the coordinates of any composition with respect to a given orthonormal basis. `transform_ilr()` uses the orthonormal basis (Helmert matrix) originally defined by Egozcue *et al.* (2003).

Value

An [ILR](#) object.

Author(s)

N. Frerebeau

References

Egozcue, J. J., Pawlowsky-Glahn, V., Mateu-Figueras, G. & Barceló-Vidal, C. (2003). Isometric Logratio Transformations for Compositional Data Analysis. *Mathematical Geology*, 35(3), 279-300. doi:10.1023/A:1023818214614.

Greenacre, M. J. (2019). *Compositional Data Analysis in Practice*. Boca Raton: CRC Press.

Greenacre, M. J. (2021). Compositional Data Analysis. *Annual Review of Statistics and Its Application*, 8(1): 271-299. doi:10.1146/annurevstatistics042720124436.

See Also

Other log-ratio transformations: [transform_alr\(\)](#), [transform_clr\(\)](#), [transform_inverse\(\)](#), [transform_lr\(\)](#), [transform_plr\(\)](#)

Examples

```
## Data from Aitchison 1986
data("hongite")

## Coerce to compositional data
coda <- as_composition(hongite)

## Isometric log-ratio
ilr <- transform_ilr(coda)
plr <- transform_plr(coda)

## Inverse transformation
inv_ilr <- transform_inverse(ilr)
all.equal(coda, inv_ilr)

inv_plr <- transform_inverse(plr)
all.equal(coda, inv_plr)
```

transform_inverse	<i>Inverse Log-Ratio Transformation</i>
-------------------	---

Description

Computes inverse log-ratio transformations.

Usage

```
transform_inverse(object, origin, ...)  
  
## S4 method for signature 'CLR,missing'  
transform_inverse(object)  
  
## S4 method for signature 'ALR,missing'  
transform_inverse(object)  
  
## S4 method for signature 'ILR,missing'  
transform_inverse(object)  
  
## S4 method for signature 'matrix,ILR'  
transform_inverse(object, origin)
```

Arguments

object	A LogRatio object.
origin	A LogRatio object to be used for the inverse transformation.
...	Currently not used.

Value

A [CompositionMatrix](#) object.

Author(s)

N. Frerebeau

References

- Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. London: Chapman and Hall.
- Egozcue, J. J., Pawłowsky-Glahn, V., Mateu-Figueras, G. & Barceló-Vidal, C. (2003). Isometric Logratio Transformations for Compositional Data Analysis. *Mathematical Geology*, 35(3), 279-300. doi:10.1023/A:1023818214614.
- Fišerová, E. & Hron, K. (2011). On the Interpretation of Orthonormal Coordinates for Compositional Data. *Mathematical Geosciences*, 43(4), 455-468. doi:10.1007/s110040119333x.
- Greenacre, M. J. (2019). *Compositional Data Analysis in Practice*. Boca Raton: CRC Press.

See Also

Other log-ratio transformations: [transform_alr\(\)](#), [transform_clr\(\)](#), [transform_ilr\(\)](#), [transform_lr\(\)](#), [transform_plr\(\)](#)

Examples

```
## Data from Aitchison 1986
data("hongite")

## Coerce to compositional data
coda <- as_composition(hongite)

## Centered log-ratio
clr <- transform_clr(coda)

## Inverse transformation
inv_clr <- transform_inverse(clr)
all.equal(coda, inv_clr)
```

transform_lr	<i>Pairwise Log-Ratios (LR)</i>
--------------	---------------------------------

Description

Computes all pairwise log-ratio transformation.

Usage

```
transform_lr(object, ...)
```

S4 method for signature 'CompositionMatrix'

```
transform_lr(object, weights = FALSE)
```

Arguments

object	A CompositionMatrix object.
...	Currently not used.
weights	A logical scalar: should varying weights (column means) be computed? If FALSE (the default), equally-weighted parts are used. Alternatively, a positive numeric vector of weights can be specified (will be rescaled to sum to 1). Weights will be used internally by other methods (e.g. variance()).

