Package 'SETA'

April 4, 2025

Title Single Cell Ecological Taxonomic Analysis

Version 0.1.1

Author Kyle Kimler <kkimler@broadinstitute.org>

Maintainer Kyle Kimler <kkimler@broadinstitute.org>

Description
This package provides functions for compositional analysis and visualization for single-cell data.

License GPL-3

Encoding UTF-8

Depends R (>= 4.0)

Imports methods, stats

Suggests testthat (>= 3.0.0), knitr, rmarkdown, MASS, Matrix, Seurat, SummarizedExperiment, SingleCellExperiment

VignetteBuilder knitr

biocViews SingleCell

RoxygenNote 7.3.2

Contents

setaALR *Additive Log-Ratio Transform*

Description

Applies an ALR transform to a matrix of counts using a specified reference column.

Usage

```
setaALR(counts, ref, pseudocount = 1)
```

Arguments

counts	A matrix of counts.
ref	The reference column, specified either by name or by index.
pseudocount	Numeric. Added to avoid log(0). Default is 1.

Details

ALR transforms the data by taking $\log(x_i/x_{ref})$ for each column i except the reference column. This is another way to map data from the simplex to a Euclidean space in compositional data analysis.

Value

A list with:

method A string noting `ALR_ref=<ref>`.

counts A matrix of dimension `nrow(counts) x (ncol(counts) - 1)`.

References

Aitchison, J. (1982). The Statistical Analysis of Compositional Data. *Journal of the Royal Statistical Society. Series B (Methodological)*, 44(2), 139-177.

Examples

```
mat <- matrix(c(1,2,4,8), nrow=2, byrow=TRUE)
colnames(mat) <- c("A", "B")
out <- setaALR(mat, ref="A", pseudocount=0)
out$counts
```

`setaCLR` *Centered Log-Ratio Transform*

Description

Applies a CLR transform to a matrix of counts. This transform is commonly used in compositional data analysis (CoDA) to project counts into a log-ratio space.

Usage

```
setaCLR(counts, pseudocount = 1)
```

Arguments

`counts`	A matrix of counts (rows = features, columns = samples).
`pseudocount`	Numeric. Added to all entries to avoid taking log(0). Default is 1.

Details

The CLR transform is defined as $\log(x/g(x))$ where $g(x)$ is the geometric mean of each row (sample) in the log scale. A pseudocount helps avoid log(0) - default is 1, as scRNA data can be sparse.

Value

A list with two elements:

method A string indicating the transform ("CLR").

counts A matrix of the same dimensions as the input after CLR transform.

References

Aitchison, J. (1982). The Statistical Analysis of Compositional Data. *Journal of the Royal Statistical Society. Series B (Methodological)*, 44(2), 139-177.

Examples

```
mat <- matrix(c(1,2,4,8), nrow=2, byrow=TRUE)
out <- setaCLR(mat, pseudocount=0)
out$counts
```

setaCounts *Extract Taxonomic Counts from Various Single Cell Objects*

Description

Given a SingleCellExperiment, Seurat, or long-form data.frame, this function produces a type-by-sample matrix of cell counts. For SingleCellExperiment or Seurat, it looks for type and sample columns in the object-level metadata. For long-form data frames, it expects columns bc, type, and sample.

Usage

```
setaCounts(obj)
```

Arguments

obj	Either a SingleCellExperiment, a Seurat object, or a data.frame with columns bc, type, and sample.

Details

- **SingleCellExperiment**: Reads colData(obj) for type and sample.
- **Seurat**: Uses obj@meta.data for type and sample.
- **data.frame**: Duplicates (by bc) are removed so each cell is counted once.

If type or sample columns are missing, an error is thrown.

Value

A matrix whose rows are cell types and whose columns are samples, with entries giving the count of unique barcodes per type-sample combination.

Examples

```
# For a data.frame:
df <- data.frame(
  bc = paste0("cell", 1:10),
  type = sample(c("Tcell", "Bcell"), 10, TRUE),
  sample = sample(c("sample1","sample2"), 10, TRUE)
)
cmat <- setaCounts(df)
cmat
```

`setaILR` *Isometric Log-Ratio Transform*

Description

Applies an ILR transform to a matrix of counts, using a Helmert basis by default. Optionally includes a Box-Cox–like step on the log scale.

Usage

```
setaILR(counts, boxcox_p = 0, taxTree = NULL, pseudocount = 1)
```

Arguments

`counts`	A matrix of counts.
`boxcox_p`	Numeric. If nonzero, applies a Box-Cox–type transform to the log-values.
`taxTree`	Currently unused. Reserved for future taxonomic-balance approaches.
`pseudocount`	Numeric. Pseudocount to avoid log(0). Default is 1.

