

Package ‘COPASutils’

February 19, 2015

Title Tools for processing COPAS large-particle flow cytometer data

Version 0.1.6

Author Tyler Shimko <TShimko126@gmail.com>, Erik Andersen
<erik.andersen@northwestern.edu>

Maintainer Erik Andersen <erik.andersen@northwestern.edu>

Description A logical workflow for the reading, processing, and visualization of data obtained from the Union Biometrica Complex Object Parametric Analyzer and Sorter (COPAS) platform large-particle flow cytometers and a powerful suite of functions for the rapid processing and analysis of large high-throughput screening data sets. It combines the speed of dplyr with the elegance of ggplot2 to make analysis of COPAS data fast and painless.

Depends R (>= 3.0), ggplot2 (>= 0.9), reshape2 (>= 1.0)

Imports dplyr (>= 0.2), kernlab (>= 0.9), stringr (>= 0.6), knitr (>= 1.6)

License GPL (>= 2)

LazyData TRUE

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2015-01-15 06:47:49

R topics documented:

BioSorter	2
bubbleSVMmodel_noProfiler	2
COPASutils	2
doseData	2
edgeEffect	3
extractTime	3
fillWells	3
plateData1	4

plateData2	4
plotCompare	4
plotCorMatrix	5
plotDR	5
plotDR_allTraits	6
plotTrait	6
presentation	7
readPlate	7
readSorter	8
removeWells	8
summarizePlate	9

Index 10

BioSorter	<i>BioSorter</i>
-----------	------------------

Description

Raw data resulting from a BioSorter Machine with LP Sampler

bubbleSVMmodel_noProfiler	<i>bubbleSVMmodel_noProfiler</i>
---------------------------	----------------------------------

Description

Data used by the SVM when removing bubbles in readPlate

COPASutils	<i>COPASutils - Easy reading, processing, and manipulation of COPAS BIOSORT data.</i>
------------	---

Description

COPASutils - Easy reading, processing, and manipulation of COPAS BIOSORT data.

doseData	<i>doseData</i>
----------	-----------------

Description

Data resulting from a dose response curve experiment

edgeEffect	<i>Detect edge effects on 96-well plates</i>
------------	--

Description

Test for an effect of the position of wells in a 96 well plate. This function will split a plate population by edge wells and center wells and test the two populations for significant differences in either a specific trait or all traits if a trait is not specified.

Usage

```
edgeEffect(plate_summary, trait = NULL)
```

Arguments

plate_summary	a summarized and filled plate data frame
trait	a singular trait to test, defaults to NULL and will test all traits

extractTime	<i>Set time to relative</i>
-------------	-----------------------------

Description

Sets time relative to first well run, used in other functions, not meant to be used on its own.

Usage

```
extractTime(plate)
```

Arguments

plate	a plate to extract the time from
-------	----------------------------------

fillWells	<i>Fill in any missing well rows with NA values</i>
-----------	---

Description

Returns a data frame with any missing wells filled in as NA for all measured data

Usage

```
fillWells(plate)
```

Arguments

plate	the data frame of the plate to filled
-------	---------------------------------------

plateData1	<i>plateData1</i>
------------	-------------------

Description

Data resulting from a *C. elegans* GWAS experiment setup

plateData2	<i>plateData2</i>
------------	-------------------

Description

Data resulting from a *C. elegans* GWAS experiment score

plotCompare	<i>Visualize and compare values and distributions across multiple plates</i>
-------------	--

Description

Plot the value (bar plot, if summarized) or distribution (boxplot, if unsummarized) of the data from each well across multiple plates.

Usage

```
plotCompare(plates, trait, plateNames = NULL)
```

Arguments

plates	the list of plate data frames to compare
trait	the trait to compare, entered as a string
plateNames	an optional character vector with the names of the individual plates; if no names are entered, numbers will be used in the order the data frames are entered

Examples

```
#### COPASutils Figures
### Figure 3a
# plotCompare(list(doseData, plateData2), "TOF")

### Figure 3b
# sumDose <- summarizePlate(doseData)
# sumPlate <- summarizePlate(plateData2)
# plotCompare(list(sumDose, sumPlate), "n")
```

plotCorMatrix	<i>Plot a correlation matrix within a plate or between plates</i>
---------------	---

Description

Returns a ggplot2 object of a correlation plot for all traits within a single plate or between two plates. The data from each plate must be summarized prior to being passed to this function.

Usage

```
plotCorMatrix(plate_summary1, plate_summary2 = plate_summary1)
```

Arguments

plate_summary1 one summarized plate to compare in the correlation matrix
plate_summary2 an optional summarized plate to compare with plate_summary1, defaults to plate_summary1; if no argument is entered plate_summary1 will be compared to itself

Examples

```
#### COPASutils Figures  
### Figure 4  
# library(dplyr)  
# sumDose <- summarizePlate(doseData)  
# corDose <- select(sumDose, -n.sorted)  
# plotCorMatrix(corDose)
```

plotDR	<i>Plot a dose response curve by strain across a plate</i>
--------	--

Description

Return a ggplot2 object of a dose response curve by strain across a plate

Usage

```
plotDR(plate_summary, dosages, trait = "n")
```

Arguments

plate_summary the summarized plate data frame to be plotted
dosages a vector of dosages in the plate, entered by row
trait the trait to be plotted on the y axis

Examples

```
#### COPASutils figure
### Figure 5
# strains <- rep(c("A", "B", "C", "D"), each=6, times=4)
# sumDose <- summarizePlate(doseData, strains=strains)
# dose <- rep(c(1, 5, 10, 15, 20, NA), times=16)
# plotDR(sumDose, "n", dosages=dose)
```

plotDR_allTraits *Plot dose response curves for all traits*

Description

Return a list of ggplot2 objects of dose response curves by strain across a plate. Plots for specific traits can be accessed by name from the returned list (i.e. "plots\$n" will return the dose response plot for the trait "n").