Value

A [LR](#) object.

Author(s)

N. Frerebeau

References

- Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. London: Chapman and Hall.
- Greenacre, M. J. (2019). *Compositional Data Analysis in Practice*. Boca Raton: CRC Press.
- Greenacre, M. J. (2021). Compositional Data Analysis. *Annual Review of Statistics and Its Application*, 8(1): 271-299. doi:[10.1146/annurevstatistics042720124436](https://doi.org/10.1146/annurevstatistics042720124436).

See Also

Other log-ratio transformations: [transform_alr\(\)](#), [transform_clr\(\)](#), [transform_ilr\(\)](#), [transform_inverse\(\)](#), [transform_plr\(\)](#)

Examples

```
## Data from Aitchison 1986
data("hongite")

## Coerce to compositional data
coda <- as_composition(hongite)

## Pairwise log-ratio
lr <- transform_lr(coda)
```

transform_plr	<i>Pivot Log-Ratios (PLR)</i>
---------------	-------------------------------

Description

Computes PLR transformations.

Usage

```
transform_plr(object, ...)

## S4 method for signature 'CompositionMatrix'
transform_plr(object, pivot = 1)
```

Arguments

object	A CompositionMatrix object.
...	Currently not used.
pivot	An integer giving the index of the pivotal variable.

Value

A [PLR](#) object.

Author(s)

N. Frerebeau

References

Fišerová, E. & Hron, K. (2011). On the Interpretation of Orthonormal Coordinates for Compositional Data. *Mathematical Geosciences*, 43(4), 455-468. doi:10.1007/s110040119333x.

Greenacre, M. J. (2019). *Compositional Data Analysis in Practice*. Boca Raton: CRC Press.

Greenacre, M. J. (2021). Compositional Data Analysis. *Annual Review of Statistics and Its Application*, 8(1): 271-299. doi:10.1146/annurevstatistics042720124436.

Hron, K., Filzmoser, P., de Caritat, P., Fišerová, E. & Gardlo, A. (2017). Weighted Pivot Coordinates for Compositional Data and Their Application to Geochemical Mapping. *Mathematical Geosciences*, 49(6), 797-814. doi:10.1007/s110040179684z.

See Also

Other log-ratio transformations: [transform_alr\(\)](#), [transform_clr\(\)](#), [transform_ilr\(\)](#), [transform_inverse\(\)](#), [transform_lr\(\)](#)

Examples

```
## Data from Aitchison 1986
data("hongite")

## Coerce to compositional data
coda <- as_composition(hongite)

## Isometric log-ratio
ilr <- transform_ilr(coda)
plr <- transform_plr(coda)

## Inverse transformation
inv_ilr <- transform_inverse(ilr)
all.equal(coda, inv_ilr)

inv_plr <- transform_inverse(plr)
all.equal(coda, inv_plr)
```

variance

Log-Ratios Variances

Description

Computes log-ratio (weighted) variances.

Usage

```
variance(x, ...)

## S4 method for signature 'LogRatio'
variance(x, row_weights = NULL, column_weights = TRUE)
```

Arguments

`x` A [CompositionMatrix](#) object.

`...` Currently not used.

`row_weights` A [numeric](#) vector of row weights. If NULL (the default), equal weights are used.

`column_weights` A [logical](#) scalar: should the weights of the log-ratio be used? If FALSE, equally-weighted parts are used. Alternatively, a positive [numeric](#) vector of weights can be specified.

Value

A [numeric](#) vector of individual variances.

Author(s)

N. Frerebeau

References

Greenacre, M. J. (2019). *Compositional Data Analysis in Practice*. Boca Raton: CRC Press.

