Package ‘fcoex’

April 15, 2020

Title FCBF-based Co-Expression Networks for Single Cells

Version 1.0.0

Description The fcoex package implements an easy-to use interface to co-expression analysis-based on the FCBF (Fast Correlation-Based Filter) algorithm. it was implemented specifically to deal with single-cell data. The modules found can be used to redefine cell populations, unrevel novel gene associations and predict gene function by guilt-by-association. The package structure is based on the CEMiTool package.

Depends R (>= 3.6)

Imports FCBF, parallel, progress, dplyr, ggplot2, ggrepel, igraph, grid, intergraph, stringr, clusterProfiler, data.table, grDevices, methods, network, scales, sna, utils, stats, SingleCellExperiment, pathwayPCA

Suggests testthat (>= 2.1.0), devtools, BiocManager, TENxPBMCData, scater, gridExtra, scran, Seurat, knitr

License GPL-3

Encoding UTF-8

biocViews GeneExpression, Transcriptomics, GraphAndNetwork, mRNAMicroarray, RNASeq, Network, NetworkEnrichment, Pathways, ImmunoOncology, SingleCell

LazyData true

RoxygenNote 6.1.1

VignetteBuilder knitr

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

git_branch RELEASE_3_10

git_last_commit 8798edb

git_last_commit_date 2019-10-29

Date/Publication 2020-04-14

Author Tiago Lubiana [aut, cre],
Helder Nakaya [aut, ths]

Maintainer Tiago Lubiana <tiago.lubiana.alves@usp.br>

R topics documented:

`.get_correlates` *.get_correlates*

Description

auxiliary function for find_cbf_modules

Usage

```
.get_correlates(i, su_i_j_matrix, discretized_exprs, exprs_small)
```

Arguments

`i`	A gene to be correlated
`su_i_j_matrix`	the dataframe with the correlations to be updated
`discretized_exprs`	the dataframe with discretized expression to extract a gene
`exprs_small`	the dataframe to after the filtering step

Value

the updated column of the su_i_j_matrix

`.plot_one_interaction` *Network visualization*

Description

Creates a graph based on interactions provided

Usage

```
.plot_one_interaction(adjacency_matrix, n, color, name)
```

Arguments

`adjacency_matrix`	An adajcency matrix from the `fcoex` object.
`n`	Number of genes to be shown
`color`	Color of the module to be plotted
`name`	Name of the module to be plotted
`...`	Optional parameters.

Value

A ggplot2 ('gg') object

`discretize` *Set the discretized expression attribute Uses the discretize_exprs function of the FCBF package*

Description

Set the discretized expression attribute Uses the discretize_exprs function of the FCBF package

Usage

```
discretize(fc, number_of_bins = 4, method = "varying_width",
  alpha = 1, centers = 3, min_max_cutoff = 0.25, show_pb = TRUE)

## S4 method for signature 'fcoex'
discretize(fc, number_of_bins = 4,
  method = "varying_width", alpha = 1, centers = 3,
  min_max_cutoff = 0.25, show_pb = TRUE)
```

Arguments

`fc` Object of class `fcoex`

`number_of_bins` Number of equal-width bins for discretization. Note: it is a binary discretization, with the first bin becoming one class ('low') and the other bins, another class ('high').#' Defaults to 4.

`method` Method applied to all genes for discretization. Methods available: "varying_width" (Binarization modulated by the number_of_bins param), "mean" (Split in ON/OFF by each gene mean expression), "median" (Split in ON/OFF by each gene median expression), "mean_sd"(Split in low/medium/high by each assigning "medium" to the interval between mean +- standard_deviation. Modulated by the alpha param, which enlarges (>1) or shrinks (<1) the "medium" interval.),), "kmeans"(Split in different groups by the kmeans algorithm. As many groups as specified by the centers param) and "min_max_%" (Similat to the "varying width", a binarization threshold in a "GMM" (A Gaussian Mixture Model as implemented by the package mclust, trying to fit 2:5 Gaussians). Default is "varying_width"

`alpha` Modulator for the "mean_sd" method.Enlarges (>1) or shrinks (<1) the "medium" interval. Defaults to 1.

`centers` Modulator for the "kmeans" method. Defaults to 3.

`min_max_cutoff` <- Modulator for the "min_max_%" method. Defaults to 0.25.

`show_pb` Enables a progress bar for the discretization. Defaults to TRUE.

Value

A data frame with the discretized features in the same order as previously

Examples

```
library(SingleCellExperiment)
data("mini_pbmc3k")
targets <- colData(mini_pbmc3k)$clusters
exprs <- as.data.frame(assay(mini_pbmc3k, "logcounts"))
fc <- new_fcoex(exprs, targets)
fc <- discretize(fc)
```

`fc` *Example fcoex object*

Description

Example fcoex object processed from the mini_pbmc3k dataset.

