Package ‘scCNAutils’

November 4, 2018

Title Functions to analyze copy number aberrations in single-cell data

Version 0.0.0.9000

Description Functions to analyze copy number aberrations in single-cell data. A bunch of scripts and workflows to read and analyze scRNA-seq data and look at CNA-oriented signal.

Depends R (¿= 3.4.4)

License MIT License + file LICENSE

Encoding UTF-8

LazyData true

Imports Matrix,
dplyr,
magrittr,
tidyr,
rlang,
parallel,
ggplot2,
data.table,
Rtsne,
FNN,
igraph

Suggests testthat

RoxygenNote 6.1.0

R topics documented:

`auto_cna_signal` *Automated pipeline to compute CNA signal from scRNA expression*

Description

Goes from reading raw gene counts to CNA-level signal, tSNE and community detection.

Usage

```
auto_cna_signal(data, genes_coord, prefix = "scCNAutils_out",
  nb_cores = 1, pause_after_qc = FALSE, max_mito_prop = 0.2,
  min_total_exp = 0, chrs = c(1:22, "X", "Y"), cell_cycle = NULL,
  bin_mean_exp = 3, z_wins_th = 3, smooth_wsize = 3, cc_sd_th = 3,
  nb_pcs = 10, comm_k = 100)
```

Arguments

`data` a data.frame with gene expression of the path to the folder with the 'matrix.mtx', 'genes.tsv' and 'barcodes.tsv' files.

`genes_coord` either a file name or a data.frame with coordinates and gene names.

`prefix` the prefix to use for the files created by this function (e.g. graphs).

`nb_cores` the number of processors to use.

`pause_after_qc` pause after the QC to pick custom QC thresholds.

`max_mito_prop` the maximum proportion of mitochondrial RNA.

`min_total_exp` the minimum total cell expression

`chrs` the chromosome names to keep. NULL to include all the chromosomes.

`cell_cycle` if non-null, either a file or data.frame to compute cell cycle scores. See details.

`bin_mean_exp` the desired minimum mean expression in the bin.

`z_wins_th` the threshold to winsorize Z-score. Default is 3

`smooth_wsize` the window size for smoothing. Default is 3.

`cc_sd_th` the number of SD used for the thresholds when defining cycling cells.

`nb_pcs` the number of PCs used in the community detection or tSNE.

`comm_k` the number of nearest neighbor for the KNN graph. Default 100.

Value

a data.frame with QC, community and tSNE for each cell.

Author(s)

Jean Monlong

`bin_genes` *Merge consecutive genes into expressed bins*

Description

Merge consecutive genes into expressed bins

Usage

```
bin_genes(ge_df, mean_exp = 3, nb_cores = 1)
```

Arguments

`ge_df`	the input gene expression with coordinate columns (chr, start, end) and then one column per cell.
`mean_exp`	the desired minimum mean expression in the bin.
`nb_cores`	the number of processors to use.

Value

a data.frame with bin expression.

Author(s)

Jean Monlong

`convert_to_coord` *Convert gene symbols to coordinates*

Description

If *genes_coord* is a filename, the file is expected to be a tab-delimited file with four columns: 'chr', 'start', 'end', 'symbol'. The order of the columns is not important.

Usage

```
convert_to_coord(ge_df, genes_coord, chrs = c(1:22, "X", "Y"),
  rm_dup = TRUE)
```

Arguments

`ge_df`	the data.frame with gene expression and one column 'symbol' with gene names.
`genes_coord`	either a file name or a data.frame with coordinates and gene names.
`chrs`	the chromosome names to keep. NULL to include all the chromosomes.
`rm_dup`	remove duplicated coordinates? Default is TRUE.

Details

The gene names in column 'symbol' should match the gene names in the input *ge_df*.

Value

a data.frame with columns 'chr', 'start', 'end' columns with genes coordinates (and still one column per barcode).

Author(s)

Jean Monlong

`define_cycling_cells` *Define cycling cells*

Description

Define cycling cells

Usage

```
define_cycling_cells(qc_df, sd_th = 3)
```

Arguments

`qc_df`	the output data.frame from `qc_cells` (ran with a non-null *cell_cycle* parameter)
`sd_th`	the number of SD used for the thresholds.

Value

a list with

`cells.noc`	a vector with the names of non-cycling cells
`graphs`	a list of ggplot2 graphs

Author(s)

Jean Monlong

`find_communities` *Community detection*

Description

Community detection

Usage

```
find_communities(pca_o, nb_pcs = 10, k = 100)
```

Arguments

`pca_o`	the output of `run_pca`
`nb_pcs`	the number of PCs to use. Default 10.
`k`	the number of nearest neighbor for the KNN graph. Default 100.

Value

a data.frame with two columns: 'cell' and 'community'.

Author(s)

Jean Monlong

`norm_ge` *Normalize gene expression*

Description

Normalize gene expression

Usage

```
norm_ge(ge_df, method = c("tmm", "total"), nb_cores = 1)
```

Arguments

`ge_df`	the input gene expression
`method`	the normalization method
`nb_cores`	the number of processors to use.

Value

a data.frame with the normalized expression.

