Package 'cytofkit'

October 9, 2015

Type Package

Title cytofkit: an integrated analysis pipeline for mass cytometry data

Version 1.1.0

Date 2015-2-12

Author Jinmiao Chen, Hao Chen

Description An integrated mass cytometry data analysis pipeline that enables simultaneous illustration of cellular diversity and progression.

License GPL-3

Maintainer Jinmiao Chen <Chen_Jinmiao@immunol.a-star.edu.sg>

biocViews BiomedicalInformatics, GUI, CellBiology, Clustering, DimensionReduction, MassSpectrometry

Depends R (>= 2.10.0)

VignetteBuilder knitr

Suggests knitr, RUnit, BiocGenerics

Imports tcltk, stats, Rtsne, e1071, flowCore, ggplot2, gplots, vegan, reshape, Biobase

NeedsCompilation no

R topics documented:

`add_col_to_fcs` *Add new columns to the fcs expression data*

Description

Store the new dimension transformed data and cluster data into the exprs matrix in new fcs files under `analyzedFCSdir`

Usage

```
add_col_to_fcs(data, rawFCSdir, analyzedFCSdir, transformed_col = c("tsne_1",
  "tsne_2"), cluster_col = c("cluster"))
```

Arguments

`data`	The new data matrix to be added in.
`rawFCSdir`	The directory containing the original fcs files.
`analyzedFCSdir`	The directory to store the new fcs files.
`transformed_col`	the column name of the dimension transformend data in `data`.
`cluster_col`	the column name of the cluster data in `data`.

Value

new fcs files stored under `analyzedFCSdir`

`cluster_gridPlot` *Grid scatter plot of the cluster results with multiple samples*

Description

Grid dot plot visualization of the cluster results, with color indicating different clusters, and panels of different samples.

Usage

```
cluster_gridPlot(clusterData, title = "cluster", point_size = NULL)
```

Arguments

clusterData	The matrix of cluster results, with rownames of their sample name and cell id.
title	the title name of the plot.
point_size	the size of the dot.

Value

the grid scatter dot plot

Examples

```
x <- c(rnorm(100, mean = 1), rnorm(100, mean = 3), rnorm(100, mean = 9))
y <- c(rnorm(100, mean = 2), rnorm(100, mean = 8), rnorm(100, mean = 5))
c <- c(rep(1,100), rep(2,100), rep(3,100))
rnames <- paste(paste('sample_', c('A','B','C'), sep = ''), rep(1:100,each = 3), sep='_')
clusterData <- data.frame(dim1 = x, dim2 = y, cluster = c)
rownames(clusterData) <- rnames
cluster_gridPlot(clusterData)
```

---

cluster_plot *Scatter plot of the cluster results*

Description

Dot plot visualization of the cluster results, with color indicating different clusters, and shape of different samples.

Usage

```
cluster_plot(clusterData, title = "cluster", point_size = NULL)
```

Arguments

clusterData	The matrix of cluster results, with rownames of their sample name and cell id.
title	the title name of the plot.
point_size	the size of the dot.

Value

the scatter dot plot

Examples

```
x <- c(rnorm(100, mean = 1), rnorm(100, mean = 3), rnorm(100, mean = 9))
y <- c(rnorm(100, mean = 2), rnorm(100, mean = 8), rnorm(100, mean = 5))
c <- c(rep(1,100), rep(2,100), rep(3,100))
rnames <- paste(paste('sample_', c('A','B','C'), sep = ''), rep(1:100,each = 3), sep='_')
clusterData <- data.frame(dim1 = x, dim2 = y, cluster = c)
rownames(clusterData) <- rnames
cluster_plot(clusterData)
```

---

clust_mean_heatmap *Heatmap plot of cluster mean value results*

Description

Heatmap plot of cluster mean value results

Usage

```
clust_mean_heatmap(clust_mean, baseName = "Cluster_mean",
  scaleMethod = "none")
```

Arguments

clust_mean	cluster mean data from results of clust_state.
baseName	The name as a prefix in the title of the heatmap.
scaleMethod	character indicating if the values should be centered and scaled in either the row direction or the column direction, or none. The default is "none".