Usage

```
fc
```

Format

An object of class `fcoex`

Examples

```
data(fc)
fc
```

`fcoex-class` *An S4 class to represent the fcoex analysis.*

Description

An S4 class to represent the fcoex analysis.

Slots

`expression` Normalized gene expression table from single-cells `data.frame`.

`discretized_expression` Discretized gene expression table from single-cells `data.frame`.

`target` Original target classes for the cells (`factor`).

`selected_genes` Character `vector` containing the names of genes selected for analysis

`module_list` `list` containing genes in each module.

`adjacency` `data.frame` containing the adjacency table for the selected genes.

`coex_network_plot` list of ggplot graphs with module gene interactions.

`new_clusters` `list` containing gene interactions present in modules.

`mod_colors` character `vector` containing colors associated with each network module.

`ora` Over-representation analysis results `data.frame`.

`barplot_ora` list of ggplot graphs with over-representation analysis results per module.

`mod_idents` Identities of cells based on each co-expression module. Determined by the "recluster" method

`parameters` `list` containing analysis parameters.

`find_cbf_modules` *find_cbf_modules*

Description

find_cbf_modules uses Symmetrical Uncertainty as a correlation measure and the FCBF algorithm to

Usage

```
find_cbf_modules(fc, n_genes = NULL, FCBF_threshold = 0.1,
  verbose = TRUE, is_parallel = FALSE)

## S4 method for signature 'fcoex'
find_cbf_modules(fc, n_genes = NULL,
  FCBF_threshold = 0.1, verbose = TRUE, is_parallel = FALSE)
```

Arguments

`fc`	A fcoex object containing a discretized expression table
`n_genes`	Sets the number of genes to be selected in the first part of the algorithm. If left unchanged, it defaults to NULL and the thresh parameter is used. Caution: it overrides the thresh parameter altogether.
`FCBF_threshold`	A threshold for the minimum correlation (as determined by symettrical uncertainty) between each variable and the class used for wrapped FCBF function. Defaults to 0.1.
`verbose`	Adds verbosity. Defaults to TRUE
`is_parallel`	Uses package parallel to paralleliza calculations. Defaults to FALSE.

Details

1 - Filter the gene list by correlations to a class (Step 1)

and

2 - Determine soft thresholds for coexpression to genes predominantly correlated to a class.

Value

Returns a list with the CBF modules found or a adjacency matrix of the graph

Examples

```
library(SingleCellExperiment)
data("mini_pbmc3k")
targets <- colData(mini_pbmc3k)$clusters
exprs <- as.data.frame(assay(mini_pbmc3k, "logcounts"))
fc <- new_fcoex(exprs, targets)
fc <- discretize(fc)
fc <- find_cbf_modules(fc)
```

---

get_nets *Network visualization*

Description

Creates network visualizations based on the adjacency matrix obtained with the find_cbf_modules method

Usage

```
get_nets(fc, n = 10, min_elements = 5, ...)

## S4 method for signature 'fcoex'
get_nets(fc, n = 10, min_elements = 5, ...)
```

Arguments

fc	Object of class `fcoex`.
n	number of nodes to label
min_elements	Minimum number of elements in a module for it to be plotted. Defaults to 5.
...	Optional parameters.

Value

Object of class `fcoex` with profile plots

Examples

```
library(SingleCellExperiment)
data("mini_pbmc3k")
targets <- colData(mini_pbmc3k)$clusters
exprs <- as.data.frame(assay(mini_pbmc3k, "logcounts"))
fc <- new_fcoex(exprs, targets)
fc <- discretize(fc)
fc <- find_cbf_modules(fc)
fc <- get_nets(fc)
```

idents *Retrieves module identities from the recluster function*

Description

Retrieves module identities from the recluster function

Usage

```
idents(fc)

## S4 method for signature 'fcoex'
idents(fc)
```

Arguments

fc	Object of class `fcoex`

Value

Named object of class `list` with clusterings derived from the recluster function.

References

Guangchuang Yu, Li-Gen Wang, Yanyan Han, Qing-Yu He. clusterProfiler: an R package for comparing biological themes among gene clusters. OMICS: A Journal of Integrative Biology. 2012, 16(5):284-287.

Examples

```
data("fc")
idents(fc)
```

`mini_pbmc3k` *Processed subset of the pbmc3k dataset from PBMC genomics*

Description

A subset with 600 sampled cells and the top 1700 variable genes from the TENxPBMCData package pbmc3k dataset.

