

# Package ‘fcoex’

October 17, 2020

**Title** FCBF-based Co-Expression Networks for Single Cells

**Version** 1.2.0

**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, unveil novel gene associations and predict gene function by guilt-by-association. The package structure is adapted from the CEMiTool package, relying on visualizations and code designed and written by CEMiTool's authors.

**Depends** R (>= 3.4)

**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, TENxPBMCDData, 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** 7.1.0

**VignetteBuilder** knitr

**git\_url** <https://git.bioconductor.org/packages/fcoex>

**git\_branch** RELEASE\_3\_11

**git\_last\_commit** ddc71e

**git\_last\_commit\_date** 2020-04-27

**Date/Publication** 2020-10-16

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

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

**R topics documented:**

<code>.get_correlates</code>	2
<code>.plot_one_interaction</code>	3
<code>discretize</code>	3
<code>fc</code>	4
<code>fcoex-class</code>	5
<code>find_cbf_modules</code>	6
<code>get_nets</code>	7
<code>idents</code>	8
<code>mini_pbmc3k</code>	8
<code>module_genes</code>	9
<code>mod_colors</code>	10
<code>mod_gene_num</code>	10
<code>mod_names</code>	11
<code>mod_ora</code>	11
<code>new_fcoex</code>	12
<code>nmodules</code>	13
<code>ora_data</code>	13
<code>plot_ora</code>	14
<code>recluster</code>	15
<code>save_plots</code>	16
<code>show,fcoex-method</code>	16
<code>show_net</code>	17
<code>show_ora</code>	18

<b>Index</b>	<b>19</b>
--------------	-----------

---

<code>.get_correlates</code>	<i>.get_correlates</i>
------------------------------	------------------------

---

**Description**

auxiliary function for `find_cbf_modules`

**Usage**

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

**Arguments**

<code>i</code>	A gene to be correlated
<code>su_i_j_matrix</code>	the dataframe with the correlations to be updated
<code>discretized_exprs</code>	the dataframe with discretized expression to extract a gene
<code>exprs_small</code>	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 This function was copied and adapted from the CEMiTool package. The visualization of networks in this function is derivative of the intellectual work of CEMiTool's authors.

### Usage

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

### Arguments

adjacency_matrix	An adjacency 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	<i>Set the discretized expression attribute Uses the discretize_exprs function of the FCBF package</i>
------------	--

---

### 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,
```

```

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 before trimming.

`adjacency_trimmed` `data.frame` containing the adjacency table for the selected genes after trimming.

`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_idsents` 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 symmetrical 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

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

## Details

This function was copied and adapted from the `CEMiTool` package. The visualization of networks in this function is derivative of the intellectual work of `CEMiTool`'s authors.

## 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	<i>Retrieves module identities from the recluster function</i>
--------	--

---

**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**

Guangchuan 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	<i>Processed subset of the pbmc3k dataset from PBMC genomics</i>
-------------	--

---

**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	<i>Get the module genes in a fcoex object</i>
--------------	---

---

## Description

This function was copied and adapted from the CEMiTool package.

## 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	<i>Set module colors mod_colors attribute</i>
------------	---

---

**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	<i>Get the number of genes in modules in a fcoex object This function was copied and adapted from the CEMiTool package.</i>
--------------	---

---

**Description**

Get the number of genes in modules in a fcoex object This function was copied and adapted from the CEMiTool package.

**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	<i>Get module names in a fcoex object</i>
-----------	---

---

**Description**

This function was copied and adapted from the CEMiTool package.

**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	<i># Run module overrepresentation analysis</i>
---------	---

---

**Description**

This function was modified from the CEMiTool package. Chunks of code were retained "as is"

**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	<i>Create a fcoex object</i>
-----------	------------------------------

---

**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	<i>Get the number of modules in a fcoex object</i>
----------	--

---

**Description**

This function was copied and adapted from the CEMiTool package.

**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	<i>Retrieve over representation analysis (ORA) results</i>
----------	--

---

**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.

This function was *ipsis litteris* adapted from the `CEMiTool` package.

## Value

Object of class `data.frame` with ORA data

## References

Guangchuan 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\_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

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

## Details

This function was copied and adapted from the `CEMiTool` package. The visualization in this function is derivative of the intellectual work of `CEMiTool`'s authors.

**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 clusters. 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	<i>Save fcoex object plots</i>
------------	--------------------------------

---

**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"

This function was modified from the CEMiTool package. Chunks of code were retained "as is"

**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	<i>Print a fcoex object</i>
--------------------	-----------------------------

---

**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	<i>Retrieve fcoex net plots</i>
----------	---------------------------------

---

**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**

This function was copied and adapted from the CEMiTool package. The visualization in this function is derivative of the intellectual work of CEMiTool's authors.

**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

- \* **datasets**
  - fc, 4
  - mini\_pbmc3k, 8
  - .get\_correlates, 2
  - .plot\_one\_interaction, 3
- discretize, 3
- discretize, fcoex-method (discretize), 3
- fc, 4
- fcoex-class, 5
- find\_cbf\_modules, 6
- find\_cbf\_modules, fcoex-method (find\_cbf\_modules), 6
- get\_nets, 7
- get\_nets, fcoex-method (get\_nets), 7
- idents, 8
- idents, fcoex-method (idents), 8
- mini\_pbmc3k, 8
- mod\_colors, 10
- mod\_colors, fcoex-method (mod\_colors), 10
- mod\_gene\_num, 10
- mod\_gene\_num, fcoex-method (mod\_gene\_num), 10
- mod\_names, 11
- mod\_names, fcoex-method (mod\_names), 11
- mod\_ora, 11
- mod\_ora, fcoex-method (mod\_ora), 11
- module\_genes, 9
- module\_genes, fcoex-method (module\_genes), 9
- new\_fcoex, 12
- nmodules, 13
- nmodules, fcoex-method (nmodules), 13
- ora\_data, 13
- ora\_data, fcoex-method (ora\_data), 13
- plot\_ora, 14
- plot\_ora, fcoex-method (plot\_ora), 14
- recluster, 15
- recluster, fcoex-method (recluster), 15
- save\_plots, 16
- save\_plots, fcoex-method (save\_plots), 16
- show, fcoex-method, 16
- show\_net, 17
- show\_net, fcoex-method (show\_net), 17
- show\_ora, 18
- show\_ora, fcoex-method (show\_ora), 18