Package: lefser (via r-universe)

July 3, 2024

Type Package

Title R implementation of the LEfSE method for microbiome biomarker discovery

Description lefser is an implementation in R of the popular ``LDA Effect Size (LEfSe)'' method for microbiome biomarker discovery. It uses the Kruskal-Wallis test, Wilcoxon-Rank Sum test, and Linear Discriminant Analysis to find biomarkers of groups and sub-groups.

Version 1.15.3

Date 2024-06-22

License Artistic-2.0

Depends SummarizedExperiment, R (>= 4.0.0)

Imports coin, MASS, ggplot2, S4Vectors, stats, methods, utils, dplyr, testthat

Suggests knitr, rmarkdown, curatedMetagenomicData, BiocStyle, phyloseq, pkgdown, covr, withr

Encoding UTF-8

BugReports `https://github.com/waldronlab/lefser/issues`

URL `https://github.com/waldronlab/lefser`

VignetteBuilder knitr

biocViews Software, Sequencing, DifferentialExpression, Microbiome, StatisticalMethod, Classification

RoxygenNote 7.3.2

Roxygen list(markdown = TRUE)

Repository https://bioc.r-universe.dev

RemoteUrl https://github.com/bioc/lefser

RemoteRef HEAD

RemoteSha c4bbf228263fefe2636fecb52f350b7181bda012

Contents

`get_terminal_nodes` *Identify which elements of a string are terminal nodes*

Description

A terminal node in a taxonomy does not have any child nodes. For example, a species is a terminal node if there are no subspecies or strains that belong to that species. This function identifies which elements of a vector are terminal nodes simply by checking whether that element appears as a substring in any other element of the vector.

Usage

```
get_terminal_nodes(string)
```

Arguments

`string`	A character vector of strings to check for terminal nodes

Value

A logical vector indicating which elements of the string are terminal nodes

Examples

```
# What does it do?
data("zeller14")
rownames(zeller14)[988:989]
get_terminal_nodes(rownames(zeller14)[988:989])
# How do I use it to keep only terminal nodes for a lefser analysis?
terminal_nodes <- get_terminal_nodes(rownames(zeller14))
zeller14sub <- zeller14[terminal_nodes, ]
# Then continue with your analysis!
```

lefser *R implementation of the LEfSe method*

Description

Perform a LEfSe analysis: the function carries out differential analysis between two sample groups for multiple features and uses linear discriminant analysis to establish their effect sizes. Subclass information for each class can be incorporated into the analysis (see examples). Features with large differences between two sample groups are identified as biomarkers.

Usage

```
lefser(
  relab,
  kruskal.threshold = 0.05,
  wilcox.threshold = 0.05,
  lda.threshold = 2,
  groupCol = "GROUP",
  blockCol = NULL,
  assay = 1L,
  trim.names = FALSE,
  checkAbundances = TRUE,
  method = "none",
  ...,
  expr
)
```

Arguments

`relab` A SummarizedExperiment with relative abundances in the assay

`kruskal.threshold`
numeric(1) The p-value for the Kruskal-Wallis Rank Sum Test (default 0.05). If multiple hypothesis testing is performed, this threshold is applied to corrected p-values.

`wilcox.threshold`
numeric(1) The p-value for the Wilcoxon Rank-Sum Test when 'blockCol' is present (default 0.05). If multiple hypothesis testing is performed, this threshold is applied to corrected p-values.

`lda.threshold` numeric(1) The effect size threshold (default 2.0).

`groupCol` character(1) Column name in `colData(relab)` indicating groups, usually a factor with two levels (e.g., `c("cases", "controls")`; default "GROUP").

`blockCol` character(1) Optional column name in `colData(relab)` indicating the blocks, usually a factor with two levels (e.g., `c("adult", "senior")`; default NULL).

`assay` The i-th assay matrix in the `SummarizedExperiment` ('relab'; default 1).

trim.names
: Default is FALSE. If TRUE, this function extracts the most specific taxonomic rank of organism.

checkAbundances
: logical(1) Whether to check if the assay data in the relab input are relative abundances or counts. If counts are found, a warning will be emitted (default TRUE).

method
: Default is "none" as in the original LEfSe implementation. Character string of length one, passed on to p.adjust to set option for multiple testing. For multiple pairwise comparisons, each comparison is adjusted separately. Options are "holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr" (synonym for "BH"), and "none".

expr
: (deprecated) Use relab instead. A SummarizedExperiment with relative abundances in the assay

...
: Additional inputs to lower level functions (not used).

Details

The LEfSe method expects relative abundances in the expr input. A warning will be emitted if the column sums do not result in 1. Use the relativeAb helper function to convert the data in the SummarizedExperiment to relative abundances. The checkAbundances argument enables checking the data for presence of relative abundances and can be turned off by setting the argument to FALSE.

Value

The function returns a data.frame with two columns, which are names of features and their LDA scores.

Examples

```
data(zeller14)
zeller14 <- zeller14[, zeller14$study_condition != "adenoma"]
tn <- get_terminal_nodes(rownames(zeller14))
zeller14tn <- zeller14[tn,]
zeller14tn_ra <- relativeAb(zeller14tn)

# (1) Using classes only
res_group <- lefser(zeller14tn_ra,
                    groupCol = "study_condition")
# (2) Using classes and sub-classes
res_block <- lefser(zeller14tn_ra,
                    groupCol = "study_condition",
                    blockCol = "age_category")
```

`lefserPlot` *Plots results from* `lefser` *function*

Description

This function plots the biomarkers found by LEfSe, that are ranked according to their effect sizes and linked to their abundance in each class.

Usage

```
lefserPlot(
  df,
  colors = c("red", "forestgreen"),
  trim.names = TRUE,
  title = "",
  label.font.size = 3
)
```

Arguments

`df`	Data frame produced by `lefser`.
`colors`	A character(2). Colors corresponding to class 0 and 1. Defaults to `c("red", "forestgreen")`.
`trim.names`	Under the default (`TRUE`), this function extracts the most specific taxonomic rank of organism.
`title`	A character(1). The title of the plot.
`label.font.size`	A numeric(1). The font size of the feature labels. The default is `3`.

Value

Function returns plot of effect size scores produced by `lefser`. Positive scores represent the biomarker is more abundant in class '1'. Negative scores represent the biomarker is more abundant in class '0'.

Examples

```
example("lefser")
lefserPlot(res_group)
```

relativeAb *Utility function to calculate relative abundances*

Description

The function calculates the column totals and divides each value within the column by the respective column total.

This function calculates the relative abundance of each feature in the SummarizedExperiment object containing count data, expressed as counts per million (CPM)

Usage

```
relativeAb(se, assay = 1L)
```

Arguments

se	A SummarizedExperiment object with counts
assay	The i-th assay matrix in the `SummarizedExperiment` ('relab'; default 1).

Value

returns a new SummarizedExperiment object with counts per million calculated and added as a new assay named rel_abs.

Examples

```
se <- SummarizedExperiment(
    assays = list(
        counts = matrix(
            rep(1, 4), ncol = 1, dimnames = list(LETTERS[1:4], "SAMP")
        )
    )
)
assay(se)
assay(relativeAb(se))
```

zeller14 *Example dataset for lefser*

Description

The ZellerG_2014 dataset contains microbiome count data for CRC patients and controls. It was for curatedMetagenomicData using the script in the package directory "data-raw".

Usage

```
data("zeller14")
```

Format

A SummarizedExperiment with 1585 features, 199 samples

study_condition adenoma, control, CRC

age_category adult, senior

Source

`https://pubmed.ncbi.nlm.nih.gov/25432777/`

Index