Usage

```
data(mini_pbmc3k)
```

Format

An object of class `SingleCellExperiment`

Details

Preprocessed in accordance to OSCA (August 2019, https://osca.bioconductor.org/)

scater::normalized . PCA with 50 components. snn graph on the PCA space + louvain clustering to yield 8 clusters . UMAP already ran

Examples

```
data(mini_pbmc3k)
mini_pbmc3k
```

`module_genes` *Get the module genes in a fcoex object*

Description

Get the module genes in a fcoex object

Usage

```
module_genes(fc, module = NULL)

## S4 method for signature 'fcoex'
module_genes(fc, module = NULL)
```

Arguments

`fc`	Object of class `fcoex`
`module`	A character string with the name of the module of which genes are to be returned. Defaults to `NULL`, which returns the full list of genes and modules.

Value

Object of class `data.frame` containing genes and their respective module

Examples

```
data("fc")
module_genes(fc)
```

mod_colors *Set module colors mod_colors attribute*

Description

Set module colors mod_colors attribute

Usage

```
mod_colors(fc)

## S4 method for signature 'fcoex'
mod_colors(fc)
```

Arguments

fc	Object of class fcoex

Value

A vector with color names.

Examples

```
data("fc")
mod_colors(fc)
```

mod_gene_num *Get the number of genes in modules in a fcoex object*

Description

Get the number of genes in modules in a fcoex object

Usage

```
mod_gene_num(fc, module = NULL)

## S4 method for signature 'fcoex'
mod_gene_num(fc, module = NULL)
```

Arguments

fc	Object of class fcoex
module	Default is NULL. If a character string designating a module is#' given, the number of genes in that module is returned instead.

Value

The number of genes in module(s)

Examples

```
data("fc")
mod_gene_num(fc, module = "TYROBP")
```

mod_names *Get module names in a fcoex object*

Description

Get module names in a fcoex object

Usage

```
mod_names(fc, include_NC = TRUE)

## S4 method for signature 'fcoex'
mod_names(fc, include_NC = TRUE)
```

Arguments

fc	Object of class fcoex
include_NC	Logical. Whether or not to include "Not.Correlated" module. Defaults to TRUE.

Value

Module names

Examples

```
data("fc")
mod_names(fc)
```

mod_ora *# Run module overrepresentation analysis*

Description

# Run module overrepresentation analysis

Usage

```
mod_ora(fc, gmt, verbose = FALSE)

## S4 method for signature 'fcoex'
mod_ora(fc, gmt, verbose = FALSE)
```

Arguments

`fc`	A fcoex object.
`gmt`	A gmt file with gene sets for ora analysis
`verbose`	Controls verbosity. Defaults to FALSE.

Value

A fcoex object containing over-representation analysis data

Examples

```
data("fc")
gmt_fname <- system.file("extdata", "pathways.gmt", package = "CEMiTool")
gmt_in <- pathwayPCA::read_gmt(gmt_fname)
fc <- mod_ora(fc, gmt_in)
```

`new_fcoex` *Create a fcoex object*

Description

Create a fcoex object

Usage

```
new_fcoex(expression = data.frame(), target = vector())
```

Arguments

`expression`	Normalized gene expression table from single-cells `data.frame`.
`target`	Original target classes for the cells (`factor`).

Value

Object of class `fcoex`

Examples

```
# Create new fcoex object
library(SingleCellExperiment)
data("mini_pbmc3k")
targets <- colData(mini_pbmc3k)$clusters
exprs <- as.data.frame(assay(mini_pbmc3k, "logcounts"))
fc <- new_fcoex(exprs, targets)
```

nmodules *Get the number of modules in a fcoex object*

Description

Get the number of modules in a fcoex object

Usage

```
nmodules(fc)

## S4 method for signature 'fcoex'
nmodules(fc)
```

Arguments

fc	Object of class fcoex

Value

number of modules

Examples

```
data("fc")
nmodules(fc)
```

ora_data *Retrieve over representation analysis (ORA) results*

Description

Retrieve over representation analysis (ORA) results

Usage

```
ora_data(fc)

## S4 method for signature 'fcoex'
ora_data(fc)
```

Arguments

fc	Object of class fcoex

Details

This function returns the results of the mod_ora function on the fcoex object. The ID column corresponds to pathways in the gmt file for which genes in the modules were enriched. The Count column shows the number of genes in the module that are enriched for each pathway. The GeneRatio column shows the proportion of genes in the module enriched for a given pathway out of all the genes in the module enriched for any given pathway. The BgRatio column shows the proportion of genes in a given pathway out of all the genes in the gmt file. For more details, please refer to the clusterProfiler package documentation.

