Package ‘aIc’

May 7, 2026

Type Package

Title Testing for Compositional Pathologies in Datasets

Version 1.0

Date 2022-09-27

Author Greg Gloor

Maintainer Greg Gloor <`ggloor@uwo.ca`>

Description A set of tests for compositional pathologies. Tests for coherence of correlations with aIc.coherent() as suggested by (Erb et al. (2020) <doi:10.1016/j.acags.2020.100026>), compositional dominance of distance with aIc.dominant(), compositional perturbation invariance with aIc.perturb() as suggested by (Aitchison (1992) <doi:10.1007/BF00891269>) and singularity of the covariation matrix with aIc.singular(). Currently tests five data transformations: prop, clr, TMM, TMMwsp, and RLE from the R packages 'ALDEx2', 'edgeR' and 'DESeq2' (Fernandes et al (2014) <doi:10.1186/2049-2618-2-15>, Anders et al. (2013)<doi:10.1038/nprot.2013.099>).

License GPL (>= 3)

Encoding UTF-8

URL `https://github.com/ggloor/aIc`

BugReports `https://github.com/ggloor/aIc/issues`

RoxygenNote 7.2.1

VignetteBuilder knitr

Imports matrixcalc, zCompositions, shiny, edgeR, ALDEx2, vegan

Suggests BiocStyle, knitr, rmarkdown

Depends R (>= 3.5.0),

NeedsCompilation no

Repository CRAN

Date/Publication 2022-10-04 23:40:04 UTC

Contents

`aIc.coherent` *Calculate the subcompositional coherence of samples in a dataset for a given correction.*

Description

'aIc.coherent' compares the correlation coefficients of features in common of the full dataset and a subset of the dataset. This is expected to be false for all compositional datasets and transforms.

Usage

```
aIc.coherent(
  data,
  norm.method = "prop",
  zero.remove = 0.95,
  zero.method = "prior",
  log = FALSE,
  group = NULL,
  cor.test = "spearman"
)
```

Arguments

`data`	can be any dataframe or matrix with samples by column
`norm.method`	can be prop, clr, RLE, TMM, TMMwsp, lvha, iqlr
`zero.remove`	is a value. Filter data to remove features that are 0 across at least that proportion of samples: default 0.95
`zero.method`	can be any of NULL, prior, GBM or CZM. NULL will not impute or change 0 values, GBM and CZM are from the zCompositions R package, and prior will simply add 0.5 to all counts.

`log`	is a logical. log transform the prop, RLE or TMM outputs, default=FALSE
`group`	is a vector containing group information. Required for clr, RLE,
`cor.test`	is either the pearson or spearman method (default)

Value

Returns a list with the correlation in `cor`, a yes/no binary decision in `is.coherent`, the `x` and `y` values for a scatterplot of the correlations in the full and subcompositions, and the plot and axis labels in `main` `xlab` and `ylab`.

Author(s)

Greg Gloor

Examples

```
data(selex)
group = c(rep('N', 7), rep('S', 7))
x <- aIc.coherent(selex, group=group, norm.method='clr', zero.method='prior')
plot(x$plot[,1], x$plot[,2], main=x$main, ylab=x$ylab, xlab=x$xlab)
```

`aIc.dominant` *`aIc.dominant` calculates the subcompositional dominance of a sample in a dataset for a given correction. This compares the distances of samples of the full dataset and a subset of the dataset. This is expected to be true if the transform is behaving rationally in compositional datasets.*

Description

`aIc.dominant` calculates the subcompositional dominance of a sample in a dataset for a given correction. This compares the distances of samples of the full dataset and a subset of the dataset. This is expected to be true if the transform is behaving rationally in compositional datasets.

Usage

```
aIc.dominant(
  data,
  norm.method = "prop",
  zero.remove = 0.95,
  zero.method = "prior",
  log = FALSE,
  distance = "euclidian",
  group = NULL
)
```

Arguments

`data`	can be any dataframe or matrix with samples by column
`norm.method`	can be prop, clr, RLE, TMM, TMMwsp
`zero.remove`	is a value. Filter data to remove features that are 0 across at least that proportion of samples: default 0.95
`zero.method`	can be any of NULL, prior, GBM or CZM. NULL will not impute or change 0 values, GBM (preferred) and CZM are from the zCompositions R package, and prior will simply add 0.5 to all counts.
`log`	is a logical. log transform the RLE or TMM outputs, default=FALSE
`distance`	can be euclidian, bray, or jaccard. euclidian on log-ratio transformed data is the same as the Aitchison distance. default=euclidian
`group`	is a vector containing group information. Required for clr, RLE, TMM, lvha, and iqlr based normalizations.

