Package 'CytobankAPIstats'

June 27, 2018

Type Package

Title Computes Signaling and Population Stats for Cytometry Data on Cytobank using 'CytobankAPI'

Version 2.0

Date 2018-06-27

Author Allison Throm

Maintainer Allison Throm <throm@email.wustl.edu>

Description Tools to process cytometry data from Cytobank into easily usable form for analysis of populations, markers, and signaling using the 'CytobankAPI' package. Learn more about Cytobank at <https://www.cytobank.org>. For more information about types of cytometry data that can be analyzed, please see: Bendall, S. C., Simonds, E. F., Qiu, P., Amir, E. D., Krutzik, P. O., Finck, R.,... Nolan, G. P. (2011) <doi:10.1126/science.1198704> and Adan, izada, G., Kiraz, Y., Baran, Y., Nalbant, A. (2017). <doi:10.3109/07388551.2015.1128876>.

License Artistic-2.0

Imports CytobankAPI, shiny, xlsx, shinyFiles, pheatmap

Suggests httr, methods, curl, stats, jsonlite

RoxygenNote 6.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2018-06-27 14:58:30 UTC

R topics documented:

`analyzedata` *Returns a matrix of event counts or raw medians, as specified by inputs. Columns correspond to fcs files and rows to markers in cell types*

Description

Returns a matrix of event counts or raw medians, as specified by inputs. Columns correspond to fcs files and rows to markers in cell types

Usage

```
analyzedata(cyto_session, markersofinterest, popsofinterest, exptID, type)
```

Arguments

`cyto_session`	- API authentication token for session
`markersofinterest`	- Names of channel parameters in Cytobank as list of strings
`popsofinterest`	- Names of gates of interest in Cytobank as list of strings
`exptID`	- Integer representing an experiment ID on Cytobank account
`type`	- boolean with TRUE to analyze events, FALSE to analyze marker intensity statistics

Value

Returns a data matrix of event counts or raw signal medians, as specified by variable type

Examples

```
library(CytobankAPI)
cyto_session <- authenticate(site="premium", username="myusername", password="mypassword")
markersofinterest<-c("CD3","CD56")
popsofinterest<-c("CD4 T cells","NK cells")
exptID=4
type=TRUE
analyzedata(cyto_session,markersofinterest,popsofinterest,exptID,type)
```

asinnorm *Computes the arcsinh ratio of a matrix in relation to the specified column*

Description

Computes the arcsinh ratio of a matrix in relation to the specified column

Usage

```
asinnorm(mat, col, cofactor)
```

Arguments

mat	- The result of a call to the parsestats function
col	- The index of column to compute ratios against
cofactor	- The cofactor for arcsinh transformation; typically set as 5 for CyTOF

Value

Returns a matrix with values as the arcsinh ratio of mat normalized to selected column with the desired cofactor

Examples

```
#Example starting with obtaining data from Cytobank
library(CytobankAPI)
popsofinterest<-c("CD4 T cells","NK cells")
cyto_session <- authenticate(site="premium", username="myusername", password="mypassword")
exptno<-2
popsinterest<-getpops(popsofinterest,exptno,cyto_session)
fcs<-getfcsfiles(exptno,cyto_session)
type=TRUE
results<-analyzedata(cyto_session,markersofinterest,popsofinterest,exptno,type)
asinnorm(results,col=2,cofactor=5)

#Example with simple data matrix
data<-matrix(1:9,nrow=3,ncol=3,byrow=TRUE)
colnames(data)<-c("Control","Patient1","Patient2")
rownames(data)<-c("Marker1","Marker2","Marker3")
#Normalizing patient data to control sample with cofactor of 5
asinnorm(data,1,5)
```

calcperevent *Calculates percentages of of cell types of interest out of total cell population*

Description

Calculates percentages of of cell types of interest out of total cell population

Usage

```
calcperevent(results)
```

Arguments

results	- The result of a call to the parseevents function

Value

Returns a matrix with values as percent of first column. Columns correspond to cell types. First column corresponds to the population as a total reference, eg. all live cells run. Rows correspond to fcs files.

