

Package: nexus (via r-universe)

October 23, 2026

Title Sourcing Archaeological Materials by Chemical Composition

Version 0.6.1.9000

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://codeberg.org/tesselle/nexus>,
<https://packages.tesselle.org/nexus/>,
<https://tesselle.r-universe.dev/nexus>

BugReports <https://codeberg.org/tesselle/nexus/issues>

Depends R (>= 3.5), dimensio (>= 0.14.0)

Imports arkhe (>= 1.11.0), graphics, grDevices, isopleuros (>= 1.4.0),
khroma (>= 1.16.0), MASS, methods, stats, utils

Suggests cluster, folio (>= 1.5.0), fontquiver, igraph, knitr,
markdown, rsvg, svglite, tinysnapshot (>= 0.2.0), tinytest

VignetteBuilder knitr

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

X-schema.org-applicationCategory Archaeological Science

X-schema.org-isPartOf <https://www.tesselle.org>

X-schema.org-keywords archaeology, archaeological-science,
archaeometry, compositional-data, provenance-studies, r-package

Collate 'AllClasses.R' 'AllGenerics.R' 'aggregate.R' 'barplot.R'
'bind.R' 'boxplot.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' 'pairs.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'

Config/roxygen2/version 8.0.0

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

Date/Publication 2026-06-23 08:58:04 UTC

RemoteUrl <https://codeberg.org/tesselle/nexus>

RemoteRef HEAD

RemoteSha f7ce146051f876fbfcea57be2c09396508b1623c

Contents

aggregate	3
arctic	4
arithmetic	5
as.data.frame	6
as_amounts	7
as_composition	8
as_graph	10
barplot	11
bind	13
boxite	14
boxplot	15
chemistry	17
closure	19
condense	20
covariance	21
coxite	22
describe	23
detect_outlier	24
dist	26
group	27
group_names	29
group_split	31
group_subset	33
hist	34
hongite	35
is_assigned	36
kongite	37
labels	38
lava	39
mahalanobis	39

margin	40
mean	41
missing	43
mix	44
pairs	45
pca	46
perturbation	48
pip	50
plot	51
plot_outlier	53
powering	55
predator	56
quantile	57
replace_NA	58
replace_zero	59
scalar	60
scale	61
slides	62
subset	63
totals	65
transform_alr	66
transform_clr	67
transform_ilr	69
transform_inverse	71
transform_lr	72
transform_plr	74
variance	75
variance_total	76
variation	78
weights	80
Index	81

 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)

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

Arguments

x	A CompositionMatrix object.
by	A vector or a list of grouping elements, each as long as the variables in x (see group()).
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.data.frame

Coerce to a Data Frame

Description

Coerce to a Data Frame

Usage

```
## S4 method for signature 'CompositionMatrix'
as.data.frame(x, ..., group_var = ".group", group_after = 0)

## S4 method for signature 'LogRatio'
as.data.frame(x, ..., group_var = ".group", group_after = 0)

## S4 method for signature 'OutlierIndex'
as.data.frame(x, row.names = rownames(x), optional = FALSE, ...)
```

Arguments

x	An R object (typically, a CompositionMatrix object).
...	Further parameters to be passed to as.data.frame() .
group_var	A character string specifying the name of the column to create for group attribution (only used if x is grouped).
group_after	An integer specifying a subscript, after which the new group_var column is to be appended (only used if x is grouped).
row.names	A character vector giving the row names for the data frame, or NULL.
optional	A logical scalar: should the names of the variables in the data frame be checked? If FALSE then the names of the variables in the data frame are checked to ensure that they are syntactically valid variable names and are not duplicated.

Value

A [data.frame](#).

Author(s)

N. Frerebeau

Examples

```
## Create a count matrix
A1 <- matrix(data = sample(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 <- as.data.frame(B)
head(X)
```

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(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 <- as.data.frame(B)
head(X)
```

as_composition

Coerce to a Closed Compositional Matrix

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,
  autodetect = TRUE,
  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 and autodetect is TRUE (the default), all numeric 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.
autodetect	A logical scalar: should numeric variables be automatically used as compositional parts?
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(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 <- as.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\(\)](#), [boxplot\(\)](#), [hist\(\)](#), [pairs\(\)](#), [plot](#)

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,  
  ...,
```

```

order_columns = FALSE,
order_rows = NULL,
decreasing = TRUE,
names = TRUE,
space = 0.2,
offset = 0.025,
color = NULL,
border = NA,
axes = TRUE,
legend = TRUE
)

```

Arguments

height	A CompositionMatrix object.
...	Further graphical parameters.
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?
names	A logical scalar: should row names be displayed?
space	A length-one numeric vector giving 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 amount of space (as a fraction) left between groups (defaults to 0.025). Only used if groups is not NULL.
color	A (named) vector of colors (will be mapped to the group names of object) or a function that when called with a single argument (an integer specifying the number of colors) returns a 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: it results in a graphic being displayed (invisibly return height).

Author(s)

N. Frerebeau

See Also

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

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)

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

Arguments

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

Details

rbind2() combine by rows.

Value

A [CompositionMatrix](#) objects.

Author(s)

N. Frerebeau

See AlsoOther subsetting methods: [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, parts = c("Ca", "Fe", "Na"))

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

## Better to use grouped matrix
Y <- group(Y, by = X$groups)
(s2 <- group_split(Y))

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

boxite

*Boxite Mineralogy***Description**

Mineral compositions and depths of 25 specimens of boxite.

Usage

boxite

FormatA [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](#)

boxplot	<i>Boxplot of Log-Ratios</i>
---------	------------------------------

Description

Displays box-and-whisker plots of the given (grouped) values.