Value

a heatmap object from gplots

Examples

```
m1 <- c(rnorm(300, 10, 2), rnorm(400, 4, 2), rnorm(300, 7))
m2 <- c(rnorm(300, 4), rnorm(400, 16), rnorm(300, 10, 3))
m3 <- c(rnorm(300, 16), rnorm(400, 40, 3), rnorm(300, 10))
m4 <- c(rnorm(300, 7, 3), rnorm(400, 30, 2), rnorm(300, 10))
m5 <- c(rnorm(300, 27), rnorm(400, 40, 1),rnorm(300, 10))
c <- c(rep(1,300), rep(2,400), rep(3,300))
rnames <- paste(paste('sample_', c('A','B','C','D'), sep = ''),
rep(1:250,each = 4), sep='_')
exprs_cluster <- data.frame(cluster = c, m1 = m1, m2 = m2, m3 = m3, m4 = m4, m5 = m5)
row.names(exprs_cluster) <- sample(rnames, 1000)
clust_statData <- clust_state(exprs_cluster)
clust_mean_heatmap(clust_statData[[1]])
```

clust_percentage_heatmap *Heatmap plot of cluster percentage results*

Description

Heatmap plot of cluster percentage results

Usage

```
clust_percentage_heatmap(clust_cellCount, baseName = "Cluster percentage",
  scaleMethod = "none")
```

Arguments

clust_cellCount	cluster count data from results of clust_state
baseName	The name as a prefix in the title of the heatmap
scaleMethod	character indicating if the values should be centered and scaled in either the row direction or the column direction, or none. The default is "none".

Value

a heatmap object from gplots

Examples

```
m1 <- c(rnorm(300, 10, 2), rnorm(400, 4, 2), rnorm(300, 7))
m2 <- c(rnorm(300, 4), rnorm(400, 16), rnorm(300, 10, 3))
m3 <- c(rnorm(300, 16), rnorm(400, 40, 3), rnorm(300, 10))
m4 <- c(rnorm(300, 7, 3), rnorm(400, 30, 2), rnorm(300, 10))
m5 <- c(rnorm(300, 27), rnorm(400, 40, 1),rnorm(300, 10))
c <- c(rep(1,300), rep(2,400), rep(3,300))
rnames <- paste(paste('sample_', c('A','B','C','D'), sep = ''),
rep(1:250,each = 4), sep='_')
exprs_cluster <- data.frame(cluster = c, m1 = m1, m2 = m2, m3 = m3, m4 = m4, m5 = m5)
row.names(exprs_cluster) <- sample(rnames, 1000)
clust_statData <- clust_state(exprs_cluster)
clust_percentage_heatmap(clust_statData[[2]])
```

clust_state *Statistical analysis of the cluster results*

Description

Calculate the mean value of each markers in each cluster, If there are multiple samples, the percentage of cells in each cluster in each sample will be calculated

Usage

```
clust_state(exprs_cluster, stat = "mean")
```

Arguments

exprs_cluster	the expression matrix combined with the cluster results
stat	the method used for statistical analysis, like mean, median...

Value

a list contains a matrix of `clust_mean` and a matrix of `clust_cellCount`

Examples

```
m1 <- c(rnorm(300, 10, 2), rnorm(400, 4, 2), rnorm(300, 7))
m2 <- c(rnorm(300, 4), rnorm(400, 16), rnorm(300, 10, 3))
m3 <- c(rnorm(300, 16), rnorm(400, 40, 3), rnorm(300, 10))
m4 <- c(rnorm(300, 7, 3), rnorm(400, 30, 2), rnorm(300, 10))
m5 <- c(rnorm(300, 27), rnorm(400, 40, 1),rnorm(300, 10))
c <- c(rep(1,300), rep(2,400), rep(3,300))
rnames <- paste(paste('sample_', c('A','B','C','D'), sep = ''),
rep(1:250,each = 4), sep='_')
exprs_cluster <- data.frame(cluster = c, m1 = m1, m2 = m2, m3 = m3, m4 = m4, m5 = m5)
row.names(exprs_cluster) <- sample(rnames, 1000)
clust_statData <- clust_state(exprs_cluster)
```

cytofkit *cytofkit: an integrated analysis pipeline for mass cytometry data*

Description

This package is designed to facilitate the analysis workflow of mass cytometry data with automatic subset identification and population boundary detection. Both command line and a GUI are provided for runing the workflow easily.

Details

This package integrates merge methods of multiple FCS files, dimension reduction (PCA, t-SNE and ISOMAP) with density-based clustering (DensVM) for rapid subset detection. Subset-clustering scatter plot and heat map will be generated for objective comparative analysis and statistical testing. This workflow can be easily done using the main function `cytof_tsne_densvm` or a GUI for the main function `cytof_tsne_densvm_GUI`.