Examples

```
#Example starting with obtaining data from Cytobank
library(CytobankAPI)
results1<-statistics.event_counts(UserSession, experiment_id, gate_version = 1,
experiment_version, compensation_id,fcs_files, populations = c("Live","NK cells"),
output = "default",  timeout = UserSession@long_timeout)
popsofinterest<-c("CD4 T cells","NK cells")
cyto_session <- authenticate(site="premium", username="myusername", password="mypassword")
exptno<-2
popsinterest<-getpops(popsofinterest,exptno,cyto_session)
fcs<-getfcsfiles(exptno,cyto_session)
results<-parseevents(results1,popsinterest,fcs)
calcperevent(results)

#Example from simple dataset
data<-matrix(9:1,nrow=3,ncol=3,byrow=FALSE)
rownames(data)<-c("Control","Patient1","Patient2")
colnames(data)<-c("Live cells","Cell type 1","Cell type 2")
calcperevent(data)
```

CytobankAPIstatsGUI *Exports processed events and signaling data*

Description

Exports processed events and signaling data

Usage

```
CytobankAPIstatsGUI()
```

Examples

```
 ## Not run:
library(CytobankAPIstats)
CytobankAPIstatsGUI()
## End(Not run)
```

filterfiles *Filters a list of fcs files by search terms*

Description

Filters a list of fcs files by search terms

Usage

```
filterfiles(files, string)
```

Arguments

files	- List of fcs file IDs with FCS file name as names for list
string	- List of one or more strings of interest as a list to filter samples

Value

Returns a list of file IDs matching with names matching string(s)

Examples

```
#Example starting with obtaining data from Cytobank
library(CytobankAPI)
exptno<-4
cyto_session <- authenticate(site="premium", username="myusername", password="mypassword")
files<-getfcsfiles(exptno,cyto_session)
string<-c("patient","IL-1b")
filterfiles(file,string)

#Simple example when list of file names is already available
files<-1:4
names(files)<-c("Pt1 unst.fcs","Pt2 stim.fcs","Ctrl1 unst.fcs","Ctrl2 stim.fcs")
#Filtering file list to contain only unstimulated files
filterfiles(files,"unstimulated")
#Filtering file list to contain only patient files
filterfiles(files,"Pt")
#Filtering file list to contain both unstimulated and patient files
filterfiles(files,c("Pt","unst"))
```

---

get1status *Filters matrix based on single sample name condition*

Description

Filters matrix based on single sample name condition

Usage

```
get1status(key, results)
```

Arguments

key	- Search string of interest for names
results	- Results matrix with fcs files corresponding to columns

Value

Returns a matrix with columns matching all search keys

Examples

```
#Example starting with obtaining data from Cytobank
library(CytobankAPI)
cyto_session <- authenticate(site="premium", username="myusername", password="mypassword")
markersofinterest<-c("CD3","CD56")
popsofinterest<-c("CD4 T cells","NK cells")
exptID=4
```

```
type=F
results<-analyzedata(cyto_session,markersofinterest,popsofinterest,exptID,type)
key<-c("Patient","Control")
get1status(key,results)

#Example with simple data matrix
data<-matrix(1:18,nrow=3,ncol=6,byrow=TRUE)
colnames(data)<-c("Ctrl1 unst","Pt1 unst","Pt3 unst","Ctrl1 stim","Pt1 stim","Pt3 stim")
rownames(data)<-c("Marker1","Marker2","Marker3")
#Getting all patient samples
get1status("Pt",data)
#Getting all patient and stimulated samples
get1status(c("Pt","stim"),data)
```

`get2status` *Filters matrix columns based on two conditions per file, e.g. patient status, stimluation, time points, etc.*

Description

Filters matrix columns based on two conditions per file, e.g. patient status, stimluation, time points, etc.