See Also

Other statistics: [aggregate\(\)](#), [condense\(\)](#), [covariance\(\)](#), [dist](#), [mahalanobis\(\)](#), [margin\(\)](#), [mean\(\)](#), [pip\(\)](#), [quantile\(\)](#), [scale\(\)](#), [variance_total\(\)](#), [variation\(\)](#)

Examples

```
## Data from Aitchison 1986
data("hongite")

## Coerce to compositional data
coda <- as_composition(hongite)

## Total variance (1)
variance_total(coda)

## Metric standard deviation
variance_total(coda, sd = TRUE)

## CLR transformation
clr <- transform_clr(coda)

## Individual log-ratio variances
```

```

variance(clr)

## Total log-ratio variance (2)
variance_total(clr)

## Proportionality between (1) and (2)
## See Aitchison 1997
variance_total(coda) * (1 / ncol(coda)) * (1 - (1 / nrow(coda)))

```

variance_total	<i>Total Variance</i>
----------------	-----------------------

Description

Computes the total (or metric) variance, a global measure of spread.

Usage

```

variance_total(x, ...)

## S4 method for signature 'CompositionMatrix'
variance_total(x, sd = FALSE)

## S4 method for signature 'LogRatio'
variance_total(x, row_weights = NULL, column_weights = TRUE)

```

Arguments

x	A CompositionMatrix object.
...	Currently not used.
sd	A logical scalar: should the metric standard deviation be returned instead of the metric variance?
row_weights	A numeric vector of row weights. If NULL (the default), equal weights are used.
column_weights	A logical scalar: should the weights of the log-ratio be used? If FALSE, equally-weighted parts are used. Alternatively, a positive numeric vector of weights can be specified.

Details

Two methods are available, see below.

Value

A [numeric](#) vector.

Methods (by class)

- `variance_total(CompositionMatrix)`: The total variance of compositional data is the trace of the [centred log-ratio covariance](#) matrix (i.e. `totvar1` in Aitchison 1997).
- `variance_total(LogRatio)`: Computes the total log-ratio variance. This is identical to the weighted sum-of-squared distances between samples (i.e. `totvar2` in Aitchison 1997).

Author(s)

N. Frerebeau

References

- Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. London: Chapman and Hall, p. 64-91.
- Aitchison, J. (1997). The One-Hour Course in Compositional Data Analysis or Compositional Data Analysis Is Simple. In V. Pawlowsky-Glahn (ed.), *IAMG'97*. Barcelona: International Center for Numerical Methods in Engineering (CIMNE), p. 3-35.
- Boogaart, K. G. van den & Tolosana-Delgado, R. (2013). *Analyzing Compositional Data with R*. Berlin Heidelberg: Springer-Verlag. doi:10.1007/9783642368097.
- Greenacre, M. J. (2019). *Compositional Data Analysis in Practice*. Boca Raton: CRC Press.
- Hron, K. & Kubáček, L. (2011). Statistical Properties of the Total Variation Estimator for Compositional Data. *Metrika*, 74 (2): 221-230. doi:10.1007/s0018401002993.
- Pawlowsky-Glahn, V. & Egozcue, J. J. (2001). Geometric Approach to Statistical Analysis on the Simplex. *Stochastic Environmental Research and Risk Assessment*, 15(5): 384-398. doi:10.1007/s004770100077.

See Also

Other statistics: [aggregate\(\)](#), [condense\(\)](#), [covariance\(\)](#), [dist](#), [mahalanobis\(\)](#), [margin\(\)](#), [mean\(\)](#), [pip\(\)](#), [quantile\(\)](#), [scale\(\)](#), [variance\(\)](#), [variation\(\)](#)

Examples

```
## Data from Aitchison 1986
data("hongite")

## Coerce to compositional data
coda <- as_composition(hongite)

## Total variance (1)
variance_total(coda)

## Metric standard deviation
variance_total(coda, sd = TRUE)

## CLR transformation
clr <- transform_clr(coda)
```

```
## Individual log-ratio variances
variance(cclr)

## Total log-ratio variance (2)
variance_total(cclr)

## Proportionality between (1) and (2)
## See Aitchison 1997
variance_total(coda) * (1 / ncol(coda)) * (1 - (1 / nrow(coda)))
```

variation	<i>Variation Matrix</i>
-----------	-------------------------

Description

Computes the variation matrix (Aitchison 1986, definition 4.4).

