

Package ‘cytofast’

March 30, 2021

Title cytofast - A quick visualization and analysis tool for CyTOF data

Version 1.6.0

Maintainer K.A. Stam <k.a.stam@hotmail.com>

Description

Multi-parametric flow and mass cytometry allows exceptional high-resolution exploration of the cellular composition of the immune system. Together with tools like FlowSOM and Cytosplore it is possible to identify novel cell types. By introducing cytofast we hope to offer a workflow for visualization and quantification of cell clusters for an efficient discovery of cell populations associated with diseases or other clinical outcomes.

Depends R (>= 3.6.0)

Imports flowCore, ggplot2, ggridges, RColorBrewer, reshape2, stats, grDevices, Rdpack, methods, grid, FlowSOM

RdMacros Rdpack

Suggests BiocStyle, knitr, rmarkdown

License GPL-3

Encoding UTF-8

LazyData true

biocViews FlowCytometry, Visualization, Clustering

RoxygenNote 6.1.1

Roxygen list(markdown = TRUE)

VignetteBuilder knitr

Collate 'cellCounts.R' 'cfList-class.R' 'cfList-accessors.R'
'cfList-methods.R' 'cytoBoxplots.R' 'cytoHeatmaps.R'
'cytotest.R' 'drawTree.R' 'msiPlot.R' 'readCytosploreFCS.R'
'spitzer-data.R'

git_url <https://git.bioconductor.org/packages/cytofast>

git_branch RELEASE_3_12

git_last_commit c9be79d

git_last_commit_date 2020-10-27

Date/Publication 2021-03-29

Author K.A. Stam [aut, cre],
G. Beyrend [aut]

R topics documented:

cellCounts	2
cfList-class	3
cytoBoxplots	4
cytoHeatmaps	5
cytottest	6
drawTree	7
msiPlot	7
readCytosplereFCS	8
spitzer	9
[,cfList,ANY-method	10

Index	11
--------------	-----------

cellCounts	<i>Extract cell counts from cfList</i>
------------	--

Description

A function to add the frequency (or abundance) of cell clusters per sample to a cfList.

Usage

```
cellCounts(cfList, frequency = FALSE, scale = FALSE)
```

Arguments

cfList	a cfList object. It should contain at least data in the 'expr' slot.
frequency	one of <ul style="list-style-type: none"> • a logical value. if FALSE, the abundance of the cell counts are used. If TRUE, the frequency of the total amount of given cells is used. • a numeric vector of same length as amount of samples
scale	a logical value. Do the cell frequencies need to be centered and scaled? The default scale function is called.

Value

Returns a cfList with a 'counts' slot.

Examples

```
# Read Data
dirFCS <- system.file("extdata", package="cytofast")
cfData <- readCytosplereFCS(dir = dirFCS, colNames = "description")

# Add cell counts to cfList
cfData <- cellCounts(cfData)
```

cfList-class	<i>Class cfList</i>
--------------	---------------------

Description

This is an S4 class that contains all the data and results for the cytofast workflow. Most functions in this package will both use and return a `cfList`.

Details

This class is used throughout the cytofast workflow, most functions in this package will both use and return a `cfList`. FCS files created by Cytosplore can be easily read in with `readCytosploreFCS` and will be returned as a `cfList`. It is also possible to manually create instances of this class if another clustering method is preferred. Below an example is shown for `FlowSOM`.

Slots

`samples` data frame containing all meta information on each sample
`expr` data frame containing the marker expression
`counts` data frame containing (standardized) cell counts per cluster and per sample
`results` list containing any results

Methods

`samples` a method for obtaining the samples data
`expr` a method for obtaining the marker expression
`counts` a method for obtaining the cell counts
`results` a method for obtaining any results from performed tests

Examples

```
### manually create instance of cfList class
library(FlowSOM)

## Cluster with FlowSOM
dirFCS <- system.file("extdata", package="cytofast")
fSOM <- FlowSOM(input = dirFCS,
               transform = FALSE,
               scale = FALSE,
               colsToUse = c(9:11, 15:52),
               nClus = 10, # Note that this is an ambiguous choice
               seed = 123)

## expr slot
# retrieve clusters
clusterID <- as.factor(fSOM$FlowSOM$map$mapping[,1])
levels(clusterID) <- fSOM$metaclustering

# retrieve samples (As example, we assume each FCS file is its own sample)
sampleID <- lapply(fSOM$FlowSOM$metaData, function(x){rep(x[1], each = length(x[1]:x[2]))})
attr(sampleID, 'names') <- NULL
```

```

sampleID <- as.factor(unlist(sampleID))
levels(sampleID) <- paste("ID", 1:10, sep="_")

exprD <- data.frame(clusterID,
                    sampleID,
                    fSOM$FlowSOM$data[, c(9:11, 15:52)])

## samples slot
samplesD <- data.frame(sampleID = levels(sampleID),
                       group = rep(c("group1", "group1"), each=5))

## create cfList
cfList(samples = samplesD,
        expr = exprD)

```

cytoBoxplots

Draw boxplots for cfList

Description

Draw boxplots for all given clusters. Values are based on the counts slot in the cfList.