Usage

```
get2status(key1, key2, results)
```

Arguments

`key1`	- Search string of interest for names
`key2`	- Search string of interest for names
`results`	- Results matrix with fcs files corresponding to columns

Value

Returns a list of IDs with names matching search both serach strings with names being the description of these features

Examples

```
#Example starting with obtaining data from Cytobank
library(CytobankAPI)
cyto_session <- authenticate(site="premium", username="myusername", password="mypassword")
markersofinterest<-c("CD3","CD56")
popsofinterest<-c("CD4 T cells","NK cells")
exptID=4
type=F
```

```
results<-analyzedata(cyto_session,markersofinterest,popsofinterest,exptID,type)
key1<-"Patient"
key2<-"Untreated"
get2status(key1,key2,results)

#Example with simple data matrix
data<-matrix(1:18,nrow=3,ncol=6,byrow=TRUE)
colnames(data)<-c("Ctrl1 unst","Pt1 unst","Pt3 unst","Ctrl1 stim","Pt1 stim","Pt3 stim")
rownames(data)<-c("Marker1","Marker2","Marker3")
#Getting all stimulated patient samples
get2status(c("Pt","Ctrl"),"stim", data)
#Getting all stimulated patient and control samples
get2status(c("Pt","Ctrl"),"stim", data)
```

`getfcsfiles` *Gets fcs ID numbers and sample names from a given experiment*

Description

Gets fcs ID numbers and sample names from a given experiment

Usage

```
getfcsfiles(exptno, cyto_session)
```

Arguments

`exptno`	- Integer representing an experiment ID on Cytobank account
`cyto_session`	- API authentication token for session

Value

Returns a list of fcs file IDs with names of fcs files as names of list

Examples

```
library(CytobankAPI)
cyto_session <- authenticate(site="premium", username="myusername", password="mypassword")
exptno<-2
getfcsfiles(exptno,cyto_session)
```

getmarkers *Gets appropriate marker IDs for channels of interest*

Description

Gets appropriate marker IDs for channels of interest

Usage

```
getmarkers(markersofinterest, exptno, cyto_session)
```

Arguments

markersofinterest	- Names of channel parameters in Cytobank as list of strings
exptno	- Integer representing an experiment ID on Cytobank account
cyto_session	- API authentication token for session

Value

Returns a list of IDs for markers of interest with names of markers as names of list

Examples

```
library(CytobankAPI)
markersofinterest<-c("CD3","CD56")
cyto_session <- authenticate(site="premium", username="myusername", password="mypassword")
exptno<-2
getmarkers(markersofinterest,exptno,cyto_session)
```

getnewind *Rearranges signaling results matrix with rows in the desired order as outputs*

Description

Rearranges signaling results matrix with rows in the desired order as outputs

Usage

```
getnewind(fixlabels, results)
```

Arguments

fixlabels	- List of strings with desired order of labels
results	- Output of call to parsestats

Value

- Returns a matrix with rows organized in desired order specified by fixlabels

Examples

```
#Example starting with obtaining data from Cytobank
library(CytobankAPI)
cyto_session <- authenticate(site="premium", username="myusername", password="mypassword")
markersofinterest<-c("CD3","CD56")
popsofinterest<-c("CD4 T cells","NK cells")
exptID=4
type=F
results<-analyzedata(cyto_session,markersofinterest,popsofinterest,exptID,type)
fixlabels<-c("CD4 T cells CD56","NK cells CD56","CD4 T cells CD3","NK cells CD3")
getnewind(fixlabels,results)

#Example with simple matrix
data<-matrix(1:8,nrow=4,ncol=2,byrow=TRUE)
colnames(data)<-c("Control","Patient")
rownames(data)<-c("NK cells CD3","CD4 T cells CD3","CD4 T cells CD56","NK cells CD56")
fixlabels<-c("CD4 T cells CD56","NK cells CD56","CD4 T cells CD3","NK cells CD3")
getnewind(fixlabels,data)
```

