Package 'nexus'

February 26, 2024

Title Sourcing Archaeological Materials by Chemical Composition

Version 0.2.0

Maintainer Nicolas Frerebeau <nicolas.frerebeau@u-bordeaux-montaigne.fr>

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

License GPL (>= 3)

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

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

Depends R (>= 3.5), dimensio (>= 0.6.0)

Imports arkhe (>= 1.6.0), graphics, grDevices, isopleuros (>= 1.2.0), methods, MASS, stats, utils

Suggests cluster, folio (>= 1.4.0), igraph, knitr, markdown, rsvg, svglite, tinysnapshot, tinytest

VignetteBuilder knitr

Encoding UTF-8

LazyData true

RoxygenNote 7.3.0

Collate 'AllClasses.R' 'AllGenerics.R' 'coerce.R' 'data.R' 'graph.R' 'mix.R' 'mutators.R' 'nexus-internal.R' 'nexus-package.R' 'outliers.R' 'pca.R' 'plot.R' 'reexport.R' 'replace.R' 'show.R' 'simplex.R' 'statistics.R' 'subset.R' 'summary.R' 'transform.R' 'validate.R' 'zzz.R'

NeedsCompilation no

Author Nicolas Frerebeau [aut, cre] (<https://orcid.org/0000-0001-5759-4944>, Université Bordeaux Montaigne), Anne Philippe [aut] (<https://orcid.org/0000-0002-5331-5087>, Université de Nantes),

Brice Lebrun [ctb] (<https://orcid.org/0000-0001-7503-8685>, Logo designer),
Arthur Leck [ctb] (<https://orcid.org/0009-0004-8819-9970>, Université Bordeaux Montaigne),
Université Bordeaux Montaigne [fnd],
CNRS [fnd]

Repository CRAN

Date/Publication 2024-02-26 17:20:02 UTC

R topics documented:

`aggregate` *Compute Summary Statistics of Data Subsets*

Description

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

Usage

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

Arguments

`x`	A `CompositionMatrix` object.
`by`	A vector or a list of grouping elements, each as long as the variables in `x`. The elements are coerced to factors before use.
`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: covariance(), dist, mahalanobis(), margin(), mean(), metric_var(), quantile(), scale(), variation()

Examples

```
## Create a data.frame
X <- data.frame(
  samples = c("A", "A", "A", "B", "B", "B", "C", "C", "C"),
  groups = c("X", "X", "X", NA, 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)

## Compositional mean by sample
aggregate(Y, by = get_samples(Y), FUN = mean)

## Metric variance by group
aggregate(Y, by = get_groups(Y), FUN = metric_var)
```

---

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: chemistry, hongite, petrography, slides

---

arithmetic *Operations in the Simplex*

Description

Operators performing operations in the simplex.

Usage

```
x %perturbe% y

x %power% y

## S4 method for signature 'CompositionMatrix,CompositionMatrix'
x %perturbe% y

## S4 method for signature 'CompositionMatrix,numeric'
x %power% y

## S4 method for signature 'numeric,CompositionMatrix'
x %power% y
```

Arguments

x	A CompositionMatrix object.
y	A CompositionMatrix object or a numeric vector.

Details

%perturbe% Perturbation operation.

%power% Powering operation.

Value

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

Author(s)

N. Frerebeau

See Also

Other operations in the simplex: closure(), perturbation(), powering(), scalar()

Examples

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

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

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

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

`as_amounts` *Coerce to Amounts*

Description

Coerce to Amounts

Usage

```
as_amounts(from, ...)

## S4 method for signature 'CompositionMatrix'
as_amounts(from)
```

Arguments

`from`	A `CompositionMatrix` object.
`...`	Currently not used.

Value

A `numeric` `matrix`.