Pre-processing

Using function `fcs_lgcl_merge`, one or multiple FCS files were imported via the *read.FCS* function in the *flowCore* package. Then logicle transformation was applied to the expression value of selected markers of each FCS file. Auto logicle transformation and fixed logicle transformation are provided, then mutilple FCS files are merged using method `all`, `min`, `fixed` or `ceil`.

Dimensionality reduction

Using function `cytof_dimReduction`, t-Distributed Stochastic Neighbor Embedding (`tsne`) is suggested for dimensionality reduction although we also provide methods like `isomap` and `pca`.

Cluster analysis using DensVM

Density-based clustering aided by support Vector Machine (`densVM_cluster`) are used to automate subset detection from the dimension-reducted map. By using DensVM, we are able to objectively assign every cell to an appropriate cluster.

Post-processing

Cluster results are annotated by using scatter plot and heatmap. Scatter plot visualize the cell points with colour indicating their assigned clusters and point shape representing their belonging samples (`cluster_plot` and `cluster_gridPlot`). Cell events are also grouped by clusters and samples, and mean intensity values per cluster for every marker is calculated (`clust_mean_heatmap` and `clust_percentage_heatmap`). Heat map visualizing the mean expression of every marker in every cluster is generated with no scaling on the row or column direction. Hierarchical clustering was generated using Euclidean distance and complete agglomeration method. We used the heat maps to interrogate marker expression to identify each cluster's defining markers. All intermediate files and the plots can be saved using the function `cytof_write_results`.

References

http://signbioinfo.github.io/cytofkit/

See Also

`cytof_tsne_densvm`, `cytof_tsne_densvm_GUI`

Examples

```
## Run on GUI
#cytof_tsne_densvm_GUI()  # remove the hash symbol to launch the GUI

## Run on command
dir <- system.file('extdata',package='cytofkit')
file <- list.files(dir, pattern='.fcs$', full=TRUE)
parameters <- list.files(dir, pattern='.txt$', full=TRUE)
## remove the hash symbol to run the following command
```

```
#cytof_tsne_densvm(fcsFile = file, paraFile = parameters, rawFCSdir = dir, baseName = 'test')

## Checking the vignettes for more details
if(interactive()) browseVignettes(package = 'cytofkit')
```

`cytof_dimReduction` *Dimension reduction of CyTof expression data*

Description

Apply dimension reduction on the CyTof expression data, with method `isomap`, `pca`, or `tsne`.

Usage

```
cytof_dimReduction(data, method = "tsne", distMethod = "euclidean",
  isomap_k = 5, isomap_ndim = NULL, isomapFragmentOK = TRUE)
```

Arguments

`data`	An expression data matrix
`method`	Method chosed for dimensition reduction, must be one of `isomap`, `pca` or `tsne`
`distMethod`	Method for distance calcualtion
`isomap_k`	Number of shortest dissimilarities retained for a point, parameter for `isomap` method
`isomap_ndim`	Number of axes in metric scaling, parameter for `isomap` method
`isomapFragmentOK`	What to do if dissimilarity matrix is fragmented, parameter for `isomap` method

Value

a matrix of the dimension reducted data, with colnames and rownames(if have, same as the input)

Author(s)

Chen Jinmiao

Examples

```
data(iris)
in_data <- iris[, 1:4]
out_data <- cytof_dimReduction(in_data)
```

cytof_tsne_densvm *CyTOF data analysis for subpopulation detection*

Description

cytof_tsne_densvm provides a workflow for one or multiple CyToF data analysis, including data preprocess with merging methods of multiple fcs file, logicle transformation, dimension reduction with PCA, isomap or tsne(default), and a kernal-based local maxima clustering combined with SVM for subpopulation detection. The intermediate results can be saved into seperate files and the cluster results can be visualized in heatmaps and scatter plots.