Usage

```
cytoBoxplots(cfList, group, stat)
```

Arguments

cfList	a cfList object. It should contain at least data in the 'counts' slot.
group	one of: <ul style="list-style-type: none"> a character referring to a column name in the samples slot of the cfList. a factor indicating the grouping (x-axis) for the boxplots.
stat	ignore, will be implemented soon.

Value

None

Examples

```

# Read Data
dirFCS <- system.file("extdata", package="cytofast")
cfData <- readCytosploreFCS(dir = dirFCS, colNames = "description")

# Add cell counts to cfList and add meta data
cfData <- cellCounts(cfData, frequency = TRUE, scale = TRUE)
meta <- spitzer[match(row.names(cfData@samples), spitzer$CSPLR_ST),]
cfData@samples <- cbind(cfData@samples, meta)

# Remove unnecessary markers
cfData@expr <- cfData@expr[,-c(3:10, 13:16, 55:59, 61:63)]

```

```
# Draw boxplots
cytoBoxplots(cfData, group="group")
```

cytoHeatmaps

Draw heatmaps for cfList

Description

Function to draw two heatmaps. They visualize the median intensity of the markers for the created clusters. The ordering of the clusters is based on the default hierarchical cluster analysis [hclust](#). Note that hclust takes the data after the median intensity is calculated per cluster, thus placing the most similar clusters next to each other.

Usage

```
cytoHeatmaps(cfList, group, legend = FALSE)
```

Arguments

cfList	a cfList object.
group	one of: <ul style="list-style-type: none"> a character vector referring to a column name in the samples slot of the cfList. a factor indicating the grouping (x-axis) for the boxplots.
legend	logical, whether a legend should be added

Value

None

Examples

```
# Read Data
dirFCS <- system.file("extdata", package="cytofast")
cfData <- readCytosplereFCS(dir = dirFCS, colNames = "description")

# Add cell counts to cfList and add meta data
cfData <- cellCounts(cfData, frequency = TRUE, scale = TRUE)
meta <- spitzer[match(row.names(cfData@samples), spitzer[, "CSPLR_ST"]),]
cfData@samples <- cbind(cfData@samples, meta)

# Remove unnecessary markers
cfData@expr <- cfData@expr[,-c(3:10, 13:16, 55:59, 61:63)]

# Draw heatmaps
cytoHeatmaps(cfData, group = "group", legend = TRUE)
```

cytottest	<i>perform t-test on cfList</i>
-----------	---------------------------------

Description

Performs a separate t-test on each cluster within a cfList. The output is added and can also be used by other functions

Usage

```
cytottest(cfList, group, adjustMethod, ...)
```

Arguments

cfList	a cfList object. It should contain at least data in the 'counts' slot.
group	one of: <ul style="list-style-type: none"> • a character vector referring to a column name in the samples slot of the cfList. • a factor indicating the grouping for the t.test.
adjustMethod	character, correction method used in <code>p.adjust</code> , choose from <code>c("holm", "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "p.adjust.method")</code> . If omitted, no correction takes place.
...	further arguments passed on to <code>t.test</code> .

Value

Returns a cfList with a 'results' slot.

Examples

```
# Read Data
dirFCS <- system.file("extdata", package="cytofast")
cfData <- readCytosplreFCS(dir = dirFCS, colNames = "description")

# relabeling of clusterID
levels(cfData@expr[, "clusterID"]) <- gsub("[^0-9]", "", levels(cfData@expr[, "clusterID"]))

# Add cell counts to cfList and add meta data
cfData <- cellCounts(cfData, frequency = TRUE, scale = TRUE)
meta <- spitzer[match(row.names(cfData@samples), spitzer$CSPLR_ST),]
cfData@samples <- cbind(cfData@samples, meta)

# Run t-test
cfData@samples$effect <- gsub("_D\\d", "", spitzer$group)
cfData <- cytottest(cfData, group = "effect", adjustMethod = "bonferroni")
cfData@results
```

drawTree	<i>Draw a dendrogram with ggplot</i>
----------	--------------------------------------

Description

Function to draw only a dendrogram with ggplot, the tree can be easily aligned to other graphs (e.g. heatmaps).

Usage

```
drawTree(hclust)
```

Arguments

hclust an object of class `hclust`.

Value

None

Examples

```
hc <- hclust(dist(data.frame(x1 = rnorm(10), x2 = rnorm(10), x3 = rnorm(10))))
drawTree(hc)
```

msiPlot	<i>Draw a density (median signal intensity) plot</i>
---------	--

Description

Function to plot the density (median signal intensity) for given markers.

Usage

```
msiPlot(cfList, markers, byGroup = NULL, byCluster = NULL, ...)
```

Arguments

cfList a cfList object. It should contain at least data in the 'expr' and 'samples' slots.

markers character vector with column names of the markers to be plotted. A numeric vector is also accepted, note that 1 starts after removing columns 'clusterID' and 'sampleID'.

byGroup a character, referring to a column name in the samples slot of the cfList. This will be used as the grouping for the y-axis.

byCluster character vector specifying which cluster to be plotted. This will be used as the grouping for the y-axis.