Usage

```
## S4 method for signature 'LogRatio'
boxplot(
  x,
  ...,
  range = 1.5,
  width = NULL,
  varwidth = FALSE,
  notch = FALSE,
  outline = TRUE,
  plot = TRUE,
  horizontal = FALSE,
  xlab = NULL,
  ylab = NULL,
  main = NULL,
  sub = NULL,
  ann = graphics::par("ann")
)

## S4 method for signature 'GroupedLogRatio'
boxplot(
  x,
  ...,
  range = 1.5,
  width = NULL,
```

```

varwidth = FALSE,
notch = FALSE,
outline = TRUE,
plot = TRUE,
horizontal = FALSE,
color = NULL,
xlab = NULL,
ylab = NULL,
main = NULL,
sub = NULL,
ann = graphics::par("ann"),
legend = list(x = "topright")
)

```

Arguments

x	A LogRatio object.
...	Further graphical parameters.
range	A length-one numeric vector specifying how far the plot whiskers extend out from the box (see graphics::boxplot()).
width	A numeric vector giving the relative widths of the boxes making up the plot.
varwidth	A logical scalar: should the boxes be drawn with widths proportional to the square-roots of the number of observations in the groups?
notch	A logical scalar: should a notch be drawn in each side of the boxes?
outline	A logical scalar: should the outliers be drawn?
plot	A logical scalar: should a boxplot be produced? If FALSE, the summaries which the boxplots are based on are invisibly returned.
horizontal	A logical scalar: should the boxplots be horizontal?
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?
color	A (named) vector of colors (will be mapped to the group names of object) or a function that when called with a single argument (an integer specifying the number of colors) returns a vector of colors.
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

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

Author(s)

N. Frerebeau

See Also`graphics::boxplot()`Other plot methods: `as_graph()`, `barplot()`, `hist()`, `pairs()`, `plot`**Examples**

```
## 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, parts = 3:22) # Coerce to compositional data

## Log ratio
clr <- transform_clr(coda)

## Boxplot
boxplot(clr)

## Use ceramic types for grouping
grp <- group(coda, by = kommos$type)
clr <- transform_clr(grp)
boxplot(clr)
```

chemistry*Chemical Elements and Oxides*

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 <code>CompositionMatrix</code> object.
...	Currently not used.
min	A length-one <code>numeric</code> vector specifying the lower bound for element identification.
max	A length-one <code>numeric</code> 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	<i>Closure Operation</i>
---------	--------------------------

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

x	A numeric vector or matrix.
...	Currently not used.
total	A numeric vector specifying the total amount to which the compositions should be closed (defaults to 1).
na.rm	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, ignore_na = FALSE, ignore_zero = TRUE, verbose = FALSE, ...)

## S4 method for signature 'GroupedComposition'
condense(
  x,
  by = NULL,
  ignore_na = FALSE,
  ignore_zero = TRUE,
  verbose = FALSE,
  ...
)

```

Arguments

x	A CompositionMatrix object.
...	Currently not used.
by	A vector or a list of grouping elements, each as long as the variables in x (see group()).
ignore_na	A logical scalar: should missing values be stripped before the computation proceeds?
ignore_zero	A logical scalar: should zeros be stripped before the computation proceeds?
verbose	A logical scalar: should R report extra information on progress?

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	<i>Covariance Matrix</i>
------------	--------------------------

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

Coxite Mineralogy

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

Data Description

Description

Describes an object.

Usage

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

```
## S4 method for signature 'GroupedComposition'  
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)

## Quick description
describe(coda)

## Group
coda <- group(coda, by = slides$slide)
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,
  ...,
  quantile = 0.975,
  robust = TRUE,
  method = c("mve", "mcd")
)

## 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](https://doi.org/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 <- group_subset(coda, which = "CJ")
out <- detect_outlier(coda, reference = ref, method = "mcd")
plot(out, type = "dotchart")
```

dist	<i>Distances</i>
------	------------------

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

x	A CompositionMatrix object.
method	A character string specifying the distance measure to be used. See stats::dist() for the available distances.
diag	A logical scalar indicating whether the diagonal of the distance matrix should be printed.
upper	A logical scalar indicating whether the upper triangle of the distance matrix should be printed.
p	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)
```

group

Group by One or More Variables

Description

Define or remove the (reference) groups to which the observations belong.

Usage

```

group(object, ...)

ungroup(object, ...)

## S4 method for signature 'CompositionMatrix'
group(object, by, verbose = getOption("nexus.verbose"), ...)

## S4 method for signature 'GroupedComposition'
group(object, by, add = FALSE, verbose = getOption("nexus.verbose"), ...)

## S4 method for signature 'GroupedComposition'
ungroup(object)

## S4 method for signature 'GroupedLR'
ungroup(object)

## S4 method for signature 'GroupedCLR'
ungroup(object)

## S4 method for signature 'GroupedALR'
ungroup(object)

## S4 method for signature 'GroupedILR'
ungroup(object)

## S4 method for signature 'GroupedPLR'
ungroup(object)

```

Arguments

object	An R object (typically, a CompositionMatrix object).
...	Further parameters to be passed to internal methods.
by	A possible value for the groups of object (typically, a character vector). If value is a list , interaction(by) defines the grouping.
verbose	A logical scalar: should R report extra information on progress?
add	A logical scalar. If TRUE, add to existing groups.

Details

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

Value

- `group()` returns a grouped object of the same sort as object.
- `ungroup()` returns an ungrouped object of the same sort as object.