Author(s)

N. Frerebeau

See Also

Other compositional data tools: `as_composition()`, `as_features()`

Examples

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

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

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

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

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

---

as_composition *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,
  codes = NULL,
  samples = NULL,
  groups = NULL,
  auto = getOption("nexus.autodetect"),
  verbose = getOption("nexus.verbose")
)
```

Arguments

`from`	A `matrix` or `data.frame` to be coerced.
`...`	Currently not used.

codes	An integer giving the index of the column to be used as laboratory codes (unique identifiers).
samples	An integer giving the index of the column to be used for sample identification: allows duplicates to identify replicated measurements. If NULL (the default), row names will be used as sample IDs.
groups	An integer giving the index of the column to be used to group the samples. If NULL (the default), no grouping is stored.
auto	A logical scalar: try to automatically detect codes, samples and groups columns?
verbose	A logical scalar: should R report extra information on progress?

Details

The CompositionMatrix class has special slots:

- codes for laboratory codes,
- samples for repeated measurements/observation,
- groups to group data by site/area.

When coercing a data.frame to a CompositionMatrix object, an attempt is made to automatically assign values to these slots by mapping column names (case insensitive, plural insensitive). This behavior can be disabled by setting options(nexus.autodetect = FALSE) or overridden by explicitly specifying the columns to be used.

See vignette("nexus").

Value

A CompositionMatrix object.

Note

All non-numeric variable will be removed.

Author(s)

N. Frerebeau

See Also

Other compositional data tools: as_amounts(), as_features()

Examples

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

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

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

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

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

`as_features` *Coerce to Features*

Description

Converts an object to a collection of features.

Usage

```
as_features(from, ...)

## S4 method for signature 'CompositionMatrix'
as_features(from)
```

Arguments

`from`	A `CompositionMatrix` object.
`...`	Currently not used.

Value

A `data.frame` with all informations as extra columns.

Author(s)

N. Frerebeau

See Also

Other compositional data tools: `as_amounts()`, `as_composition()`

Examples

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

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

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

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

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

---

as_graph *Graph of Log-ratios*

Description

Produces a graph of log-ratios.

Usage

```
as_graph(object, ...)

## S4 method for signature 'LR'
as_graph(object)

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

## S4 method for signature 'ILR'
as_graph(object)
```

Arguments

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

Value

An **igraph** graph object.

Author(s)

N. Frerebeau

See Also

Other plot methods: barplot(), hist(), plot_logratio, 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 = NULL,
  decreasing = FALSE,
  groups = get_groups(height),
  horiz = TRUE,
```

```
  xlab = NULL,
  ylab = NULL,
  main = NULL,
  sub = NULL,
  ann = graphics::par("ann"),
  axes = TRUE,
  col = grDevices::hcl.colors(ncol(height), "viridis"),
  legend = list()
)
```

Arguments

height	A CompositionMatrix object.
...	Further parameters to be passed to graphics::barplot().
order	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 be increasing or decreasing?
groups	A factor in the sense that as.factor(groups) defines the grouping. If set, a matrix of panels defined by groups will be drawn.
horiz	A logical scalar. If FALSE, the bars are drawn vertically with the first bar to the left. If TRUE (the default), the bars are drawn horizontally with the first at the bottom.
xlab, ylab	A character vector giving the x and y axis labels.
main	A character string giving a main title for the plot.
sub	A character string giving a subtitle for the plot.
ann	A logical scalar: should the default annotation (title and x and y axis labels) appear on the plot?
axes	A logical scalar: should axes be drawn on the plot?
col	A vector of colors for the bar components.
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

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

Author(s)

N. Frerebeau

See Also

Other plot methods: as_graph(), hist(), plot_logratio, plot()

Examples

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

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

## Bar plot
barplot(coda, order = 2)

## 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

barplot(coda, order = 1)
barplot(coda, order = 1, horiz = FALSE)
```

---

chemistry *Can Sora Chemical Data*

Description

Can Sora Chemical Data

Usage

```
chemistry
```

Format

A data.frame with 30 variables.

References

Cau, M.-A. (1999). Importaciones de cerámica tardorromana de cocina en las Iles Balears: el caso de Can Sora (Eivissa). In J. Capel Martinez, *Arqueometria y Arqueologia*, p. 197-219. Granada: Editorial Universidad de Granada. Monografica Arte y Arqueología 47.

See Also

Other datasets: arctic, hongite, petrography, slides

closure *Closure Operation*

Description

Closes compositions to sum up to 1.

Usage

```
closure(x, ...)