Usage

```
variation(x, ...)
```

S4 method for signature 'CompositionMatrix'
variation(x)

Arguments

x A [CompositionMatrix](#) object.
... Currently not used.

Value

A [matrix](#).

Author(s)

N. Frerebeau

References

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. London: Chapman and Hall, p. 64-91.
Greenacre, M. J. (2019). *Compositional Data Analysis in Practice*. Boca Raton: CRC Press.

See Also

Other statistics: [aggregate\(\)](#), [condense\(\)](#), [covariance\(\)](#), [dist](#), [mahalanobis\(\)](#), [margin\(\)](#), [mean\(\)](#), [pip\(\)](#), [quantile\(\)](#), [scale\(\)](#), [variance\(\)](#), [variance_total\(\)](#)

Examples

```
## Data from Aitchison 1986
data("hongite")

## Coerce to compositional data
coda <- as_composition(hongite)

## Variation matrix
## (Aitchison 1986, definition 4.4)
(varia <- variation(coda))

## Cluster dendrogram
d <- as.dist(varia)
h <- hclust(d, method = "ward.D2")
plot(h)

## Heatmap
stats::heatmap(
  varia,
  distfun = stats::as.dist,
  hclustfun = function(x) stats::hclust(x, method = "ward.D2"),
  symm = TRUE,
  scale = "none"
)
```


Index

- * **compositional data tools**
 - as_amounts, 6
 - as_composition, 7
- * **data summaries**
 - describe, 20
- * **datasets**
 - arctic, 4
 - boxite, 13
 - coxite, 19
 - hongite, 28
 - kongite, 29
 - lava, 30
 - predator, 48
 - slides, 54
- * **imputation methods**
 - missing, 34
 - replace_NA, 50
 - replace_zero, 51
- * **log-ratio transformations**
 - transform_alr, 59
 - transform_clr, 61
 - transform_ilr, 62
 - transform_inverse, 63
 - transform_lr, 65
 - transform_plr, 66
- * **multivariate analysis**
 - pca, 37
- * **mutators**
 - groups, 25
 - mutators, 36
 - totals, 58
- * **operations in the simplex**
 - arithmetic, 5
 - closure, 15
 - perturbation, 39
 - powering, 47
 - scalar, 52
- * **outlier detection methods**
 - detect_outlier, 20
 - plot_outlier, 44
- * **plot methods**
 - as_graph, 8
 - barplot, 10
 - hist, 27
 - plot, 41
 - plot_logratio, 43
- * **sourcing methods**
 - mix, 35
- * **statistics**
 - aggregate, 3
 - condense, 16
 - covariance, 17
 - dist, 23
 - mahalanobis, 30
 - margin, 31
 - mean, 32
 - pip, 40
 - quantile, 49
 - scale, 53
 - variance, 67
 - variance_total, 69
 - variation, 71
- * **subsetting methods**
 - bind, 12
 - extract, 24
 - split, 55
 - subset, 56
- * **tools**
 - chemistry, 14
 - [,CompositionMatrix,index,index,logical-method (subset), 56
 - [,CompositionMatrix,index,index,missing-method (subset), 56
 - [,CompositionMatrix,index,missing,logical-method (subset), 56
 - [,CompositionMatrix,index,missing,missing-method (subset), 56
 - [,CompositionMatrix,missing,index,logical-method

