Package ‘lefser’

October 17, 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 1.14.0

Date 2024-04-27

License Artistic-2.0

Depends SummarizedExperiment, R (>= 4.0.0)

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

Suggests knitr, rmarkdown, curatedMetagenomicData, BiocStyle, phyloseq, testthat, 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 7.3.1

Roxygen list(markdown = TRUE)

git_url https://git.bioconductor.org/packages/lefser

git_branch RELEASE_3_19

git_last_commit 0ab467f

git_last_commit_date 2024-04-30

Repository Bioconductor 3.19

Date/Publication 2024-10-16

Author Asya Khleborodova [cre, aut],
Ludwig Geistlinger [ctb],
Marcel Ramos [ctb] (<https://orcid.org/0000-0002-3242-0582>),
Samuel Gamboa-Tuz [ctb],
Levi Waldron [ctb],
Sehyun Oh [ctb]

Maintainer Asya Khleborodova <asya.bioconductor@gmail.com>

Contents

`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,
  ...,
  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).
`wilcox.threshold`	numeric(1) The p-value for the Wilcoxon Rank-Sum Test when 'blockCol' is present (default 0.05).
`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`).
`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

```
# (1) Using classes only
data(zeller14)
# exclude 'adenoma'
zeller14 <- zeller14[, zeller14$study_condition != "adenoma"]
res_group <- lefser(relativeAb(zeller14),
                    groupCol = "study_condition")
head(res_group)
```

```
# (2) Using classes and sublasses
data(zeller14)
# exclude 'adenoma'
zeller14 <- zeller14[, zeller14$study_condition != "adenoma"]
res_block <- lefser(relativeAb(zeller14),
                    groupCol = "study_condition",
                    blockCol = "age_category")
head(res_block)
```

`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