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

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

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)
```

`covariance` *Covariance Matrix*

Description

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

Usage

```
covariance(x, ...)

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

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

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

Arguments

`x`	A `CompositionMatrix` object.
`...`	Currently not used.
`center`	A `logical` scalar: should the *centered* 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()`, `dist`, `mahalanobis()`, `margin()`, `mean()`, `metric_var()`, `quantile()`, `scale()`, `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)
```

`dist` *Distances*

Description

Computes the log-ratio variance matrix.

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()`, `covariance()`, `mahalanobis()`, `margin()`, `mean()`, `metric_var()`, `quantile()`, `scale()`, `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)
```

---

groups *Working With Groups*

Description

Retrieves or defines the groups to which the observations belong.

Usage

```
any_assigned(x)

is_assigned(x)

get_groups(x)

set_groups(x) <- value

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

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

## S4 method for signature 'OutlierIndex'
is_assigned(x)

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

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

## S4 method for signature 'OutlierIndex'
any_assigned(x)

## S4 method for signature 'CompositionMatrix'
get_groups(x)

## S4 method for signature 'LogRatio'
get_groups(x)

## S4 method for signature 'OutlierIndex'
get_groups(x)

## S4 replacement method for signature 'CompositionMatrix'
set_groups(x) <- value
```

Arguments

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

Details

See vignette("nexus").

Value

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

Author(s)

N. Frerebeau

See Also

Other mutators: `identifiers`, `samples`, `split()`, `subset()`, `totals`

`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,
  ...,
  freq = FALSE,
  ncol = NULL,
  flip = FALSE,
  main = NULL,
  sub = NULL,
  ann = graphics::par("ann"),
  axes = TRUE,
  frame.plot = axes
)
```

Arguments

`x`	A `CompositionMatrix` object.
`...`	Further parameters to be passed to `graphics::hist()`.
`freq`	A `logical` scalar: should absolute frequencies (counts) be displayed (see `graphics::hist()`)?
`ncol`	An `integer` specifying the number of columns to use when `facet` is `"multiple"`. Defaults to 1 for up to 4 series, otherwise to 2.

flip	A logical scalar: should the y-axis (ticks and numbering) be flipped from side 2 (left) to 4 (right) from variable to variable?
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: is results in a graphic being displayed (invisibly return x).

Author(s)

N. Frerebeau

References

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

See Also

Other plot methods: as_graph(), barplot(), plot_logratio, plot()

Examples

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

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

## Boxplot plot
hist(coda)

hist(coda)
hist(coda[, 1, drop = FALSE])
```

hongite *Hongite Mineralogy*

Description

Mineral compositions of 25 rock specimens of hongite type.

Usage

```
hongite
```

Format

A data.frame with 5 variables (minerals):

A Albite (percent).

B Blandite (percent).

C Cornite (percent).

D Daubite (percent).

E Endite (percent).

References

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

See Also

Other datasets: arctic, chemistry, petrography, slides

identifiers *Unique Identifiers*

Description

Retrieves or defines the unique identifier (eg. laboratory codes) of each observation.

Usage

```
get_identifiers(x)

set_identifiers(x) <- value

## S4 method for signature 'CompositionMatrix'
get_identifiers(x)

## S4 method for signature 'LogRatio'
get_identifiers(x)

## S4 method for signature 'OutlierIndex'
get_identifiers(x)

## S4 replacement method for signature 'CompositionMatrix'
set_identifiers(x) <- value
```

Arguments

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

Details

See vignette("nexus").