Author(s)

N. Frerebeau

See AlsoOther grouping methods: [group_names\(\)](#), [group_split\(\)](#), [group_subset\(\)](#), [is_assigned\(\)](#)**Examples**

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

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

## Grouping metadata
group_levels(coda)

group_names(coda)

group_indices(coda)

group_rows(coda)

group_n(coda)

group_size(coda)
```

group_names*Grouping Metadata*

Description

Retrieve the (reference) groups to which the observations belong.

Usage

```
group_names(object)

group_levels(object)

group_factor(object, ...)

group_rows(object)

group_n(object)

group_size(object)
```

```
group_indices(object)

## S4 method for signature 'ReferenceGroups'
group_levels(object)

## S4 method for signature 'ReferenceGroups'
group_names(object)

## S4 method for signature 'ReferenceGroups'
group_factor(object, exclude = NA)

## S4 method for signature 'ReferenceGroups'
group_indices(object)

## S4 method for signature 'ReferenceGroups'
group_rows(object)

## S4 method for signature 'ReferenceGroups'
group_n(object)

## S4 method for signature 'ReferenceGroups'
group_size(object)
```

Arguments

object	A grouped R object.
...	Currently not used.
exclude	A character vector of values to be excluded when forming the set of levels.

Value

- `group_levels()` returns a [character](#) vector giving the group names.
- `group_size()` returns an [integer](#) vector giving the size of each group.
- `group_n()` gives the total number of groups.
- `group_names()` returns a [character](#) vector giving the name of the group that each observation belongs to.
- `group_factor()` returns a [factor](#) vector giving the name of the group that each observation belongs to.
- `group_indices()` returns an [integer](#) vector giving the group that each value belongs to.
- `group_rows()` returns a list of [integer](#) vectors giving the observation that each group contains.

Author(s)

N. Frerebeau

See Also

Other grouping methods: [group\(\)](#), [group_split\(\)](#), [group_subset\(\)](#), [is_assigned\(\)](#)

Examples

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

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

## Grouping metadata
group_levels(coda)

group_names(coda)

group_indices(coda)

group_rows(coda)

group_n(coda)

group_size(coda)
```

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

Description

Divides a compositional matrix by groups.

Usage

```
group_split(object, ...)
```

S4 method for signature 'CompositionMatrix'

```
split(x, f, drop = FALSE, ...)
```

S4 method for signature 'LogRatio'

```
split(x, f, drop = FALSE, ...)
```

S4 method for signature 'GroupedComposition'

```
group_split(object, ...)
```

S4 method for signature 'GroupedLogRatio'

```
group_split(object, ...)
```

Arguments

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

Value

A list of [CompositionMatrix](#) objects.

Author(s)

N. Frerebeau

See Also

Other grouping methods: [group\(\)](#), [group_names\(\)](#), [group_subset\(\)](#), [is_assigned\(\)](#)

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, parts = c("Ca", "Fe", "Na"))

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

## Better to use grouped matrix
Y <- group(Y, by = X$groups)
(s2 <- group_split(Y))

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

group_subset	<i>Group-based Subset</i>
--------------	---------------------------

Description

Group-based Subset

Usage

```
group_subset(object, ...)  
  
## S4 method for signature 'GroupedComposition'  
group_subset(object, which)  
  
## S4 method for signature 'GroupedLogRatio'  
group_subset(object, which)
```

Arguments

object	A GroupedComposition object.
...	Currently not used.
which	A character vector specifying the groups of object to extract.

Value

A [CompositionMatrix](#) object.

Author(s)

N. Frerebeau

See Also

Other grouping methods: [group\(\)](#), [group_names\(\)](#), [group_split\(\)](#), [is_assigned\(\)](#)

Examples

```
## Data from Aitchison 1986  
data("slides")  
  
## Coerce to compositional data  
coda <- as_composition(slides, groups = 2)  
  
## Grouping metadata  
group_levels(coda)  
  
group_names(coda)
```

```

group_indices(coda)

group_rows(coda)

group_n(coda)

group_size(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](https://doi.org/10.1016/j.scitotenv.2009.08.008).

See Also

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

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](#)

is_assigned

Grouped Data

Description

Retrieve the (reference) groups to which the observations belong.

Usage

```
is_assigned(object)

any_assigned(object)

all_assigned(object)

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

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

## S4 method for signature 'ReferenceGroups'
all_assigned(object)
```

Arguments

object A [grouped R](#) object.

Value

- `is_assigned()` returns a **logical** vector specifying whether or not an observation belongs to a group.
- `any_assigned()` returns an **logical** scalar specifying if any observation belongs to a group.
- `all_assigned()` returns an **logical** scalar specifying if all observations belong to a group.

Author(s)

N. Frerebeau

See Also

Other grouping methods: [group\(\)](#), [group_names\(\)](#), [group_split\(\)](#), [group_subset\(\)](#)

Examples

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

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

## Grouping metadata
group_levels(coda)

group_names(coda)

group_indices(coda)

group_rows(coda)

group_n(coda)

group_size(coda)
```

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](#)

labels	<i>Labels</i>
--------	---------------

Description

Find a suitable set of labels from an object.

Usage

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

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

Arguments

- object An object from which to get the labels.
- ... Currently not used.

Author(s)

N. Frerebeau

See Also

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

lava	<i>Skye Lavas Compositions</i>
------	--------------------------------

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	<i>Mahalanobis Distance</i>
-------------	-----------------------------

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, ..., ignore_na = FALSE, ignore_zero = TRUE)
```

Arguments

x	A <code>CompositionMatrix</code> object.
...	Currently not used.
ignore_na	A <code>logical</code> scalar: should <code>missing values</code> be stripped before the computation proceeds?
ignore_zero	A <code>logical</code> scalar: should zeros be stripped before the computation proceeds?

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 <- pcoa(mix1) # Multi-Dimensional Scaling
plot(mds1)
```

pairs

Plot Compositional Data

Description

Displays a matrix of ternary plots.