Value

Returns a list with the overlap between distances in the full and subcompositon in `ol` (expect 0), a yes/no binary decision in `is.dominant` and the table of distances for the whole and subcomposition in `dist.all` and `dist.sub`, a plot showing a histogram of the resulting overlap in distances in `plot`, and the plot and axis labels in `main` `xlab` and `ylab`

Author(s)

Greg Gloor

Examples

```
data(selex)
group = c(rep('N', 7), rep('S', 7))
x <- aIc.dominant(selex, group=group, norm.method='clr', distance='euclidian', zero.method='prior')
plot(x$plot, main=x$main, ylab=x$ylab, xlab=x$xlab)
```

`aIc.perturb` *`aIc.perturb` calculates the perturbation invariance of distance for samples with a given correction. This compares the distances of samples of the full dataset and a the perturbed dataset. This is expected to be true if the transform is behaving rationally in compositional datasets.*

Description

`aIc.perturb` calculates the perturbation invariance of distance for samples with a given correction. This compares the distances of samples of the full dataset and a the perturbed dataset. This is expected to be true if the transform is behaving rationally in compositional datasets.

Usage

```
aIc.perturb(
  data,
  norm.method = "prop",
  zero.remove = 0.95,
  zero.method = "prior",
  distance = "euclidian",
  log = FALSE,
  group = NULL
)
```

Arguments

`data`	can be any dataframe or matrix with samples by column
`norm.method`	can be prop, clr, RLE, TMM, TMMwsp
`zero.remove`	is a value. Filter data to remove features that are 0 across at least that proportion of samples: default 0.95
`zero.method`	can be any of NULL, prior, GBM or CZM. NULL will not impute or change 0 values, GBM (preferred) and CZM are from the zCompositions R package, and prior will simply add 0.5 to all counts.
`distance`	can be euclidian, bray, or jaccard. euclidian on log-ratio transformed data is the same as the Aitchison distance. default=euclidian
`log`	is a logical. log transform the RLE or TMM outputs, default=FALSE
`group`	is a vector containing group information. Required for clr, RLE, TMM, lvha, and iqlr based normalizations.

Value

Returns a list with the maximum proportional perturbation in `ol` (expect 0, but values up to 1 `is.perturb`, the table of distances for the whole and perturbaton in `dist.all` and `dist.perturb`, the histogram of the perturbations in `plot`, and the plot and axis labels in `main` `xlab` and `ylab`. .

Author(s)

Greg Gloor

Examples

```
data(selex)
group = c(rep('N', 7), rep('S', 7))
x <- aIc.perturb(selex, group=group, norm.method='clr', distance='euclidian', zero.method='prior')
plot(x$plot, main=x$main, ylab=x$ylab, xlab=x$xlab)
```

`aIc.plot` — `aIc.plot` *plots the result of the distance tests.*

Description

`aIc.plot` plots the result of the distance tests.

Usage

```
aIc.plot(test.out)
```

Arguments

`test.out`	is the output from either aIc.dominant, aIc.scale, aIc.perturb

Value

returns a plot of the density of the distance test results. test result.

Author(s)

Greg Gloor

Examples

```
data(selex)
group = c(rep('N', 7), rep('S', 7))
test.out <- aIc.dominant(selex, norm.method='prop', group=group)
aIc.plot(test.out)
```

`aIc.runExample` — `aIc.runExample` *loads the associated shiny app This will load the selex example dataset with the default group sizes, the user can upload their own local dataset and adjust groups accordingly.*

Description

`aIc.runExample` loads the associated shiny app This will load the selex example dataset with the default group sizes, the user can upload their own local dataset and adjust groups accordingly.

Usage

```
aIc.runExample()
```

Value

No return value, but instead opens a shiny connection to your default web browser with the selex dataset as an example.

Author(s)

Greg Gloor

Examples

```
library(aIc)
aIc.runExample()
```

`aIc.scale` *`aIc.scale` calculates the scaling invariance of a sample in a dataset for a given correction. This compares the distances of samples of the full dataset and a scaled version of the dataset. This is expected to be true if the transform is behaving rationally in compositional datasets.*

Description

`aIc.scale` calculates the scaling invariance of a sample in a dataset for a given correction. This compares the distances of samples of the full dataset and a scaled version of the dataset. This is expected to be true if the transform is behaving rationally in compositional datasets.

Usage

```
aIc.scale(
  data,
  norm.method = "prop",
  zero.remove = 0.95,
  zero.method = "prior",
  distance = "euclidian",
  log = FALSE,
  group = NULL
)
```

Arguments

`data`	can be any dataframe or matrix with samples by column
`norm.method`	can be prop, clr, iqlr, lvha, RLE, TMM, TMMwsp
`zero.remove`	is a value. Filter data to remove features that are 0 across at least that proportion of samples: default 0.95
`zero.method`	can be any of NULL, prior, GBM or CZM. NULL will not impute or change 0 values, GBM (preferred) and CZM are from the zCompositions R package, and prior will simply add 0.5 to all counts.