- (subset), 56
- [,CompositionMatrix,missing,index,missing-method (subset), 56
- [,CompositionMatrix,missing,missing,logical-method (subset), 56
- [,CompositionMatrix,missing,missing,missing-method (subset), 56
- [<- ,CompositionMatrix-method (subset), 56
- [[<- ,CompositionMatrix-method (subset), 56
- %perturbe% (arithmetic), 5
- %perturbe%,CompositionMatrix,CompositionMatrix-method (arithmetic), 5
- %power% (arithmetic), 5
- %power%,CompositionMatrix,numeric-method (arithmetic), 5
- %power%,numeric,CompositionMatrix-method (arithmetic), 5
- '%perturbe%',CompositionMatrix,CompositionMatrix-method (arithmetic), 5
- '%perturbe%'-method (arithmetic), 5
- '%power%',CompositionMatrix,numeric-method (arithmetic), 5
- '%power%',numeric,CompositionMatrix-method (arithmetic), 5
- '%power%'-method (arithmetic), 5
- aggregate, 3, 17, 18, 24, 31–33, 41, 49, 54, 68, 70, 71
- aggregate(), 17
- aggregate,CompositionMatrix-method (aggregate), 3
- ALR, 60
- any_assigned (groups), 25
- any_assigned,CompositionMatrix-method (groups), 25
- any_assigned,LogRatio-method (groups), 25
- any_assigned-method (groups), 25
- arctic, 4, 13, 19, 29, 30, 48, 55
- arithmetic, 5, 16, 40, 47, 53
- as.factor(f), 55
- as.integer(), 57
- as_amounts, 6, 8
- as_amounts,CompositionMatrix-method (as_amounts), 6
- as_amounts-method (as_amounts), 6
- as_composition, 6, 7
- as_composition,data.frame-method (as_composition), 7
- as_composition,matrix-method (as_composition), 7
- as_composition,numeric-method (as_composition), 7
- as_composition-method (as_composition), 7
- as_graph, 8, 11, 28, 42, 44
- as_graph,ALR-method (as_graph), 8
- as_graph,ILR-method (as_graph), 8
- as_graph,LR-method (as_graph), 8
- as_graph-method (as_graph), 8
- barplot, 9, 10, 28, 42, 44
- barplot,CompositionMatrix-method (barplot), 10
- base::split(), 55
- bind, 12, 25, 56, 57
- coxite, 5, 13, 19, 29, 30, 48, 55
- centred log-ratio covariance, 70
- character, 18, 21, 23, 24, 26, 27, 31, 32, 42–45, 57
- chemistry, 14
- closure, 5, 15, 40, 47, 53, 58
- closure,matrix-method (closure), 15
- closure,numeric-method (closure), 15
- closure-method (closure), 15
- CLR, 61
- CLR-transformed, 23
- cluster::daisy(), 35
- CompositionMatrix, 3, 5, 6, 8, 10, 12, 14, 17, 18, 20, 21, 23, 24, 27, 31–34, 38–40, 42, 47, 49–53, 55, 59, 61, 62, 64–66, 68, 69, 71
- condense, 4, 16, 18, 24, 31–33, 41, 49, 54, 68, 70, 71
- condense,CompositionMatrix-method (condense), 16
- condense-method (condense), 16
- covariance, 4, 17, 17, 24, 31–33, 41, 49, 54, 68, 70, 71
- covariance,ALR-method (covariance), 17
- covariance,CLR-method (covariance), 17
- covariance,CompositionMatrix-method (covariance), 17
- covariance-method (covariance), 17
- coxite, 5, 13, 19, 29, 30, 48, 55