Usage

```
## S4 method for signature 'CompositionMatrix'
pairs(x, margin = NULL, ...)

## S4 method for signature 'GroupedComposition'
pairs(x, ..., margin = NULL, color = NULL, symbol = NULL)
```

Arguments

<code>x</code>	A <code>CompositionMatrix</code> object.
<code>margin</code>	A <code>character</code> string or an <code>integer</code> giving the index of the column to be used as the third part of the ternary plots. If <code>NULL</code> (the default), marginal compositions will be used (i.e. the geometric mean of the non-selected parts).
<code>...</code>	Further <code>graphical parameters</code> .
<code>color</code>	A (named) vector of colors (will be mapped to the group names of object) or a <code>function</code> that when called with a single argument (an integer specifying the number of colors) returns a vector of colors.
<code>symbol</code>	A (named) vector of colors (will be mapped to the group names of object).

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()`, `boxplot()`, `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, parts = 3:8, groups = 1)

## Use ceramic types for grouping
pairs(coda)

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

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
)
```

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](https://doi.org/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](https://doi.org/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, extra_quali = group_names(coda))
viz_variables(X)
```

perturbation

Perturbation Operation

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	<i>Proportionality Index of Parts (PIP)</i>
-----	---

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](#).

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 Log-Ratios

Description

Displays a scatter plot.

Usage

```
## S4 method for signature 'LogRatio,missing'
plot(
  x,
  ...,
  jitter_factor = 1,
  jitter_amount = NULL,
  xlab = NULL,
  ylab = NULL,
  main = NULL,
  sub = NULL,
  ann = graphics::par("ann"),
  axes = TRUE,
  frame.plot = axes
)
```

```
## S4 method for signature 'GroupedLogRatio,missing'
plot(
  x,
  ...,
  jitter_factor = 1,
  jitter_amount = NULL,
  color = NULL,
  symbol = NULL,
  xlab = NULL,
  ylab = NULL,
  main = NULL,
  sub = NULL,
  ann = graphics::par("ann"),
  axes = TRUE,
  frame.plot = axes,
  legend = list(x = "topright")
)
```

Arguments

<code>x</code>	A LogRatio object.
<code>...</code>	Further graphical parameters.
<code>jitter_factor, jitter_amount</code>	A length-one numeric vector specifying the amount of jitter (see jitter()).
<code>xlab, ylab</code>	A character vector giving the x and y axis labels.
<code>main</code>	A character string giving a main title for the plot.
<code>sub</code>	A character string giving a subtitle for the plot.
<code>ann</code>	A logical scalar: should the default annotation (title and x and y axis labels) appear on the plot?
<code>axes</code>	A logical scalar: should axes be drawn on the plot?
<code>frame.plot</code>	A logical scalar: should a box be drawn around the plot?
<code>color</code>	A (named) vector of colors (will be mapped to the group names of object) or a function that when called with a single argument (an integer specifying the number of colors) returns a vector of colors.
<code>symbol</code>	A (named) vector of colors (will be mapped to the group names of object).
<code>legend</code>	A list of additional arguments to be passed to graphics::legend() ; names of the list are used as argument names. If <code>NULL</code> , 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\(\)](#), [boxplot\(\)](#), [hist\(\)](#), [pairs\(\)](#)

Examples

```
## 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, parts = 3:22) # Coerce to compositional data

## Log ratio
clr <- transform_clr(coda)

## Boxplot
plot(clr)

## Use ceramic types for grouping
grp <- group(coda, by = kommos$type)
clr <- transform_clr(grp)
plot(clr)
```

plot_outlier

Plot Outliers

Description

Plot Outliers

Usage

```
## S4 method for signature 'OutlierIndex,missing'
plot(
  x,
  ...,
  type = c("dotchart", "distance"),
  robust = TRUE,
  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 <code>OutlierIndex</code> object.
...	Further parameters to be passed to <code>graphics::points()</code> .
type	A <code>character</code> 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 <code>logical</code> scalar: should robust Mahalanobis distances be displayed? Only used if type is "dotchart".
symbols	A length-three vector of symbol specification for non-outliers and outliers (resp.).
xlim	A length-two <code>numeric</code> vector giving the x limits of the plot. The default value, NULL, indicates that the range of the <code>finite</code> values to be plotted should be used.
ylim	A length-two <code>numeric</code> vector giving the y limits of the plot. The default value, NULL, indicates that the range of the <code>finite</code> values to be plotted should be used.
xlab, ylab	A <code>character</code> vector giving the x and y axis labels.
main	A <code>character</code> string giving a main title for the plot.
sub	A <code>character</code> string giving a subtitle for the plot.
ann	A <code>logical</code> scalar: should the default annotation (title and x and y axis labels) appear on the plot?
axes	A <code>logical</code> scalar: should axes be drawn on the plot?
frame.plot	A <code>logical</code> scalar: should a box be drawn around the plot?
panel.first	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 <code>list</code> 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

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

powering

Powering Operation

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

*Sample Quantiles***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	<i>Missing Values Replacement</i>
------------	-----------------------------------

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	<i>Zero-Replacement</i>
--------------	-------------------------

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](https://doi.org/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

<code>x, y</code>	A CompositionMatrix object.
<code>...</code>	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](https://doi.org/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](#)

subset	<i>Extract or Replace Parts of an Object</i>
--------	--

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 subsetting object of the same sort as `x`.

Author(s)

N. Frerebeau

See Also

Other subsetting methods: [bind](#)

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 object.

Value

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

Author(s)

N. Frerebeau

See Also

Other mutators: [labels\(\)](#), [weights\(\)](#)

Examples

```
## Create a count matrix
A1 <- matrix(data = sample(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 <- as.data.frame(B)
head(X)
```

transform_alr	<i>Additive Log-Ratios (ALR)</i>
---------------	----------------------------------

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 'GroupedComposition'
transform_alr(object, j = ncol(object), weights = FALSE)

## S4 method for signature 'CLR'
transform_alr(object, j = ncol(object))

## S4 method for signature 'GroupedCLR'
transform_alr(object, j = ncol(object), weights = FALSE)
```