... Additional arguments passed on to `geom_density_ridges`.

Value

None

Examples

```
# Read Data
dirFCS <- system.file("extdata", package="cytofast")
cfData <- readCytosploreFCS(dir = dirFCS, colNames = "description")

# relabeling of clusterID
levels(cfData@expr[, "clusterID"]) <- gsub("[^0-9]", "", levels(cfData@expr[, "clusterID"]))

# Add cell counts to cfList and add meta data
cfData <- cellCounts(cfData, frequency = TRUE, scale = TRUE)
meta <- spitzer[match(row.names(cfData@samples), spitzer$CSPLR_ST),]
cfData@samples <- cbind(cfData@samples, meta)

# Remove unnecessary markers
cfData@expr <- cfData@expr[,-c(3:10, 13:16, 55:59, 61:63)]

# Draw median signal intensity plot, by group
msiPlot(cfData, markers = c("MHC.II", "CD45", "CD4"), byGroup = 'group')

# Or by cluster
msiPlot(cfData, markers = c("MHC.II", "CD45", "CD4"), byCluster = c("1", "6", "10"))
```

readCytosploreFCS *Read .fcs files created by cytosplore*

Description

This function reads and combines .FCS files created specifically by Cytosplore.

Usage

```
readCytosploreFCS(dir = NULL, colNames = c("names", "description"))
```

Arguments

dir	directory containing the .FCS files created by Cytosplore
colNames	character string that specifies which label should be used as the column names. This could be the name of the markers 'names' or the description for the markers 'description'.

Value

The function returns an object of class `data.frame`. It includes both `clusterID` and `sampleID` as variables.

Note

This function is a wrapper around [read.FCS](#). For more flexibility see their help page.

Examples

```
dirFCS <- system.file("extdata", package="cytofast")
cfData <- readCytosploreFCS(dir = dirFCS, colNames = "description")
```

spitzer

Spitzer meta file

Description

A dataset containing information on the samples from the Spitzer study

Usage

```
spitzer
```

Format

A data frame with 26 observations and 3 variables:

CSPLR_ST sample tag, corresponds with .fcs files created by Cytosplore

group grouping variable, effected vs. ineffectuated, and day 3 vs. day 8

sample_name original sampleID

Source

<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5312823/>

References

Spitzer MH, Carmi Y, Reticker-Flynn NE, Kwek SS, Madhireddy D, Martins MM, Gherardini PF, Prestwood TR, Chabon J, Bendall SC, Fong L, Nolan GP, Engleman EG (2017). "Systemic Immunity is Required for Effective Cancer Immunotherapy HHS Public Access." *Cell*, **26**(168(3)), 487–502. doi: [10.1016/j.cell.2016.12.022](https://doi.org/10.1016/j.cell.2016.12.022), <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5312823/pdf/nihms842747.pdf>.

[,cfList,ANY-method] *Extract parts of cfList*

Description

Extract parts of cfList

Usage

```
## S4 method for signature 'cfList,ANY'  
x[i, j]
```

Arguments

x	a cfList object from which to extract from
i	index specifying which samples to extract. The index is either a numeric or character vector.
j	index specifying which clusters to extract. The index is either a numeric or character vector.

Value

Returns the specified clusters or samples from a cfList.

Index

- * **FCS**,
 - readCytosploreFCS, 8
 - * **FCS**
 - cytoHeatmaps, 5
 - * **boxplot**,
 - cytoBoxplots, 4
 - * **cells**
 - cellCounts, 2
 - * **counts**,
 - cellCounts, 2
 - * **cytosplore**
 - readCytosploreFCS, 8
 - * **data**,
 - readCytosploreFCS, 8
 - * **datasets**
 - spitzer, 9
 - * **dendrogram**
 - drawTree, 7
 - * **density**
 - msiPlot, 7
 - * **frequency**,
 - cellCounts, 2
 - * **heatmap**,
 - cytoHeatmaps, 5
 - * **heatmap**
 - drawTree, 7
 - * **intensity**
 - msiPlot, 7
 - * **markers**,
 - cytoHeatmaps, 5
 - * **median**
 - msiPlot, 7
 - * **read**,
 - readCytosploreFCS, 8
 - * **signal**
 - msiPlot, 7
 - * **tree**
 - drawTree, 7
- [, cfList, ANY-method, 10
- cellCounts, 2
- cfList (cfList-class), 3
- cfList-class, 3
- counts (cfList-class), 3
- counts, cfList-method (cfList-class), 3
- cytoBoxplots, 4
- cytoHeatmaps, 5
- cytottest, 6
- data.frame, 8
- drawTree, 7
- expr (cfList-class), 3
- expr, cfList-method (cfList-class), 3
- FlowSOM, 3
- geom_density_ridges, 7
- hclust, 5, 7
- msiPlot, 7
- p.adjust, 6
- read.FCS, 9
- readCytosploreFCS, 3, 8
- results (cfList-class), 3
- results, cfList-method (cfList-class), 3
- samples (cfList-class), 3
- samples, cfList-method (cfList-class), 3
- scale, 2
- show, cfList-method (cfList-class), 3
- spitzer, 9
- t.test, 6