Value

Object of class data.frame with ORA data

References

Guangchuang Yu, Li-Gen Wang, Yanyan Han, Qing-Yu He. clusterProfiler: an R package for comparing biological themes among gene clusters. OMICS: A Journal of Integrative Biology. 2012, 16(5):284-287.

Examples

```
data("fc")
ora_data(fc)
```

plot_interactions *Network visualization*

Description

Creates a graph based on interactions provided

Usage

```
plot_interactions(fc, n = 10, min_elements = 5, ...)

## S4 method for signature 'fcoex'
plot_interactions(fc, n = 10, min_elements = 5, ...)
```

Arguments

fc	Object of class fcoex.
n	number of nodes to label
min_elements	Minimum number of elements in a module for it to be plotted. Defaults to 5.
...	Optional parameters.

Value

Object of class fcoex with profile plots

Examples

```
library(SingleCellExperiment)
data("mini_pbmc3k")
targets <- colData(mini_pbmc3k)$clusters
exprs <- as.data.frame(assay(mini_pbmc3k, "logcounts"))
fc <- new_fcoex(exprs, targets)
fc <- discretize(fc)
fc <- find_cbf_modules(fc)
fc <- plot_interactions(fc)
```

---

plot_ora *ORA visualization*

Description

Creates a bar plot with the results of module overrepresentation analysis

Usage

```
plot_ora(fc, n = 10, pv_cut = 0.05, ...)

## S4 method for signature 'fcoex'
plot_ora(fc, n = 10, pv_cut = 0.05, ...)
```

Arguments

`fc`	Object of class `fcoex`.
`n`	number of enrichments to show
`pv_cut`	p-value significance cutoff. Default is 0.05.
`...`	parameters to plot_ora_single

Value

Object of class `fcoex` with ORA plots

Examples

```
data("fc")
gmt_fname <- system.file("extdata", "pathways.gmt", package = "CEMiTool")
gmt_in <- pathwayPCA::read_gmt(gmt_fname)
fc <- mod_ora(fc, gmt_in)
fc <- plot_ora(fc)
```

`recluster` *Recluster cells based on fcoex module composition*

Description

Recluster cells based on fcoex module composition

Usage

```
recluster(fc, hclust_method = "ward.D2", dist_method = "manhattan",
  k = 2, verbose = TRUE)

## S4 method for signature 'fcoex'
recluster(fc, hclust_method = "ward.D2",
  dist_method = "manhattan", k = 2, verbose = TRUE)
```

Arguments

`fc`	Object of class `fcoex`
`hclust_method`	method for the hclust function. Defaults to "ward.D2".
`dist_method`	method for the dist function. Defaults to "manhattan".
`k`	desired number of clustes. Defaults to 2.
`verbose`	Adds verbosity, defaults to TRUE.

Value

Object of class `data.frame` with new clusters

Examples

```
data("fc")
fc <- recluster(fc)
```

`save_plots` *Save fcoex object plots*

Description

Save plots into the directory specified by the `directory` argument. Note: If no directory is specified, it will save to tempdir(). A possible option is setting directory = "./Plots"

Usage

```
save_plots(fc, name, force = FALSE, directory = "tempdir()")

## S4 method for signature 'fcoex'
save_plots(fc, name, force = FALSE,
  directory = "tempdir()")
```

Arguments

fc	Object of class fcoex.
name	The name of the file to be saved.
force	If the directory exists, execution will not stop.
directory	Directory into which the files will be saved.

Value

A pdf file or files with the desired plot(s)

Examples

```
data(fc)
save_plots(fc, name = "Example")
```

show,fcoex-method *Print a fcoex object*

Description

Print a fcoex object

Usage

```
## S4 method for signature 'fcoex'
show(object)
```

Arguments

object	Object of class fcoex

Value

A fcoex object.

Examples

```
data("fc")
fc
```

show_net *Retrieve fcoex net plots*

Description

Retrieve fcoex net plots

Usage

```
show_net(fc)

## S4 method for signature 'fcoex'
show_net(fc)
```

Arguments

fc	Object of class fcoex.

Value

A plot corresponding to a fcoex analysis

Examples

```
data("fc")
show_net(fc)
```

show_ora *Retrieve fcoex ora plots*

Description

Retrieve fcoex ora plots

Usage

```
show_ora(fc)

## S4 method for signature 'fcoex'
show_ora(fc)
```

Arguments

fc	Object of class fcoex.

Value

A plot corresponding to a fcoex analysis

Examples

```
data("fc")
show_ora(fc)
```

Index