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](https://doi.org/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 'GroupedComposition'
transform_clr(object, weights = FALSE)

## S4 method for signature 'ALR'
transform_clr(object)

## S4 method for signature 'GroupedALR'
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 'GroupedComposition'
transform_ilr(object)

## S4 method for signature 'CLR'
transform_ilr(object)

## S4 method for signature 'GroupedCLR'
transform_ilr(object)

## S4 method for signature 'ALR'
transform_ilr(object)

## S4 method for signature 'GroupedALR'
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., 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](https://doi.org/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](https://doi.org/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 'GroupedCLR,missing'  
transform_inverse(object)  
  
## S4 method for signature 'ALR,missing'  
transform_inverse(object)  
  
## S4 method for signature 'GroupedALR,missing'  
transform_inverse(object)  
  
## S4 method for signature 'ILR,missing'  
transform_inverse(object)  
  
## S4 method for signature 'GroupedILR,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)

## S4 method for signature 'GroupedComposition'
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)

## S4 method for signature 'GroupedComposition'
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	<i>Log-Ratios Variances</i>
----------	-----------------------------

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(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)))
```

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"
)
```

`weights`*Weights*

Description

Extract weights from an object.

Usage

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

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

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

Arguments

<code>object</code>	An object for which the extraction of weights is meaningful.
<code>...</code>	Currently not used.

Author(s)

N. Frerebeau

See Also

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

Index

- * **coercion methods**
 - as.data.frame, 6
- * **compositional data tools**
 - as_amounts, 7
 - as_composition, 8
- * **data summaries**
 - describe, 23
- * **datasets**
 - arctic, 4
 - boxite, 14
 - coxite, 22
 - hongite, 35
 - kongite, 37
 - lava, 39
 - predator, 56
 - slides, 62
- * **grouping methods**
 - group, 27
 - group_names, 29
 - group_split, 31
 - group_subset, 33
 - is_assigned, 36
- * **imputation methods**
 - missing, 43
 - replace_NA, 58
 - replace_zero, 59
- * **log-ratio transformations**
 - transform_alr, 66
 - transform_clr, 67
 - transform_ilr, 69
 - transform_inverse, 71
 - transform_lr, 72
 - transform_plr, 74
- * **multivariate analysis**
 - pca, 46
- * **mutators**
 - labels, 38
 - totals, 65
 - weights, 80
- * **operations in the simplex**
 - arithmetic, 5
 - closure, 19
 - perturbation, 48
 - powering, 55
 - scalar, 60
- * **outlier detection methods**
 - detect_outlier, 24
 - plot_outlier, 53
- * **plot methods**
 - as_graph, 10
 - barplot, 11
 - boxplot, 15
 - hist, 34
 - pairs, 45
 - plot, 51
- * **sourcing methods**
 - mix, 44
- * **statistics**
 - aggregate, 3
 - condense, 20
 - covariance, 21
 - dist, 26
 - mahalanobis, 39
 - margin, 40
 - mean, 41
 - pip, 50
 - quantile, 57
 - scale, 61
 - variance, 75
 - variance_total, 76
 - variation, 78
- * **subsetting methods**
 - bind, 13
 - subset, 63
- * **tools**
 - chemistry, 17
 - [, CompositionMatrix, index, index, logical-method (subset), 63

- [,CompositionMatrix,index,index,missing-method (subset), 63
- [,CompositionMatrix,index,missing,logical-method (subset), 63
- [,CompositionMatrix,index,missing,missing-method (subset), 63
- [,CompositionMatrix,missing,index,logical-method (subset), 63
- [,CompositionMatrix,missing,index,missing-method (subset), 63
- [,CompositionMatrix,missing,missing,logical-method (subset), 63
- [,CompositionMatrix,missing,missing,missing-method (subset), 63
- [<-,CompositionMatrix-method (subset), 63
- [[<-,CompositionMatrix-method (subset), 63
- %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
- aggregate(), 21, 22, 27, 40–42, 50, 57, 62, 76, 78, 79
- aggregate,CompositionMatrix-method (aggregate), 3
- aggregate,GroupedComposition-method (aggregate), 3
- all_assigned(is_assigned), 36
- all_assigned,ReferenceGroups-method (is_assigned), 36
- ALR, 67
- any_assigned(is_assigned), 36
- any_assigned,ReferenceGroups-method (is_assigned), 36
- any_assigned-method(is_assigned), 36
- arctic, 4, 15, 23, 36, 38, 39, 57, 63
- arithmetic, 5, 19, 49, 56, 61
- as.data.frame, 6
- as.data.frame(), 7
- as.data.frame,CompositionMatrix-method (as.data.frame), 6
- as.data.frame,LogRatio-method (as.data.frame), 6
- as.data.frame,OutlierIndex-method (as.data.frame), 6
- as.factor(f), 32
- as.integer(), 64
- as_amounts, 7
- as_amounts(), 9
- as_amounts,CompositionMatrix-method (as_amounts), 7
- as_amounts-method (as_amounts), 7
- as_composition, 8
- as_composition(), 8
- as_composition,data.frame-method (as_composition), 8
- as_composition,matrix-method (as_composition), 8
- as_composition,numeric-method (as_composition), 8
- as_composition-method (as_composition), 8
- as_graph, 10
- as_graph(), 12, 17, 35, 46, 53
- as_graph,ALR-method (as_graph), 10
- as_graph,ILR-method (as_graph), 10
- as_graph,LR-method (as_graph), 10
- as_graph-method (as_graph), 10
- barplot, 11
- barplot(), 11, 17, 35, 46, 53
- barplot,CompositionMatrix-method (barplot), 11
- base::split(), 32
- bind, 13, 64
- boxite, 5, 14, 23, 36, 38, 39, 57, 63
- boxplot, 15
- boxplot(), 11, 12, 35, 46, 53
- boxplot,GroupedLogRatio-method (boxplot), 15
- boxplot,LogRatio-method (boxplot), 15
- centred log-ratio covariance, 77