Value

- set_identifiers() returns an object of the same sort as x with the new identifiers assigned.
- get_identifiers() returns a character vector giving the unique identifiers of x.

Author(s)

N. Frerebeau

See Also

Other mutators: groups, samples, split(), subset(), totals

---

mahalanobis *Mahalanobis Distance*

Description

Computes the squared Mahalanobis distance of all rows in x.

Usage

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

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

Arguments

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

Value

A numeric vector.

Author(s)

N. Frerebeau

See Also

stats::mahalanobis()

Other statistics: aggregate(), covariance(), dist, margin(), mean(), metric_var(), quantile(), scale(), 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(), covariance(), dist, mahalanobis(), mean(), metric_var(), quantile(), scale(), 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, ..., na.rm = FALSE)
```

Arguments

x	A CompositionMatrix object.
...	Currently not used.
na.rm	A logical scalar: should missing values be removed?

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(), covariance(), dist, mahalanobis(), margin(), metric_var(), quantile(), scale(), variation()

Examples

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

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

```
## Mean
mean(coda)

## Quantile
quantile(coda)

## Metric variance
metric_var(coda)

## Metric standard deviation
metric_sd(coda)
```

`metric_var` *Metric Variance and Standard Deviation*

Description

- `metric_var()` computes the metric variance (or total variance), i.e. a global measure of spread.
- `metric_sd()` computes the metric standard deviation.

Usage

```
metric_var(x, ...)

metric_sd(x, ...)

## S4 method for signature 'CompositionMatrix'
metric_var(x)

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

Arguments

`x`	A `CompositionMatrix` object.
`...`	Currently not used.

Details

The metric variance is the average of the CLR variances.

Value

A `numeric` vector.

Author(s)

N. Frerebeau

References

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

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()`, `covariance()`, `dist`, `mahalanobis()`, `margin()`, `mean()`, `quantile()`, `scale()`, `variation()`

Examples

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

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

## Mean
mean(coda)

## Quantile
quantile(coda)

## Metric variance
metric_var(coda)

## Metric standard deviation
metric_sd(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, two situations can be outlined:

- The presence of zeros: these are considered as observed quantities, but which happen to be below the detection limit (thus interpreted as small unknown values).
- The presence of missing values (NA): these indicate that the quantities in question have not been observed.

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 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

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

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

## First approach
mix1 <- mix(as.matrix(chem), as.matrix(petro), lambda = 2)
```

```
mds1 <- stats::cmdscale(mix1) # Multi-Dimensional Scaling
plot(mds1)
```

`outliers` *Outlier Detection*

Description

Outlier Detection

Usage

```
outliers(object, ...)

## S4 method for signature 'CompositionMatrix'
outliers(
  object,
  ...,
  groups = get_groups(object),
  robust = TRUE,
  method = c("mve", "mcd"),
  quantile = 0.975
)
```

Arguments

`object`	A `CompositionMatrix`.
`...`	Extra parameters to be passed to `MASS::cov.rob()`. Only used if `robust` is `TRUE`.
`groups`	A `factor` in the sense that `as.factor(groups)` defines the grouping. If set, XXX.
`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`.
`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

An `OutlierIndex` object.

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/s11004007914l5.

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

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

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

See Also

Other outlier detection methods: `plot_outliers`

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

## Detect outliers
out <- outliers(coda, groups = NULL, robust = FALSE)

plot(out) # Plot
plot(out, qq = TRUE) # Quantile-Quantile plot

## Detect outliers by group
out <- outliers(coda[, 1:15, drop = FALSE])
```

```
plot(out, ncol = 2) # Plot
plot(out, qq = TRUE, ncol = 4) # Quantile-Quantile plot
```

---

pca_coda *Principal Components Analysis*

Description

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

Usage

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

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

Arguments

object	A 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 package **dimensio** for details.