Details

The ILR transform is a key tool in compositional data analysis. By default, it uses a Helmert contrast matrix. The parameter `boxcox_p` allows an additional transform on the log-values, as described by whuber on https://stats.stackexchange.com/questions/259208/how-to-perform-isometric-log-ratio-tra

Value

A list with:

method A string indicating ILR with a Helmert basis (potentially noting `boxcox_p`).

counts A matrix of ILR-transformed values with `ncol(counts) - 1` columns.

References

Aitchison, J. (1982). The Statistical Analysis of Compositional Data. *Journal of the Royal Statistical Society. Series B (Methodological)*, 44(2), 139-177.

Examples

```
mat <- matrix(c(1,2,4,8), nrow=2, byrow=TRUE)
out <- setaILR(mat, boxcox_p=0)
out$counts
```

`setaLatent` *Compute a Latent Space from Transformed Counts*

Description

Given an object produced by one of the `seta*` transform functions (e.g., `setaCLR`), this function applies a dimension reduction method (PCA, PCoA, or NMDS) to the `transform_obj$counts`.

Usage

```
setaLatent(transform_obj, method = c("PCA", "PCoA", "NMDS"), dims = 2)
```

Arguments

`transform_obj`	A list returned by, e.g., `setaCLR`, `setaILR`, etc., containing a `counts` matrix.
`method`	A string specifying the dimension reduction method. One of `"PCA"`, `"PCoA"`, or `"NMDS"`.
`dims`	Integer. Number of dimensions (components) to return. Default is 2.

Details

- **PCA**: Uses `stats::prcomp` on the rows of `transform_obj$counts`.
- **PCoA**: Computes a distance matrix via `stats::dist`, then applies classical multidimensional scaling (`stats::cmdscale`).
- **NMDS**: Uses `MASS::isoMDS` to compute non-metric MDS from the distance matrix.

Each method returns a data frame of coordinates in `latentSpace`, plus additional information specific to that method.

Value

A list containing:

method The chosen latent space method.

latentSpace A data frame of coordinates in the chosen latent space, with `dims` columns.

loadings For PCA, the loadings matrix. Otherwise `NA`.

varExplained Variance explained (for PCA or PCoA) or stress (for NMDS).

Examples

```
mat <- matrix(rpois(20, lambda=5), nrow=4)  # small 4x5 matrix
colnames(mat) <- paste0("C", 1:5)
clr_out <- setaCLR(mat)
latent_pca <- setaLatent(clr_out, method="PCA", dims=2)
latent_pca$latentSpace
```

setaLogCPM *log2(CPM) Transform*

Description

Computes log2-based counts-per-million (CPM) for each column. Optionally uses provided size factors.

Usage

```
setaLogCPM(counts, pseudocount = 1, size_factors = NULL, scale_factor = 1e+06)
```

Arguments

counts	A matrix of counts.
pseudocount	Numeric. Added to counts to avoid log2(0). Default is 1.
size_factors	Optional numeric vector. If NULL, uses the column sums.
scale_factor	Numeric. The scaling factor for "per million" style. Default is 1e6.

Details

A common RNA-seq transform is log2(CPM + 1). This variant allows adjusting the pseudocount, size factors, and overall scale factor.

Value

A list with:

method "logCPM".

counts A matrix of the same dimension, containing log2(CPM + pseudocount).

Examples

```
mat <- matrix(1:4, nrow=2)
out <- setaLogCPM(mat)
out$counts
```

setaPercent *Percentage Transform*

Description

Converts columns (samples) of a counts matrix to percentages of their respective column sums.

Usage

```
setaPercent(counts)
```

Arguments

counts	A matrix of counts.

Details

Simple re-scaling for compositional-like interpretation in percentages. Useful for simplified Wilcoxon rank comparisons and such

Value

A list with:

method "percent".

counts A matrix of the same dimension, with each column summing to 100.

Examples

```
mat <- matrix(c(1,2,4,8), nrow=2, byrow=TRUE)
out <- setaPercent(mat)
out$counts
```

---

setaTransform *Wrapper for Common Compositional Transforms*

Description

A convenience function that dispatches to one of the transforms: CLR, ALR, ILR, percent, or logCPM.

Usage

```
setaTransform(
  counts,
  method = c("CLR", "ALR", "ILR", "percent", "logCPM"),
  ref = NULL,
  taxTree = NULL,
  pseudocount = 1,
  size_factors = NULL
)
```

Arguments

counts	A matrix of counts.
method	Which transform to apply. One of "CLR", "ALR", "ILR", "percent", or "logCPM".
ref	Reference column (only used if method="ALR").
taxTree	Optional tree for ILR (not yet implemented).
pseudocount	Numeric, used by CLR, ALR, ILR, logCPM. Default 1.
size_factors	For logCPM scaling. If NULL, uses column sums.

Value

A list as returned by the respective transform function.

Examples

```
mat <- matrix(c(1,2,4,8), nrow=2, byrow=TRUE)
setaTransform(mat, method="CLR")
setaTransform(mat, method="percent")
```

Index