- character, [7](#), [16](#), [21](#), [25](#), [26](#), [28](#), [30](#), [33](#), [34](#),
[40](#), [41](#), [46](#), [52](#), [54](#), [64](#)
- chemistry, [17](#)
- closure, [19](#), [65](#)
- closure(), [6](#), [49](#), [56](#), [61](#)
- closure,matrix-method (closure), [19](#)
- closure,numeric-method (closure), [19](#)
- closure-method (closure), [19](#)
- CLR, [68](#)
- CLR-transformed, [27](#)
- cluster::daisy(), [44](#)
- CompositionMatrix, [4–9](#), [12](#), [13](#), [18](#), [20](#), [21](#),
[23](#), [25](#), [26](#), [28](#), [32–34](#), [40–43](#), [46](#), [47](#),
[49](#), [50](#), [55](#), [57–61](#), [66](#), [68](#), [69](#), [71](#),
[73–75](#), [77](#), [78](#)
- condense, [20](#)
- condense(), [4](#), [22](#), [27](#), [40–42](#), [50](#), [57](#), [62](#), [76](#),
[78](#), [79](#)
- condense,CompositionMatrix-method
(condense), [20](#)
- condense,GroupedComposition-method
(condense), [20](#)
- condense-method (condense), [20](#)
- covariance, [21](#)
- covariance(), [4](#), [21](#), [27](#), [40–42](#), [50](#), [57](#), [62](#),
[76](#), [78](#), [79](#)
- covariance,ALR-method (covariance), [21](#)
- covariance,CLR-method (covariance), [21](#)
- covariance,CompositionMatrix-method
(covariance), [21](#)
- covariance-method (covariance), [21](#)
- coxite, [5](#), [15](#), [22](#), [36](#), [38](#), [39](#), [57](#), [63](#)
- data.frame, [5](#), [7](#), [9](#), [14](#), [23](#), [36](#), [38](#), [39](#), [56](#), [62](#)
- describe, [23](#)
- describe,CompositionMatrix-method
(describe), [23](#)
- describe,GroupedComposition-method
(describe), [23](#)
- detect_outlier, [24](#)
- detect_outlier(), [55](#)
- detect_outlier,CompositionMatrix,CompositionMatrix-method
(detect_outlier), [24](#)
- detect_outlier,CompositionMatrix,missing-method
(detect_outlier), [24](#)
- detect_outlier-method (detect_outlier),
[24](#)
- dimensio::biplot(), [48](#)
- dimensio::PCA, [47](#)
- dimensio::pca(), [47](#), [48](#)
- dimensio::screeplot(), [48](#)
- dimensio::viz_individuals(), [48](#)
- dimensio::viz_variables(), [48](#)
- dissimilarity matrix, [44](#)
- dist, [26](#)
- dist(), [4](#), [21](#), [22](#), [40–42](#), [50](#), [57](#), [62](#), [76](#), [78](#), [79](#)
- dist,CompositionMatrix-method (dist), [26](#)
- factor, [30](#)
- finite, [54](#)
- function, [4](#), [12](#), [16](#), [46](#), [52](#)
- get (weights), [80](#)
- graphical parameters, [46](#)
- graphics::boxplot(), [16](#), [17](#)
- graphics::hist(), [34](#)
- graphics::legend(), [16](#), [52](#), [54](#)
- graphics::points(), [54](#)
- group, [27](#)
- group(), [4](#), [20](#), [31–33](#), [37](#)
- group,CompositionMatrix-method (group),
[27](#)
- group,GroupedComposition-method
(group), [27](#)
- group-method (group), [27](#)
- group_factor (group_names), [29](#)
- group_factor,ReferenceGroups-method
(group_names), [29](#)
- group_factor-method (group_names), [29](#)
- group_indices (group_names), [29](#)
- group_indices,ReferenceGroups-method
(group_names), [29](#)
- group_indices-method (group_names), [29](#)
- group_levels (group_names), [29](#)
- group_levels,ReferenceGroups-method
(group_names), [29](#)
- group_levels-method (group_names), [29](#)
- group_n (group_names), [29](#)
- group_n,ReferenceGroups-method
(group_names), [29](#)
- group_n-method (group_names), [29](#)
- group_names, [29](#)
- group_names(), [29](#), [32](#), [33](#), [37](#)
- group_names,ReferenceGroups-method
(group_names), [29](#)
- group_names-method (group_names), [29](#)
- group_rows (group_names), [29](#)