---

getpops *Gets appropriate gate set IDs for populations of interest*

Description

Gets appropriate gate set IDs for populations of interest

Usage

```
getpops(popsofinterest, exptno, cyto_session)
```

Arguments

popsofinterest	- Names of gates of interest in Cytobank as list of strings
exptno	- Integer representing an experiment ID on Cytobank account
cyto_session	- API authentication token for session

Value

Returns a list of gateSetIDs for populations of interest with names of populations as names of list

Examples

```
library(CytobankAPI)
popsofinterest<-c("CD4 T cells","NK cells")
cyto_session <- authenticate(site="premium", username="myusername", password="mypassword")
exptno<-2
getpops(popsofinterest,exptno,cyto_session)
```

`getrawsignals` *Computes the untransformed medians for cellular markers in populations of interest*

Description

Computes the untransformed medians for cellular markers in populations of interest

Usage

```
getrawsignals(cyto_session, markersofinterest, popsofinterest, exptID,
  markerorder, stimterms, ptterms)
```

Arguments

`cyto_session`	- API authentication token for session
`markersofinterest`	- List of strings of markers of interest, corresponding to names in Cytobank
`popsofinterest`	- List of strings of populations of interest to calculate statistics
`exptID`	- Integer representing an experiment ID on Cytobank account
`markerorder`	- A list of stings corresponding to the desired marker order
`stimterms`	- A list of desired stimulation conditions to analyze in matrix
`ptterms`	- A list of desired sample conditions to analyze in matrix

Value

- Returns matrix of untransformed medians for cellular markers in populations of interest

Examples

```
library(CytobankAPI)
cyto_session <- authenticate(site="premium", username="myusername", password="mypassword")
markersofinterest<-c("CD3","CD56")
popsofinterest<-c("CD4 T cells","NK cells")
exptID=4
markerorder<-c("CD4 T cells CD56","NK cells CD56","CD4 T cells CD3","NK cells CD3")
stimterms<-c("Unstim","IL-15")
ptterm<-c("Pt","Ctrl")
```

```
getrawsignals(cyto_session,markersofinterest,popsofinterest,exptID,markerorder,stimterms,
ptterms)
```

`parseevents` *Modifies the list obtained from a call to statistics.events to a matrix with rows corresponding to fcs files and columns corresponding to the population types*

Description

Modifies the list obtained from a call to statistics.events to a matrix with rows corresponding to fcs files and columns corresponding to the population types

Usage

```
parseevents(results, popsinterest, fcs)
```

Arguments

`results`	- The results of a call to statistics.events function
`popsinterest`	- List of gateSetID numbers for populations of interest with descriptions as names
`fcs`	- List of fcs file IDs of interest with description of FCS files as names

Value

Returns a matrix of event counts with rows corresponding to fcs files and columns corresponding to populations of interest

Examples

```
library(CytobankAPI)

cyto_session <- authenticate(site="premium", username="myusername", password="mypassword")
exptno<-2
popsofinterest1<-c("CD4 T cells","NK cells")
popsinterest<-getpops(popsofinterest1,exptno,cyto_session)
fcs<-getfcsfiles(exptno,cyto_session)
results<-statistics.event_counts(cyto_session, exptno, gate_version = 1,
compensation_id=1,fcs_files=fcs,populations = popsinterest,output = "default",
timeout = UserSession@long_timeout)
parseevents(results,popsinterest,fcs)
```

parsestats *Takes the results of a call to statistics.general and returns a matrix of raw medians with columns corresponding to fcs files and rows to molecules of interest in different cell types*

Description

Takes the results of a call to statistics.general and returns a matrix of raw medians with columns corresponding to fcs files and rows to molecules of interest in different cell types

Usage

```
parsestats(results, popsinterest, fcs, markersofinterest)
```

Arguments

`results`	- The results of a call to statistics.general function
`popsinterest`	- List of gateSetID numbers for populations of interest with descriptions as name
`fcs`	- List of fcs file IDs of interest with description of FCS file names as names
`markersofinterest`	- List of ID numbers for markers of interest with descriptions as name