`distance`	can be euclidian, bray, or jaccard. euclidian on log-ratio transformed data is the same as the Aitchison distance. default=euclidian
`log`	is a logical. log transform the RLE or TMM outputs, default=FALSE
`group`	is a vector containing group information. Required for clr, RLE, TMM, lvha, and iqlr based normalizations.

Value

Returns a list with the overlap between distances in the full and scaled composition in `ol` (expect 0), a yes/no binary decision in `is.scale` and the table of distances for the whole and scaled composition in `dist.all` and `dist.scale`, a plot showing a histogram of the resulting overlap in distances in `plot`, and the plot and axis labels in `main` `xlab` and `ylab`

Author(s)

Greg Gloor

Examples

```
data(selex)
group = c(rep('N', 7), rep('S', 7))
x <- aIc.scale(selex, group=group, norm.method='clr', zero.method='prior')
plot(x$plot, main=x$main, ylab=x$ylab, xlab=x$xlab)
```

`aIc.singular` *`aIc.singular` tests for singular data. This is expected to be true if the transform is behaving rationally in compositional datasets and also true in the case of datasets with more features than samples.*

Description

`aIc.singular` tests for singular data. This is expected to be true if the transform is behaving rationally in compositional datasets and also true in the case of datasets with more features than samples.

Usage

```
aIc.singular(
  data,
  norm.method = "prop",
  zero.remove = 0.95,
  zero.method = "prior",
  log = FALSE,
  group = NULL
)
```

Arguments

`data`	can be any dataframe or matrix with samples by column
`norm.method`	can be prop, clr, RLE, TMM, TMMwsp
`zero.remove`	is a value. Filter data to remove features that are 0 across at least that proportion of samples: default 0.95
`zero.method`	can be any of NULL, prior, GBM or CZM. NULL will not impute or change 0 values, GBM (preferred) and CZM are from the zCompositions R package, and prior will simply add 0.5 to all counts.
`log`	is a logical. log transform the RLE or TMM outputs, default=FALSE
`group`	is a vector containing group information. Required for clr, RLE, TMM, lvha, and iqlr based normalizations.

Value

Returns a list with a yes/no binary decision in `is.singular` and the covariance matrix in `cov.matrix`

Author(s)

Greg Gloor

Examples

```
data(selex)
group = c(rep('N', 7), rep('S', 7))
x <- aIc.singular(selex, group=group, norm.method='clr', zero.method='prior')
```

`meta16S` *16S rRNA tag-sequencing data*

Description

A count table of a 16S rRNA amplicon data Two groups, pupils and centenarians are represented with 198 and 161 samples per group respectively. samples are by column and OTU ids are by row.

Usage

```
data(meta16S)
```

Format

A data frame with 359 columns and 860 rows

Source

doi: 10.1128/mSphere.00327-17

`metaTscome` *meta-transcriptome data*

Description

A count table of a mixed population or metatranscriptome experiment. Two groups, H and BV are represented with 7 and 10 samples per group respectively. samples are by column and functions are by row.

Usage

```
data(metaTscome)
```

Format

A data frame with 17 columns and 3647 rows

Source

doi:10.1007/978-3-030-71175-7_17 and doi:10.1007/978-1-4939-8728-3_13

`selex` *Selection-based differential sequence variant abundance dataset*

Description

This data set gives the differential abundance of 1600 enzyme variants grown under selective (NS) and selective (S) conditions

Usage

```
data(selex)
```

Format

A data frame with 14 columns and 1600 rows

Source

DOI:10.1073/pnas.1322352111

`singleCell` *single cell transcriptome data*

Description

A count table of a single cell transcriptome data subset from the count table from doi:10.1038/s41592-019-0372-4. Two groups memory T cells, and cytotoxic T cells, 1000 cells per group. samples are by column and genes are by row.

Usage

```
data(singleCell)
```

Format

A data frame with 2000 columns and 1508 rows

Source

https://www.nature.com/articles/s41592-019-0372-4

`transcriptome` *Saccharomyces cerevisiae transcriptome*

Description

A count table of a highly replicated RNA-seq experiment with samples by column and genes by row. Two groups composed of SNF2 knockout and WT, 48 samples in each.

Usage

```
data(transcriptome)
```

Format

A data frame with 96 columns and 5892 rows

Source

DOI: 10.1261/rna.053959.115 and PRJEB5348

Index