- group_rows, ReferenceGroups-method (group_names), 29
- group_rows-method (group_names), 29
- group_size (group_names), 29
- group_size, ReferenceGroups-method (group_names), 29
- group_size-method (group_names), 29
- group_split, 31
- group_split(), 29, 31, 33, 37
- group_split, GroupedComposition-method (group_split), 31
- group_split, GroupedLogRatio-method (group_split), 31
- group_split-method (group_split), 31
- group_subset, 33
- group_subset(), 29, 31, 32, 37
- group_subset, GroupedComposition-method (group_subset), 33
- group_subset, GroupedLogRatio-method (group_subset), 33
- group_subset-method (group_subset), 33
- grouped, 7, 30, 36
- GroupedComposition, 33
- groups, 33
- hist, 34
- hist(), 11, 12, 17, 46, 53
- hist, CompositionMatrix-method (hist), 34
- hongite, 5, 15, 23, 35, 38, 39, 57, 63
- ILR, 40, 70
- integer, 7, 9, 12, 26, 30, 41, 46, 47, 64, 66, 74
- interaction(by), 28
- is_assigned, 36
- is_assigned(), 29, 31–33
- is_assigned, ReferenceGroups-method (is_assigned), 36
- is_assigned-method (is_assigned), 36
- is_element_major (chemistry), 17
- is_element_major, CompositionMatrix-method (chemistry), 17
- is_element_major-method (chemistry), 17
- is_element_minor (chemistry), 17
- is_element_minor, CompositionMatrix-method (chemistry), 17
- is_element_minor-method (chemistry), 17
- is_element_trace (chemistry), 17
- is_element_trace, CompositionMatrix-method (chemistry), 17
- is_element_trace-method (chemistry), 17
- is_outlier (detect_outlier), 24
- is_outlier, OutlierIndex-method (detect_outlier), 24
- is_outlier-method (detect_outlier), 24
- is_oxide (chemistry), 17
- is_oxide, character-method (chemistry), 17
- is_oxide, CompositionMatrix-method (chemistry), 17
- is_oxide-method (chemistry), 17
- isopleuros::ternary_pairs(), 46
- isopleuros::ternary_plot(), 46
- jitter(), 52
- kongite, 5, 15, 23, 36, 37, 39, 57, 63
- labels, 38
- labels(), 65, 80
- labels, CompositionMatrix-method (labels), 38
- labels, LogRatio-method (labels), 38
- lava, 5, 15, 23, 36, 38, 39, 57, 63
- list, 16, 28, 52, 54
- logical, 4, 7, 9, 12, 16, 18–21, 25, 26, 28, 32, 34, 35, 37, 40, 42, 47, 52, 54, 57, 61, 64, 66, 68, 73, 75, 77
- LogRatio, 10, 16, 47, 52, 71
- LR, 73
- lra (pca), 46
- mahalanobis, 39
- mahalanobis(), 4, 21, 22, 27, 41, 42, 50, 57, 62, 76, 78, 79
- mahalanobis, CompositionMatrix-method (mahalanobis), 39
- mahalanobis, ILR-method (mahalanobis), 39
- margin, 40
- margin(), 4, 21, 22, 27, 40, 42, 50, 57, 62, 76, 78, 79
- margin, CompositionMatrix-method (margin), 40
- margin-method (margin), 40
- MASS::cov.rob(), 25, 40
- matrix, 4, 8, 9, 21, 44, 50, 79
- mean, 41
- mean(), 4, 21, 22, 27, 40, 41, 50, 57, 62, 76, 78, 79

- mean, CompositionMatrix-method (mean), 41
- missing, 43, 58, 60
- missing values, 20, 42
- mix, 44
- mix, dist, dist-method (mix), 44
- mix, matrix, matrix-method (mix), 44
- mix-method (mix), 44
- NA, 43
- numeric, 5, 6, 8, 12, 16, 18, 19, 25, 40, 42, 44, 47, 49, 52, 54–59, 61, 64, 66, 68, 73, 75, 77
- OutlierIndex, 25, 54
- pairs, 45
- pairs(), 11, 12, 17, 35, 53
- pairs, CompositionMatrix-method (pairs), 45
- pairs, GroupedComposition-method (pairs), 45
- pca, 46
- pca, CompositionMatrix-method (pca), 46
- pca, LogRatio-method (pca), 46
- perturbation, 48
- Perturbation operation, 5
- perturbation(), 6, 19, 56, 61
- perturbation, CompositionMatrix, CompositionMatrix-method (perturbation), 48
- perturbation, CompositionMatrix, matrix-method (perturbation), 48
- perturbation, CompositionMatrix, numeric-method (perturbation), 48
- perturbation, numeric, numeric-method (perturbation), 48
- perturbation-method (perturbation), 48
- pip, 50
- pip(), 4, 21, 22, 27, 40–42, 57, 62, 76, 78, 79
- pip, CompositionMatrix-method (pip), 50
- pip-method (pip), 50
- plot, 11, 12, 17, 35, 46, 51
- plot, GroupedLogRatio, missing-method (plot), 51
- plot, LogRatio, missing-method (plot), 51
- plot, OutlierIndex, missing-method (plot_outlier), 53
- plot_outlier, 26, 53
- PLR, 74
- powering, 55
- Powering operation, 5
- powering(), 6, 19, 49, 61
- powering, CompositionMatrix, numeric-method (powering), 55
- powering, numeric, numeric-method (powering), 55
- powering-method (powering), 55
- predator, 5, 15, 23, 36, 38, 39, 56, 63
- quantile, 57
- quantile(), 4, 21, 22, 27, 40–42, 50, 62, 76, 78, 79
- quantile, CompositionMatrix-method (quantile), 57
- rbind2, CompositionMatrix, CompositionMatrix-method (bind), 13
- rbind2, GroupedComposition, GroupedComposition-method (bind), 13
- replace_NA, 43, 58, 60
- replace_NA, CompositionMatrix-method (replace_NA), 58
- replace_zero, 43, 58, 59
- replace_zero, CompositionMatrix-method (replace_zero), 59
- scalar, 60
- scalar(), 6, 19, 49, 56
- scalar, CompositionMatrix, CompositionMatrix-method (scalar), 60
- scalar, numeric, numeric-method (scalar), 60
- scalar-method (scalar), 60
- scale, 61
- scale(), 4, 21, 22, 27, 40–42, 50, 57, 76, 78, 79
- scale, CompositionMatrix-method (scale), 61
- set (weights), 80
- slides, 5, 15, 23, 36, 38, 39, 57, 62
- split, CompositionMatrix-method (group_split), 31
- split, LogRatio-method (group_split), 31
- stats::cov(), 21
- stats::dist, 27, 44
- stats::dist(), 26, 27
- stats::mahalanobis(), 40
- subset, 14, 63
- totals, 65