- data.frame, [4](#), [7](#), [13](#), [19](#), [28–30](#), [48](#), [54](#)
- describe, [20](#)
- describe, CompositionMatrix-method
 - (describe), [20](#)
- detect_outlier, [20](#), [46](#)
- detect_outlier, CompositionMatrix, CompositionMatrix-method
 - (detect_outlier), [20](#)
- detect_outlier, CompositionMatrix, missing-method
 - (detect_outlier), [20](#)
- detect_outlier-method (detect_outlier), [20](#)
- dimensio::biplot(), [38](#)
- dimensio::PCA, [38](#)
- dimensio::pca(), [38](#)
- dimensio::screepplot(), [38](#)
- dimensio::viz_individuals(), [38](#)
- dimensio::viz_variables(), [38](#)
- dissimilarity matrix, [35](#)
- dist, [4](#), [17](#), [18](#), [23](#), [31–33](#), [41](#), [49](#), [54](#), [68](#), [70](#), [71](#)
- dist, CompositionMatrix-method (dist), [23](#)
- double, [7](#)
- extract, [12](#), [24](#), [56](#), [57](#)
- extract, CompositionMatrix-method
 - (extract), [24](#)
- extract-method (extract), [24](#)
- finite, [45](#)
- function, [3](#), [11](#), [42](#), [43](#)
- get (mutators), [36](#)
- graphical parameters, [42](#), [43](#), [45](#)
- graphics::hist(), [27](#)
- graphics::legend(), [44](#), [46](#)
- group, [24](#)
- groups, [25](#), [37](#), [59](#)
- groups, CompositionMatrix, ANY-method
 - (groups), [25](#)
- groups, CompositionMatrix, list-method
 - (groups), [25](#)
- groups, CompositionMatrix-method
 - (groups), [25](#)
- groups, LogRatio-method (groups), [25](#)
- groups, OutlierIndex-method (groups), [25](#)
- groups-method (groups), [25](#)
- groups<- (groups), [25](#)
- groups<-, CompositionMatrix, ANY-method
 - (groups), [25](#)
- groups<-, CompositionMatrix, list-method
 - (groups), [25](#)
- hist, [9](#), [11](#), [27](#), [42](#), [44](#)
- hist, CompositionMatrix-method (hist), [27](#)
- hist, LogRatio-method (hist), [27](#)
- integer, [8](#), [10](#), [23](#), [32](#), [38](#), [42](#), [43](#), [57](#), [59](#), [66](#)
- interaction(by), [3](#), [17](#)
- interaction(value), [26](#)
- is_assigned (groups), [25](#)
- is_assigned, CompositionMatrix-method
 - (groups), [25](#)
- is_assigned, LogRatio-method (groups), [25](#)
- is_assigned-method (groups), [25](#)
- is_element_major (chemistry), [14](#)
- is_element_major, CompositionMatrix-method
 - (chemistry), [14](#)
- is_element_major-method (chemistry), [14](#)
- is_element_minor (chemistry), [14](#)
- is_element_minor, CompositionMatrix-method
 - (chemistry), [14](#)
- is_element_minor-method (chemistry), [14](#)
- is_element_trace (chemistry), [14](#)
- is_element_trace, CompositionMatrix-method
 - (chemistry), [14](#)
- is_element_trace-method (chemistry), [14](#)
- is_outlier (detect_outlier), [20](#)
- is_outlier, OutlierIndex-method
 - (detect_outlier), [20](#)
- is_outlier-method (detect_outlier), [20](#)
- is_oxide (chemistry), [14](#)
- is_oxide, character-method (chemistry), [14](#)
- is_oxide, CompositionMatrix-method
 - (chemistry), [14](#)
- is_oxide-method (chemistry), [14](#)
- isopleuros::ternary_pairs(), [42](#)
- isopleuros::ternary_plot(), [42](#)
- kongite, [5](#), [13](#), [19](#), [29](#), [29](#), [30](#), [48](#), [55](#)
- labels, CompositionMatrix-method
 - (mutators), [36](#)
- labels, LogRatio-method (mutators), [36](#)
- lava, [5](#), [13](#), [19](#), [29](#), [30](#), [48](#), [55](#)
- list, [26](#), [44](#), [46](#)