Author(s)

N. Frerebeau

References

Aitchison, J. and Greenacre, M. (2002). Biplots of compositional data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 51: 375-392. doi:10.1111/14679876.00275.

Filzmoser, P., Hron, K. and Reimann, C. (2009). Principal component analysis for compositional data with outliers. *Environmetrics*, 20: 621-632. doi:10.1002/env.966.

See Also

`dimensio::pca()`, `dimensio::biplot()`, `dimensio::screeplot()`, `dimensio::viz_individuals()`, `dimensio::viz_variables()`

Examples

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

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

## PCA
X <- pca(clr, scale = FALSE)

## Explore results
viz_individuals(X, highlight = get_groups(coda), pch = 16)
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 componentwise 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)
```

`petrography` *Can Sora Petrographic Data*

Description

Can Sora Petrographic Data

Usage

```
petrography
```

Format

A `data.frame` with 21 variables:

VAR1 Optical activity.

VAR2 Inclusion orientation.

VAR3 Void orientation.

VAR4 Texture.

VAR5 Special components.

VAR6 Plutonic rocks.

VAR7 Volcanic rocks.

VAR8 Metamorphic rocks.

VAR9 Sedimentary rocks.

VAR10 Quartz.

VAR11 Feldspar.

VAR12 Plagioclase.

VAR13 Pyroxenes.

VAR14 Amphiboles.

VAR15 Micas.

VAR16 Phyllosilicates.

VAR17 Carbonates.

VAR18 Other constituents.

VAR19 Packing.

References

Cau, M.-A., Day, P. M., Baxter, M. J., Papageorgiou, I., Iliopoulos, I. & Montana, G. (2004). Exploring Automatic Grouping Procedures in Ceramic Petrology. *Journal of Archaeological Science*, 31(9): 1325-1338. doi:10.1016/j.jas.2004.03.006.

See Also

Other datasets: `arctic`, `chemistry`, `hongite`, `slides`

`plot` *Plot Compositional Data*

Description

Displays a matrix of ternary plots.

Usage

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

Arguments

`x`	A `CompositionMatrix` object.
`...`	Further graphical parameters.
`margin`	A `character` string or an `integer` giving the index of the column to be used as the third part of the ternary plots. If `NULL` (the default), marginal compositions will be used (i.e. the geometric mean of the non-selected parts).

Value

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

Author(s)

N. Frerebeau

See Also

isopleuros::ternary_pairs(), isopleuros::ternary_plot()

Other plot methods: as_graph(), barplot(), hist(), plot_logratio

Examples

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

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

## Ternary plots
plot(coda)
```

---

plot_logratio *Plot Log-Ratios*

Description

Displays a density plot.

Usage

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

Arguments

x	A LogRatio object.
...	Further graphical parameters, particularly, border and col.
order	A logical scalar: should the ratio be ordered?
decreasing	A logical scalar: should the sort order be increasing or decreasing?
groups	A factor in the sense that as.factor(groups) defines the grouping. If set, a matrix of panels defined by groups will be drawn.
rug	A logical scalar: should a *rug* representation (1-d plot) of the data be added to the plot?
ticksize	A length-one numeric vector giving the length of the ticks making up the *rug*. Positive lengths give inwards ticks. Only used if rug is TRUE.
ncol	An integer specifying the number of columns to use when facet is "multiple". Defaults to 1 for up to 4 series, otherwise to 2.
flip	A logical scalar: should the y-axis (ticks and numbering) be flipped from side 2 (left) to 4 (right) from variable to variable?
xlab, ylab	A character vector giving the x and y axis labels.
main	A character string giving a main title for the plot.
ann	A logical scalar: should the default annotation (title and x and y axis labels) appear on the plot?
axes	A logical scalar: should axes be drawn on the plot?
frame.plot	A logical scalar: should a box be drawn around the plot?
legend	A list of additional arguments to be passed to graphics::legend(); names of the list are used as argument names. If NULL, no legend is displayed.