Usage

```
cytof_tsne_densvm(rawFCSdir = getwd(), fcsFile = NULL, resDir = getwd(),
  baseName = "cytofkit_analysis", para = NULL,
  paraFile = "./parameter.txt", comp = FALSE, verbose = FALSE,
  lgclMethod = "fixed", scaleTo = NULL, q = 0.05, mergeMethod = "ceil",
  fixedNum = 10000, ifTransform = TRUE, transformMethod = "tsne",
  ifCluster = TRUE, visualizationMethods = "tsne", writeResults = TRUE,
  ...)
```

Arguments

rawFCSdir	the directory that contains fcs files to be analysed.
fcsFile	a vector containing names of fcs files to be analyzed. One or multiple fcs files are allowed.
resDir	the directory where result files will be generated.
baseName	a prefix that will be added to the names of result files.
para	the vector of selected makers. This can be provided in the paraFile.
paraFile	a text file that specifies the list of makers to be used for analysis.
comp	Boolean tells if do compensation. This will be applied to flow cytometry data.
verbose	Boolean.
lgclMethod	Logicle transformation method, either auto, sign_auto or fixed.
scaleTo	scale the expression to same scale, default is NULL, should be a vector of two numbers if scale
q	quantile of negative values removed for auto w estimation, default is 0.05
mergeMethod	when multiple fcs files are selected, cells can be combined using one of the four different methods including ceil, all, min, fixed. The default option is ceil, up to a fixed number (specified by fixedNum) of cells are sampled without replacement from each fcs file and combined for analysis. all: all cells from each fcs file are combined for analysis. min: The minimum number of cells among all the selected fcs files are sampled from each fcs file and combined for analysis. fixed: a fixed num (specified by fixedNum) of cells are sampled (with replacement when the total number of cell is less than fixedNum) from each fcs file and combined for analysis.

fixedNum	up to fixedNum of cells from each fcs file are used for analysis.
ifTransform	a boolean to decide if dimensionality reduction will be performed. Default is TRUE.
transformMethod	the method used for dimensionality reduction, including tsne, pca and isomap.
ifCluster	a boolean to determine if cluster will be conducted.
visualizationMethods	the method(s) used for visualize the cluster data, multiple selection are accepted, including tsne, pca and isomap
writeResults	if save the results, and the post-processing results including scatter plot, heatmap, and statistical results.
...	more arguments contral the logicle transformation

Value

a list containing lgclMergedExprs, transData and clustersRes. If choose 'writeResults = TRUE', results will be saved into files under resDir

Author(s)

Chen Jinmiao

References

http://signbioinfo.github.io/cytofkit/

See Also

cytofkit, cytof_tsne_densvm_GUI

Examples

```
dir <- system.file('extdata',package='cytofkit')
file <- list.files(dir, pattern='.fcs$', full=TRUE)
parameters <- list.files(dir, pattern='.txt$', full=TRUE)
## remove the hash symbol to run the following command
#cytof_tsne_densvm(fcsFile = file, paraFile = parameters, rawFCSdir = dir, baseName = 'test')
```

---

cytof_tsne_densvm_GUI *The user friendly GUI for function* cytof_tsne_densvm

---

Description

This GUI provides an easy way for CyToF data analysis using cytofkit package. All parameters for running 'cytof_tsne_densvm' were integrated in this GUI, each parameter has help button in the GUI to help user get details of the information of each parameter, and launch the cytof_tsne_densvm analysis after submitting.

Usage

```
cytof_tsne_densvm_GUI()
```

Value

the GUI for the main function `cytof_tsne_densvm`

Author(s)

Chen Hao

References

http://signbioinfo.github.io/cytofkit/

See Also

`cytof_tsne_densvm`, `cytofkit`

Examples

```
#cytof_tsne_densvm_GUI()  # remove the comment hash to run
```

---

cytof_write_results *Save the cytofkit analysis results*

Description

Scatter dot plot and heatmap of the cluster results, and all intermediate files will be generated and saved in the `resDir`

Usage

```
cytof_write_results(analysis_results, vizMethods,
  baseName = "cytofkit_analysis", rawFCSdir = getwd(), resDir = getwd())
```

Arguments

`analysis_results`	result data from output of `densVM_cluster`
`vizMethods`	visualization methods for clustering results, including `tsne`, `pca` and `isomap`.
`baseName`	a prefix that will be added to the names of result files.
`rawFCSdir`	the directory that contains fcs files to be analysed.
`resDir`	the directory where result files will be generated.