- logical, [3](#), [8](#), [10](#), [11](#), [15](#), [16](#), [18](#), [21–23](#), [26](#), [27](#), [31](#), [38](#), [43–46](#), [49](#), [53](#), [55](#), [57](#), [60](#), [61](#), [65](#), [68](#), [69](#)
- LogRatio, [9](#), [38](#), [43](#), [64](#)
- LR, [65](#)
- lra (pca), [37](#)
- mahalanobis, [4](#), [17](#), [18](#), [24](#), [30](#), [32](#), [33](#), [41](#), [49](#), [54](#), [68](#), [70](#), [71](#)
- mahalanobis, CompositionMatrix-method (mahalanobis), [30](#)
- mahalanobis, ILR-method (mahalanobis), [30](#)
- margin, [4](#), [17](#), [18](#), [24](#), [31](#), [31](#), [33](#), [41](#), [49](#), [54](#), [68](#), [70](#), [71](#)
- margin, CompositionMatrix-method (margin), [31](#)
- margin-method (margin), [31](#)
- MASS::cov.rob(), [21](#), [31](#)
- matrix, [4](#), [6](#), [7](#), [18](#), [35](#), [41](#), [71](#)
- mean, [4](#), [17](#), [18](#), [24](#), [31](#), [32](#), [32](#), [41](#), [49](#), [54](#), [68](#), [70](#), [71](#)
- mean(), [17](#)
- mean, CompositionMatrix-method (mean), [32](#)
- missing, [34](#), [50](#), [52](#)
- mix, [35](#)
- mix, dist, dist-method (mix), [35](#)
- mix, matrix, matrix-method (mix), [35](#)
- mix-method (mix), [35](#)
- mutators, [26](#), [36](#), [59](#)
- NA, [34](#)
- numeric, [5](#), [6](#), [11](#), [14](#), [16](#), [21](#), [31](#), [33](#), [35](#), [38](#), [39](#), [43](#), [45](#), [47](#), [49–51](#), [53](#), [57](#), [60](#), [61](#), [65](#), [68](#), [69](#)
- OutlierIndex, [22](#), [45](#)
- pca, [37](#)
- pca, CompositionMatrix-method (pca), [37](#)
- pca, LogRatio-method (pca), [37](#)
- perturbation, [5](#), [16](#), [39](#), [47](#), [53](#)
- Perturbation operation, [5](#)
- perturbation, CompositionMatrix, CompositionMatrix-method (perturbation), [39](#)
- perturbation, CompositionMatrix, matrix-method (perturbation), [39](#)
- perturbation, CompositionMatrix, numeric-method (perturbation), [39](#)
- perturbation, numeric, numeric-method (perturbation), [39](#)
- perturbation-method (perturbation), [39](#)
- pip, [4](#), [17](#), [18](#), [24](#), [31–33](#), [40](#), [49](#), [54](#), [68](#), [70](#), [71](#)
- pip, CompositionMatrix-method (pip), [40](#)
- pip-method (pip), [40](#)
- plot, [9](#), [11](#), [28](#), [41](#), [44](#)
- plot, CompositionMatrix, missing-method (plot), [41](#)
- plot, LogRatio, missing-method (plot_logratio), [43](#)
- plot, OutlierIndex, missing-method (plot_outlier), [44](#)
- plot_logratio, [9](#), [11](#), [28](#), [42](#), [43](#)
- plot_outlier, [22](#), [44](#)
- PLR, [66](#)
- powering, [5](#), [16](#), [40](#), [47](#), [53](#)
- Powering operation, [5](#)
- powering, CompositionMatrix, numeric-method (powering), [47](#)
- powering, numeric, numeric-method (powering), [47](#)
- powering-method (powering), [47](#)
- predator, [5](#), [13](#), [19](#), [29](#), [30](#), [48](#), [55](#)
- quantile, [4](#), [17](#), [18](#), [24](#), [31–33](#), [41](#), [49](#), [54](#), [68](#), [70](#), [71](#)
- quantile, CompositionMatrix-method (quantile), [49](#)
- rbind2, CompositionMatrix, CompositionMatrix-method (bind), [12](#)
- replace_NA, [34](#), [50](#), [52](#)
- replace_NA, CompositionMatrix-method (replace_NA), [50](#)
- replace_zero, [34](#), [50](#), [51](#)
- replace_zero, CompositionMatrix-method (replace_zero), [51](#)
- scalar, [5](#), [16](#), [40](#), [47](#), [52](#)
- scalar, CompositionMatrix, CompositionMatrix-method (scalar), [52](#)
- scalar, numeric, numeric-method (scalar), [52](#)
- scalar-method (scalar), [52](#)
- scale, [4](#), [17](#), [18](#), [24](#), [31–33](#), [41](#), [49](#), [53](#), [68](#), [70](#), [71](#)
- scale, CompositionMatrix-method (scale), [53](#)
- set (mutators), [36](#)
- slides, [5](#), [13](#), [19](#), [29](#), [30](#), [48](#), [54](#)