Value

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

Author(s)

N. Frerebeau

See Also

Other plot methods: as_graph(), barplot(), hist(), plot()

Examples

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

## Log ratio
clr <- transform_clr(coda)
plot(clr, group = NULL, flip = TRUE, border = "black", col = NA)
plot(clr, flip = TRUE)
```

plot_outliers *Plot Outliers*

Description

Plot Outliers

Usage

```
## S4 method for signature 'OutlierIndex,missing'
plot(
  x,
  ...,
  qq = FALSE,
  probs = c(0.25, 0.75),
  ncol = NULL,
  flip = FALSE,
  xlab = NULL,
  ylab = NULL,
  main = NULL,
  sub = NULL,
  ann = graphics::par("ann"),
  axes = TRUE,
  frame.plot = axes,
  panel.first = NULL,
  panel.last = NULL
)
```

Arguments

x	An OutlierIndex object.
...	Further graphical parameters.
qq	A logical scalar: should a quantile-quantile plot be produced?
probs	A length-two numeric vector representing probabilities. Corresponding quantile pairs define the line drawn (see stats::qqline()). Only used if qq is TRUE.
ncol	An integer specifying the number of columns to use when facet is "multiple". Defaults to 1 for up to 4 series, otherwise to 2.
flip	A logical scalar: should the y-axis (ticks and numbering) be flipped from side 2 (left) to 4 (right) from group to group?
xlab, ylab	A character vector giving the x and y axis labels.
main	A character string giving a main title for the plot.
sub	A character string giving a subtitle for the plot.
ann	A logical scalar: should the default annotation (title and x and y axis labels) appear on the plot?

axes	A logical scalar: should axes be drawn on the plot?
frame.plot	A logical scalar: should a box be drawn around the plot?
panel.first	An an expression to be evaluated after the plot axes are set up but before any plotting takes place. This can be useful for drawing background grids.
panel.last	An expression to be evaluated after plotting has taken place but before the axes, title and box are added.

Value

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

Author(s)

N. Frerebeau

References

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

Filzmoser, P. & Hron, K. (2008). Outlier Detection for Compositional Data Using Robust Methods. *Mathematical Geosciences*, 40(3), 233-248. doi:10.1007/s11004007914l5.

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: outliers()

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

## Detect outliers
out <- outliers(coda, groups = NULL, robust = FALSE)

plot(out) # Plot
plot(out, qq = TRUE) # Quantile-Quantile plot

## Detect outliers by group
out <- outliers(coda[, 1:15, drop = FALSE])

plot(out, ncol = 2) # Plot
plot(out, qq = TRUE, ncol = 4) # Quantile-Quantile plot
```

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)
```

---

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.

See Also

Other statistics: aggregate(), covariance(), dist, mahalanobis(), margin(), mean(), metric_var(), scale(), variation()

Examples

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

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

## Mean
mean(coda)

## Quantile
quantile(coda)

## Metric variance
metric_var(coda)

## Metric standard deviation
metric_sd(coda)
```

---

replace_NA *Missing Values Replacement*

Description

Multiplicative replacement of missing values.

Usage

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

Arguments

x	A CompositionMatrix object.
value	A numeric vector giving the replacement values.

Value

An CompositionMatrix object, where all missing values have been replaced.

Author(s)

N. Frerebeau

References

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

See Also

Other imputation methods: missing, replace_zero()

Examples

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

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

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

---

replace_zero *Zero-Replacement*

Description

Multiplicative replacement of zeros.

Usage

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

Arguments

x	A CompositionMatrix object.
value	A numeric vector giving the detection limits of each part (in $(0, 1)$).
delta	A numeric vector specifying the fraction of the detection limit to be used in replacement.

Value

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

Author(s)

N. Frerebeau

References

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

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

See Also

Other imputation methods: `missing`, `replace_NA()`

Examples

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

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

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

`samples` *Working With Samples*

Description

Retrieves or defines the sample names.