Value

save all results in the resDir

See Also

cytof_tsne_densvm, cytofkit

Examples

```
dir <- system.file('extdata',package='cytofkit')
f <- list.files(dir, pattern='.fcs$', full=TRUE)
p <- list.files(dir, pattern='.txt$', full=TRUE)
#tr <- cytof_tsne_densvm(fcsFile=f,paraFile=p,baseName='t',writeResults=FALSE)
#cytof_write_results(tr,baseName = 'test',rawFCSdir=dir)
```

---

densVM_cluster *Density-based local maxima cluster with SVM*

Description

Density-based local maxima peak finding, subpopulation assigning with the power of SVM

Usage

```
densVM_cluster(ydata, xdata)
```

Arguments

ydata	a matrix of the dimension reduced(transformed) data
xdata	a matrix of the expression data

Value

a list contains a matrix peakdata of the peak numbers with different kernel bandwidth, and a matrix clusters of the cluster results

Examples

```
d<-system.file('extdata',package='cytofkit')
fcsFile <- list.files(d,pattern='.fcs$',full=TRUE)
xdata <- fcs_lgcl_merge(fcsFile, mergeMethod = 'fixed', fixedNum = 100)
ydata <- cytof_dimReduction(xdata)
clusters <- densVM_cluster(ydata, xdata)
```

`fcs_lgcl` *Transformation of the FCS data*

Description

Read the FCS expresssion data and apply the transformation

Usage

```
fcs_lgcl(fcsFile, comp = FALSE, verbose = FALSE, markers = NULL,
  lgclMethod = "fixed", scaleTo = NULL, w = 0.1, t = 4000, m = 4.5,
  a = 0, q = 0.05)
```

Arguments

`fcsFile`	The name of the FCS file
`comp`	Boolean tells if do compensation
`verbose`	Boolean
`markers`	Selected markers for analysis, either from names or from description
`lgclMethod`	Logicle transformation method, `auto`, `sign_auto` or `fixed`
`scaleTo`	scale the expression to same scale, default is NULL, should be a vector of two numbers if scale
`w`	Linearization width in asymptotic decades
`t`	Top of the scale data value
`m`	Full width of the transformed display in asymptotic decades
`a`	Additional negative range to be included in the display in asymptotic decades
`q`	quantile of negative values removed for auto w estimation, default is 0.05

Value

The logicle transformend expression data matrix of selected markers

Examples

```
d<-system.file('extdata',package='cytofkit')
fcsFile <- list.files(d,pattern='.fcs$',full=TRUE)
transformed <- fcs_lgcl(fcsFile)
```

`fcs_lgcl_merge` *merge the transformed expression data of FCS file(s) of selected markers*

Description

Apply transformation of selected markers of each FCS file, arcsin, auto logicle transformation and fixed logicle transformation are provided, then mutilple FCS files are merged using method `all`, `min`, `fixed` or `ceil`

Usage

```
fcs_lgcl_merge(fcsFiles, comp = FALSE, verbose = FALSE, markers = NULL,
  lgclMethod = "fixed", scaleTo = NULL, w = 0.1, t = 4000, m = 4.5,
  a = 0, q = 0.05, mergeMethod = "ceil", fixedNum = 10000)
```

Arguments

`fcsFiles`	the input fcsFiles (usually more than 1 file)
`comp`	Boolean tells if do compensation
`verbose`	Boolean
`markers`	Selected markers for analysis, either from names or from description
`lgclMethod`	Logicle transformation method, `auto`, `sign_auto` or `fixed`
`scaleTo`	scale the expression to same scale, default is NULL, should be a vector of two numbers if scale
`w`	Linearization width in asymptotic decades
`t`	Top of the scale data value
`m`	Full width of the transformed display in asymptotic decades
`a`	Additional negative range to be included in the display in asymptotic decades
`q`	quantile of negative values removed for auto w estimation, default is 0.05
`mergeMethod`	merge method for mutiple FCS expression data, default is all
`fixedNum`	the fixed number of cells for merging multiple FCSs

Value

Merged FCS expression data matrix of selected markers with logicle transformation

Examples

```
d<-system.file('extdata',package='cytofkit')
fcsFile <- list.files(d,pattern='.fcs$',full=TRUE)
merged <- fcs_lgcl_merge(fcsFile)
```

peaksGamma_plot *Plot varaition of peak nums with increasing gamma*

Description

Plot varaition of peak nums with increasing gamma

Usage

```
peaksGamma_plot(peakdata)
```

Arguments

peakdata	a matrix of peakdata returned from densVM_cluster

Value

a line graph of peak nums vs. increasing gamma

Examples

```
x <- seq(0, 1, length.out = 20)
y <- c(20:6, 6, 6, 5:3)
peakdata <- data.frame(sig_range = x, numpeaks = y)
peaksGamma_plot(peakdata)
```

Index