- split, [12](#), [25](#), [55](#), [57](#)
- split, CompositionMatrix-method (split), [55](#)
- split, LogRatio-method (split), [55](#)
- stats::cov(), [18](#)
- stats::dist, [23](#), [35](#)
- stats::dist(), [23](#), [24](#)
- stats::mahalanobis(), [31](#)
- subset, [12](#), [25](#), [56](#), [56](#)

- totals, [26](#), [37](#), [58](#)
- totals, CompositionMatrix-method (totals), [58](#)
- totals, LogRatio-method (totals), [58](#)
- totals-method (totals), [58](#)
- totals<- (totals), [58](#)
- totals<-, CompositionMatrix-method (totals), [58](#)
- transform_alr, [59](#), [61](#), [63](#), [64](#), [66](#), [67](#)
- transform_alr, CLR-method (transform_alr), [59](#)
- transform_alr, CompositionMatrix-method (transform_alr), [59](#)
- transform_alr-method (transform_alr), [59](#)
- transform_clr, [60](#), [61](#), [63](#), [64](#), [66](#), [67](#)
- transform_clr, ALR-method (transform_clr), [61](#)
- transform_clr, CompositionMatrix-method (transform_clr), [61](#)
- transform_clr-method (transform_clr), [61](#)
- transform_ilr, [60](#), [61](#), [62](#), [64](#), [66](#), [67](#)
- transform_ilr, ALR, missing-method (transform_ilr), [62](#)
- transform_ilr, ALR-method (transform_ilr), [62](#)
- transform_ilr, CLR, missing-method (transform_ilr), [62](#)
- transform_ilr, CLR-method (transform_ilr), [62](#)
- transform_ilr, CompositionMatrix, missing-method (transform_ilr), [62](#)
- transform_ilr, CompositionMatrix-method (transform_ilr), [62](#)
- transform_ilr-method (transform_ilr), [62](#)
- transform_inverse, [60](#), [61](#), [63](#), [63](#), [66](#), [67](#)
- transform_inverse, ALR, missing-method (transform_inverse), [63](#)
- transform_inverse, CLR, missing-method (transform_inverse), [63](#)
- transform_inverse, ILR, missing-method (transform_inverse), [63](#)
- transform_inverse, matrix, ILR-method (transform_inverse), [63](#)
- transform_inverse-method (transform_inverse), [63](#)
- transform_lr, [60](#), [61](#), [63](#), [64](#), [65](#), [67](#)
- transform_lr, CompositionMatrix-method (transform_lr), [65](#)
- transform_lr-method (transform_lr), [65](#)
- transform_plr, [60](#), [61](#), [63](#), [64](#), [66](#), [66](#)
- transform_plr, CompositionMatrix-method (transform_plr), [66](#)
- transform_plr-method (transform_plr), [66](#)
- variance, [4](#), [17](#), [18](#), [24](#), [31–33](#), [41](#), [49](#), [54](#), [67](#), [70](#), [71](#)
- variance(), [60](#), [65](#)
- variance, LogRatio-method (variance), [67](#)
- variance-method (variance), [67](#)
- variance_total, [4](#), [17](#), [18](#), [24](#), [31–33](#), [41](#), [49](#), [54](#), [68](#), [69](#), [71](#)
- variance_total, CompositionMatrix-method (variance_total), [69](#)
- variance_total, LogRatio-method (variance_total), [69](#)
- variance_total-method (variance_total), [69](#)
- variation, [4](#), [17](#), [18](#), [24](#), [31–33](#), [41](#), [49](#), [54](#), [68](#), [70](#), [71](#)
- variation matrix, [40](#)
- variation, CompositionMatrix-method (variation), [71](#)
- variation-method (variation), [71](#)

- weights, ALR-method (mutators), [36](#)
- weights, LogRatio-method (mutators), [36](#)
- weights, LR-method (mutators), [36](#)