Usage

```
any_replicated(x)

is_replicated(x)

get_samples(x)

set_samples(x) <- value

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

```
## S4 method for signature 'LogRatio'
is_replicated(x)

## S4 method for signature 'OutlierIndex'
is_replicated(x)

## S4 method for signature 'CompositionMatrix'
any_replicated(x)

## S4 method for signature 'LogRatio'
any_replicated(x)

## S4 method for signature 'OutlierIndex'
any_replicated(x)

## S4 method for signature 'CompositionMatrix'
get_samples(x)

## S4 method for signature 'LogRatio'
get_samples(x)

## S4 method for signature 'OutlierIndex'
get_samples(x)

## S4 replacement method for signature 'CompositionMatrix'
set_samples(x) <- value
```

Arguments

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

Details

In some situations, measurements may have been repeated (e.g. multiple chemical analyses on the same sample). The presence of repeated measurements can be specified by giving several observations the same sample name.

See vignette("nexus").

Value

- set_samples() returns an object of the same sort as x with the new sample names assigned.
- get_samples() returns a character vector giving the sample names of x.
- any_replicated() returns a logical scalar specifying whether or not x has replicated observations.
- is_replicated() returns a logical vector specifying whether or not an observation is a replicate.

Author(s)

N. Frerebeau

See Also

Other mutators: groups, identifiers, split(), subset(), totals

---

scalar *Scalar Product*

Description

Computes the Aitchison scalar product of two compositions.

Usage

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

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

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

Arguments

x, y	A CompositionMatrix object.
...	Currently not used.

Value

A numeric vector.

Author(s)

N. Frerebeau

See Also

Other operations in the simplex: arithmetic, closure(), perturbation(), powering()

Examples

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

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

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

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

---

scale *Scaling and Centering of Compositional Data*

Description

Scaling and Centering of Compositional Data

Usage

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

Arguments

x	A CompositionMatrix object.
center	A logical scalar or a numeric vector giving the center to be substracted.
scale	A logical scalar or a length-one numeric vector giving a scaling factor for multiplication.

Value

A CompositionMatrix object.

Author(s)

N. Frerebeau

References

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

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

See Also

Other statistics: aggregate(), covariance(), dist, mahalanobis(), margin(), mean(), metric_var(), quantile(), 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*

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).

plagioclass Plagioclass (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, chemistry, hongite, petrography

---

split *Divide into Groups*

Description

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

Usage

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

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

Arguments

x	A CompositionMatrix object.
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?
...	Currently not used.

Value

A list of CompositionMatrix objects.

Author(s)

N. Frerebeau

See Also

Other mutators: groups, identifiers, samples, subset(), totals

Examples

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

```
)

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

## Split by group
split(Y, f = get_groups(Y))

## Split by sample
split(Y, f = get_samples(Y))
```

---

subset *Extract or Replace Parts of an Object*

Description

Operators acting on objects to extract or replace parts.

Usage

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

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

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

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

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

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

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

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

## S4 method for signature 'OutlierIndex,index,ANY'
x[[i]]

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

```
x[i, j, ...] <- value

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

Arguments

x	An object from which to extract element(s) or in which to replace element(s).
i, j	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.
...	Currently not used.
drop	A logical scalar: should the result be coerced to the lowest possible dimension? This only works for extracting elements, not for the replacement.
value	A possible value for the element(s) of x.

Value

A subsetted object of the same sort as x.

Author(s)

N. Frerebeau

See Also

Other mutators: groups, identifiers, samples, split(), totals

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, , drop = FALSE] # Get the first row

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

totals *Row Sums*

Description

Retrieves or defines the row sums (before closure).

Usage

```
get_totals(x)

set_totals(x) <- value

## S4 method for signature 'CompositionMatrix'
get_totals(x)

## S4 replacement method for signature 'CompositionMatrix'
set_totals(x) <- value
```

Arguments

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

Value

- set_totals() returns an object of the same sort as x with the new row sums assigned.
- get_totals() returns the row sums of x.