Author(s)

Jean Monlong

---

plot_communities *Community graphs*

Description

Community graphs

Usage

```
plot_communities(comm_df)
```

Arguments

`comm_df`	the output data.frame from `find_communities`

Value

a list of ggplot2 graphs.

Author(s)

Jean Monlong

---

plot_qc_cells *QC graphs*

Description

QC graphs

Usage

```
plot_qc_cells(qc_df)
```

Arguments

`qc_df`	the output data.frame from `qc_cells`

Value

a list of ggplots

Author(s)

Jean Monlong

plot_tsne *tSNE graphs*

Description

tSNE graphs

Usage

```
plot_tsne(tsne_df, qc_df = NULL)
```

Arguments

`tsne_df` the output data.frame from `run_tsne` (columns: cell, tsne1, tsne2)

`qc_df` a data.frame with QC metrics (output from `qc_cells`). Default is NULL (i.e. not used)

Value

a list of ggplot objects

Author(s)

Jean Monlong

qc_cells *Compute quality control metrics for each cell*

Description

If cell_cycle is provided it should be a data.frame (or a tsv file) with two columns: 'symbol' with gene names, and 'phase' with the cell cycle phase (e.g. either 'G1.S' or 'G2.M').

Usage

```
qc_cells(ge_df, cell_cycle = NULL)
```

Arguments

`ge_df` the input gene expression with a 'symbol' column and then one column per cell.

`cell_cycle` if non-null, either a file or data.frame to compute cell cycle scores. See details.

Value

a data.frame with qc metrics per cell.

Author(s)

Jean Monlong

qc_filter *Filter cells based on QC results*

Description

Filter cells based on QC results

Usage

```
qc_filter(ge_df, qc_df, max_mito_prop = 0.2, min_total_exp = 0)
```

Arguments

`ge_df`	the input gene expression with a ’symbol’ column and then one column per cell.
`qc_df`	the output data.frame from qc_cells
`max_mito_prop`	the maximum proportion of mitochondrial RNA.
`min_total_exp`	the minimum total cell expression

Value

ge_df with only the cells that passed the filters

Author(s)

Jean Monlong

read_mtx *Read a trio of genes, barcodes and mtx files.*

Description

Read a trio of genes, barcodes and mtx files.

Usage

```
read_mtx(mtx_file = "matrix.mtx", genes_file = "genes.tsv",
  barcodes_file = "barcodes.tsv", path = ".", rm_dup = TRUE,
  genes_col = 2)
```

Arguments

mtx_file the path to the mtx file

genes_file the path to the genes file.

barcodes_file the path to the barcodes file

path the path to the folder containing the files

rm_dup remove duplicated gene names? Default is TRUE.

genes_col the column to use in genes_file. Default is 2.

Value

a data.frame with a 'symbol' column with gene names and one column per barcode.

Author(s)

Jean Monlong

---

run_pca *Run PCA*

Description

Cells in core_cells are used to build the principal components to which all cells are then projected to. Usually used to reduce the effect of cell cycle in the PCA, by using only cells that don't cycle (see `qc_cells`) as *core_cells*.

Usage

```
run_pca(z_df, core_cells = NULL, out_pcs = 100)
```

Arguments

z_df a data.frame with z-scores for each cell

core_cells if non-NULL, a vector with the names of the cells to use as core cells. See details. Default is NULL.

out_pcs the number of top PCs to report. Default is 100.

Details

The graph (*sdev.graph*) shows the standard deviation for the top 50 PCs. To show more/less PCs, add `xlim(1,N)` to the *sdev.graph*. See examples.

Value

a list with

x the PC matrix

sdev the standard deviations of the PCs

sdev.graph a ggplot graph of the sdev

Author(s)

Jean Monlong

Examples

```
## Not run:
pca.o = run_pca(z)

## Zoom in to the top 20 PCs
pca.o$sdev.graph + xlim(1,20)

## End(Not run)
```

run_tsne *Run tSNE*

Description

Run tSNE

Usage

```
run_tsne(pca_o, nb_pcs = 10, nb_it = 1000)
```

Arguments

pca_o	the output of run_pca
nb_pcs	the number of PCs to use. Default 10.
nb_it	the number of iterations. Default 1000.

Value

a data.frame with columns: cell, tsne1, tsne2

Author(s)

Jean Monlong

`smooth_movingw` *Moving-window smoothing*

Description

Moving-window smoothing

Usage

```
smooth_movingw(df, wsize = 3, nb_cores = 1, FUN = stats::median)
```

Arguments

`df`	the input data.frame with coordinate columns (chr, start, end) and then one column per cell
`wsize`	the window size. Default is 3.
`nb_cores`	the number of processors to use.
`FUN`	the function to apply to each window. Default is median.

Value

a data.frame with smoothed signal.

Author(s)

Jean Monlong

`zscore` *Compute Z-score*

Description

Compute Z-score

Usage

```
zscore(ge_df, wins_th = 3, method = c("z", "norm"), normals = NULL)
```

Arguments

`ge_df`	the input expression data.frame
`wins_th`	the threshold to winsorize Z-score. Default is 3
`method`	the normalization method. Either 'z' or norm'.
`normals`	the cells to use as normals. If NULL (default) all cells are used as normals

Value

a data.frame with Z-scores.

Author(s)

Jean Monlong

Index