Value

Returns a matrix of median signaling intensities with columns corresponding to fcs files and rows corresponding to markers of interest in cell types of interest

Examples

```
library(CytobankAPI)

cyto_session <- authenticate(site="premium", username="myusername", password="mypassword")
exptno<-2
popsofinterest1<-c("CD4 T cells","NK cells")
popsinterest<-getpops(popsofinterest1,exptno,cyto_session)
fcs<-getfcsfiles(exptno,cyto_session)
markersofinterest1<-c("CD3","CD56")
markersofinterest<-getmarkers(markersofinterest1,exptno,cyto_session)
results<-statistics.general(UserSession=cyto_session, experiment_id=2, gate_version = -1,
compensation_id=1,fcs_files=fcs, populations = popsinterest,
output = "default",timeout = UserSession@long_timeout)
parsestats(results,popsinterest,fcs,markersofinterest)
```

parsestatsmean *Takes the results of a call to statistics.general and returns a matrix of raw means with columns corresponding to fcs files and rows to molecules of interest in different cell types*

Description

Takes the results of a call to statistics.general and returns a matrix of raw means with columns corresponding to fcs files and rows to molecules of interest in different cell types

Usage

```
parsestatsmean(results, popsinterest, fcs, markersofinterest)
```

Arguments

results	- The results of a call to statistics.general function
popsinterest	- List of gateSetID numbers for populations of interest with descriptions as name
fcs	- List of fcs file IDs of interest with description of fcs file names as names
markersofinterest	- List of ID numbers for markers of interest with descriptions as name

Value

Returns a matrix of mean signaling intensities with columns corresponding to fcs files and rows corresponding to markers of interest in cell types of interest

Examples

```
library(CytobankAPI)
cyto_session <- authenticate(site="premium", username="myusername", password="mypassword")
exptno<-2
popsofinterest1<-c("CD4 T cells","NK cells")
popsinterest<-getpops(popsofinterest1,exptno,cyto_session)
fcs<-getfcsfiles(exptno,cyto_session)
markersofinterest1<-c("CD3","CD56")
markersofinterest<-getmarkers(markersofinterest1,exptno,cyto_session)
results<-statistics.general(UserSession=cyto_session, experiment_id=2, gate_version = -1,
compensation_id=1,fcs_files=fcs, populations = popsinterest,
output = "default",timeout = UserSession@long_timeout)
parsestatsmean(results,popsinterest,fcs,markersofinterest)
```

percentevent *Calculates the percentage of cell populations given an experiment*

Description

Calculates the percentage of cell populations given an experiment

Usage

```
percentevent(cyto_session, markersofinterest, popsofinterest, exptID, grouping,
  specimennames, means)
```

Arguments

cyto_session	- API authentication token for session
markersofinterest	- List of names of channel parameters in Cytobank
popsofinterest	- List of populations of interest to calculate percentages with reference population for percentages listed first
exptID	- Integer representing an experiment ID on Cytobank account
grouping	- A list of indices corresponding to samples from the same donor ex list(c(1,2),c(3,4,5)) if rows 1 and 2 are from pt1,3,4,5 are from pt2, etc.
specimennames	- List of specimen names as strings; needs to be same length as number of groupings
means	- A boolean if mean =TRUE, a mean for all groups in the variable group is calculated, otherwise individual means are returned.

Value

- Returns either the percentage or mean percentage per specimen of each cell type, as specified by mean parameter

Examples

```
library(CytobankAPI)
cyto_session <- authenticate(site="premium", username="myusername", password="mypassword")
markersofinterest<-c("CD3","CD56")
popsofinterest<-c("CD4 T cells","NK cells")
exptID=4
grouping<-list(c(1,2),c(3,4,5),c(6,7))
specimennames<-c("Patient1","Patient2","Control1")
means=T
percentevent(cyto_session,markersofinterest,popsofinterest,exptID,grouping,specimennames,means)
```

Index