Author(s)

N. Frerebeau

See Also

Other mutators: groups, identifiers, samples, split(), subset()

Examples

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

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

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

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

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

`transform_alr` *Additive Log-Ratios (ALR)*

Description

Computes ALR transformation.

Usage

```
transform_alr(object, ...)

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

Arguments

`object`	A `CompositionMatrix` object.
`...`	Currently not used.
`j`	An `integer` giving the index of the rationing part (denominator).

Details

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

Value

An `ALR` object.

Author(s)

N. Frerebeau

References

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

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

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

See Also

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

Examples

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

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

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

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

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

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

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

inv_alr <- transform_inverse(alr)
all.equal(coda, inv_alr)

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

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

---

transform_clr *Centered Log-Ratios (CLR)*

Description

Computes CLR transformation.

Usage

```
transform_clr(object, ...)
```

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

Arguments

object	A CompositionMatrix object.
...	Currently not used.
weights	A logical scalar: sould a varying weight be used. If FALSE (the default), equally-weighted parts are used. Alternatively, a positive numeric vector of weights can be specified.

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.

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)

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

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

## Additive log-ratio
```

```
alr <- transform_alr(coda)

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

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

inv_alr <- transform_inverse(alr)
all.equal(coda, inv_alr)

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

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

`transform_ilr` *Isometric Log-Ratios (ILR)*

Description

Computes ILR transformations.

Usage

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

## S4 method for signature 'CompositionMatrix,missing'
transform_ilr(object)

## S4 method for signature 'CompositionMatrix,matrix'
transform_ilr(object, base)
```

Arguments

`object`	A `CompositionMatrix` object.
`base`	A `matrix` giving the base of the transformation.
`...`	Currently not used.

Details

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

Value

An ILR object.

Author(s)

N. Frerebeau

References

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

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

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

See Also

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

Examples

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

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

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

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

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

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

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

inv_alr <- transform_inverse(alr)
all.equal(coda, inv_alr)

inv_ilr <- transform_inverse(ilr)
```

```
all.equal(coda, inv_ilr)

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

`transform_inverse` *Inverse Log-Ratio Transformation*

Description

Computes inverse log-ratio transformations.

Usage

```
transform_inverse(object, origin, ...)

## S4 method for signature 'CLR,missing'
transform_inverse(object)

## S4 method for signature 'ALR,missing'
transform_inverse(object)

## S4 method for signature 'ILR,missing'
transform_inverse(object)

## S4 method for signature 'matrix,ILR'
transform_inverse(object, origin)
```

Arguments

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

Value

A `CompositionMatrix` object.

Author(s)

N. Frerebeau

References

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

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

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)

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

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

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

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

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

inv_alr <- transform_inverse(alr)
all.equal(coda, inv_alr)

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

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

transform_lr *Pairwise Log-Ratios (LR)*

Description

Computes all pairwise log-ratio transformation.

Usage

```
transform_lr(object, ...)

## S4 method for signature 'CompositionMatrix'
transform_lr(object)
```

Arguments

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

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.

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)

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

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

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

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

inv_alr <- transform_inverse(alr)
all.equal(coda, inv_alr)

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

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

`transform_plr` *Pivot Log-Ratios (PLR)*

Description

Computes PLR transformations.

Usage

```
transform_plr(object, ...)

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

Arguments

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

Value

A `PLR` object.

Author(s)

N. Frerebeau

References

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

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

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

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

See Also

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

Examples

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

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

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

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

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

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

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

inv_alr <- transform_inverse(alr)
all.equal(coda, inv_alr)

inv_ilr <- transform_inverse(ilr)
all.equal(coda, inv_ilr)
```

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

`variation` *Variation Matrix*

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()`, `covariance()`, `dist`, `mahalanobis()`, `margin()`, `mean()`, `metric_var()`, `quantile()`, `scale()`

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)
```

Index