Usage

```
plotDR_allTraits(plate_summary, dosages)
```

Arguments

plate_summary the summarized plate data frame to be plotted
dosages a vector of dosages in the plate, entered by row

plotTrait *Create faceted plots for every well in a 96-well plate*

Description

Returns ggplot2 object that is faceted by row and column. By default, it will plot a heat map for the trait specified as a string. Other options include scatterplots and histograms.

Usage

```
plotTrait(plate, trait, trait2 = NULL, type = "heat")
```

Arguments

plate a plate data frame, either summarized or unsummarized, to plot
trait the trait to plot in a heat map or histogram or the independent variable in a scatter plot, enter as a string
trait2 the trait which will be the dependent variable for the scatter plot, enter as a string
type the type of plot, either "heat" for heatmap, "scatter" for scatter plot, or "hist" for histogram, defaults to "heat"

Examples

```
#### COPASutils Figures
### Figure 1a
#plotTrait(doseData, trait="TOF", trait2="EXT", type="scatter")

### Figure 1b
#plotTrait(doseData, trait="TOF", type="hist")

### Figure 1c
# sumDose <- summarizePlate(doseData)
# plotTrait(sumDose, trait="n", type="heat")
```

presentation	<i>presentation</i>
--------------	---------------------

Description

Data for making pretty plots

readPlate	<i>Read in the sorter data with minimal processing</i>
-----------	--

Description

Returns a minimally processed data frame, optionally with SVM-mediated bubble prediction.

Usage

```
readPlate(file, tofmin = 0, tofmax = 10000, extmin = 0, extmax = 10000,
          SVM = TRUE, reflx = TRUE)
```

Arguments

file	path to sorter data file
tofmin	minimum cut off for time of flight, defaults to 0
tofmax	maximum cut off for time of flight, defaults to 10000
extmin	minimum cut off for extinction, defaults to 0
extmax	maximum cut off for extinction, defaults to 10000
SVM	logical dictating whether to predict bubbles with the SVM
reflx	logical indicating whether ReFLx module was used (TRUE) or LP Sampler was used (FALSE), defaults to TRUE

readSorter *Read in raw sorter data*

Description

Reads a raw sorter file into a data frame, removing NA values and any objects not fitting in to the min and max cut offs.

Usage

```
readSorter(file, tofmin = 0, tofmax = 10000, extmin = 0, extmax = 10000,
  reflx = TRUE)
```

Arguments

file	path to sorter data file
tofmin	minimum cut off for time of flight, defaults to 0
tofmax	maximum cut off for time of flight, defaults to 10000
extmin	minimum cut off for extinction, defaults to 0
extmax	maximum cut off for extinction, defaults to 10000
reflx	logical indicating whether ReFLx module was used (TRUE) or LP Sampler was used (FALSE), defaults to TRUE

removeWells *Remove wells from a data frame representing a plate*

Description

Returns a data frame representing a plate with bad wells removed (either with phenotype variables set to NA or rows dropped entirely).

Usage

```
removeWells(plate, badWells, drop = FALSE)
```

Arguments

plate	a processed data frame that has been run through processWells
badWells	a character vector consisting of all wells to remove
drop	a logical value dictating whether to drop wells from the data frame or set measured values to NA, defaults to FALSE

Examples

```
# plate <- summarizePlate(plateData1)
# plateWithoutWells <- removeWells(plate=plate, badWells=c("A1", "A2", "A3"))
```

summarizePlate	<i>Condense all objects examined to appropriate well</i>
----------------	--

Description

Returns a data frame with one row per well with summary statistics.

Usage

```
summarizePlate(plate, strains = NULL, quantiles = FALSE, log = FALSE,  
ends = FALSE)
```

Arguments

plate	plate to summarize, must be run through readSorterData, not readSorter
strains	a vector of all strain (sample) names input row-wise to add to the data frame
quantiles	if TRUE, columns of trait quantiles (every fifth) will be added to the output dataframe, defaults to FALSE
log	if TRUE, columns of log transformed EXT and red fluorescence will be added to output dataframe, defaults to FALSE
ends	if TRUE, columns of min and max values for all traits will be added to output dataframe, defaults to FALSE

Examples

```
# exampleStrains <- rep(c("N2", NA), times=48)  
# plate1 <- summarizePlate(plateData1, quantiles=TRUE, log=TRUE, ends=TRUE)  
# plate2 <- summarizePlate(plateData2, strains=exampleStrains)
```

Index

BioSorter, [2](#)
bubbleSVMmodel_noProfiler, [2](#)

COPASutils, [2](#)
COPASutils-package (COPASutils), [2](#)

doseData, [2](#)

edgeEffect, [3](#)
extractTime, [3](#)

fillWells, [3](#)

plateData1, [4](#)
plateData2, [4](#)
plotCompare, [4](#)
plotCorMatrix, [5](#)
plotDR, [5](#)
plotDR_allTraits, [6](#)
plotTrait, [6](#)
presentation, [7](#)

readPlate, [7](#)
readSorter, [8](#)
removeWells, [8](#)

summarizePlate, [9](#)