- `totals()`, [38](#), [80](#)
- `totals`, `CompositionMatrix`-method
(`totals`), [65](#)
- `totals`, `LogRatio`-method (`totals`), [65](#)
- `totals`-method (`totals`), [65](#)
- `totals<-` (`totals`), [65](#)
- `totals<-`, `CompositionMatrix`-method
(`totals`), [65](#)
- `transform_alr`, [66](#)
- `transform_alr()`, [68](#), [70](#), [72–74](#)
- `transform_alr`, `CLR`-method
(`transform_alr`), [66](#)
- `transform_alr`, `CompositionMatrix`-method
(`transform_alr`), [66](#)
- `transform_alr`, `GroupedCLR`-method
(`transform_alr`), [66](#)
- `transform_alr`, `GroupedComposition`-method
(`transform_alr`), [66](#)
- `transform_alr`-method (`transform_alr`), [66](#)
- `transform_clr`, [67](#)
- `transform_clr()`, [67](#), [70](#), [72–74](#)
- `transform_clr`, `ALR`-method
(`transform_clr`), [67](#)
- `transform_clr`, `CompositionMatrix`-method
(`transform_clr`), [67](#)
- `transform_clr`, `GroupedALR`-method
(`transform_clr`), [67](#)
- `transform_clr`, `GroupedComposition`-method
(`transform_clr`), [67](#)
- `transform_clr`-method (`transform_clr`), [67](#)
- `transform_ilr`, [69](#)
- `transform_ilr()`, [67](#), [68](#), [72–74](#)
- `transform_ilr`, `ALR`, missing-method
(`transform_ilr`), [69](#)
- `transform_ilr`, `ALR`-method
(`transform_ilr`), [69](#)
- `transform_ilr`, `CLR`, missing-method
(`transform_ilr`), [69](#)
- `transform_ilr`, `CLR`-method
(`transform_ilr`), [69](#)
- `transform_ilr`, `CompositionMatrix`, missing-method
(`transform_ilr`), [69](#)
- `transform_ilr`, `CompositionMatrix`-method
(`transform_ilr`), [69](#)
- `transform_ilr`, `GroupedALR`, missing-method
(`transform_ilr`), [69](#)
- `transform_ilr`, `GroupedALR`-method
(`transform_ilr`), [69](#)
- `transform_ilr`, `GroupedCLR`, missing-method
(`transform_ilr`), [69](#)
- `transform_ilr`, `GroupedCLR`-method
(`transform_ilr`), [69](#)
- `transform_ilr`, `GroupedComposition`, missing-method
(`transform_ilr`), [69](#)
- `transform_ilr`, `GroupedComposition`-method
(`transform_ilr`), [69](#)
- `transform_ilr`-method (`transform_ilr`), [69](#)
- `transform_inverse`, [71](#)
- `transform_inverse()`, [67](#), [68](#), [70](#), [73](#), [74](#)
- `transform_inverse`, `ALR`, missing-method
(`transform_inverse`), [71](#)
- `transform_inverse`, `CLR`, missing-method
(`transform_inverse`), [71](#)
- `transform_inverse`, `GroupedALR`, missing-method
(`transform_inverse`), [71](#)
- `transform_inverse`, `GroupedCLR`, missing-method
(`transform_inverse`), [71](#)
- `transform_inverse`, `GroupedILR`, missing-method
(`transform_inverse`), [71](#)
- `transform_inverse`, `ILR`, missing-method
(`transform_inverse`), [71](#)
- `transform_inverse`, `matrix`, `ILR`-method
(`transform_inverse`), [71](#)
- `transform_inverse`-method
(`transform_inverse`), [71](#)
- `transform_lr`, [72](#)
- `transform_lr()`, [67](#), [68](#), [70](#), [72](#), [74](#)
- `transform_lr`, `CompositionMatrix`-method
(`transform_lr`), [72](#)
- `transform_lr`, `GroupedComposition`-method
(`transform_lr`), [72](#)
- `transform_lr`-method (`transform_lr`), [72](#)
- `transform_plr`, [74](#)
- `transform_plr()`, [67](#), [68](#), [70](#), [72](#), [73](#)
- `transform_plr`, `CompositionMatrix`-method
(`transform_plr`), [74](#)
- `transform_plr`, `GroupedComposition`-method
(`transform_plr`), [74](#)
- `transform_plr`-method (`transform_plr`), [74](#)
- `ungroup` (`group`), [27](#)
- `ungroup`, `GroupedALR`-method (`group`), [27](#)
- `ungroup`, `GroupedCLR`-method (`group`), [27](#)
- `ungroup`, `GroupedComposition`-method
(`group`), [27](#)
- `ungroup`, `GroupedILR`-method (`group`), [27](#)
- `ungroup`, `GroupedLR`-method (`group`), [27](#)

ungroup, GroupedPLR-method (group), 27

variance, 75

variance(), 4, 21, 22, 27, 40–42, 50, 57, 62, 66, 73, 78, 79

variance, LogRatio-method (variance), 75

variance-method (variance), 75

variance_total, 76

variance_total(), 4, 21, 22, 27, 40–42, 50, 57, 62, 76, 79

variance_total, CompositionMatrix-method (variance_total), 76

variance_total, LogRatio-method (variance_total), 76

variance_total-method (variance_total), 76

variation, 78

variation matrix, 50

variation(), 4, 21, 22, 27, 40–42, 50, 57, 62, 76, 78

variation, CompositionMatrix-method (variation), 78

variation-method (variation), 78

weights, 80

weights(), 38, 65

weights, ALR-method (weights), 80

weights, LogRatio-method (weights), 80

weights, LR-method (weights), 80