Package 'diskImageR'

October 13, 2022

Title A Pipeline to Analyze Resistance and Tolerance from Drug Disk Diffusion Assays

Version 1.0.0

Description A pipeline to analyze photographs of disk diffusion plates. This removes the need to analyze the plates themselves, and thus analysis can be done separate from the assay. Furthermore, diskImageR removes potential researcher bias, by quantitative assessment of drug resistance as the zone diameter at multiple cutoff values of growth inhibition. This method also extends the disk diffusion assay by measuring drug tolerance (in addition to drug resistance) as the fraction of the subpopulation that is able to grow above the resistance point (``FoG''), and drug sensitivity as the rate of change from no growth to full growth (``slope'').

Depends R (>= 3.0.3)

SystemRequirements ImageJ (all OS), Xcode (Mac)

License GPL-3

LazyData true

Imports zoo, subplex, tcltk, methods

Suggests knitr

NeedsCompilation no

Author Aleeza Gerstein [aut, cre],
Inbal Hecht [ctb]

Maintainer Aleeza Gerstein <gerst035@umn.edu>

Repository CRAN

Date/Publication 2016-04-03 16:29:24

R topics documented:

---

addType *Add factor column*

Description

Add an extra factor ("type") column to the existing dataframe

Usage

```
addType(projectName, typePlace = 3, typeName = "type2", save = TRUE)
```

Arguments

projectName	the short name in use for the project.
typePlace	a number that indicates the position of the photograph name to be stored as the 'type' vector'. Defaults to 3. For more details see IJMacro
typeName	a character string that indicates what to name the typeVector. Defaults to "type2".
save	denotes whether to overwrite the existing .csv file or just update the .df in the R global environment. Defaults to TRUE.

Value

updates the existing dataframe

Examples

```
## Not run:
addType("myProject", typePlace=4, typeName="temperature")

## End(Not run)
```

aggregateData *Averages photographs of the same type*

Description

Uses a user-supplied variance measure (currently supported: standard error, coefficient of variation, built-in R functions (e.g., sd) to calculate variance among photographs of the same type

Usage

```
aggregateData(projectName, varFunc = "se", replicate = c("line", "type"),
  overwrite = TRUE, save = TRUE)
```

Arguments

projectName	the short name in use for the project.
varFunc	what type of variation measurment to perform. Currently supports varFunc = "se" to calculate the standard error, varFun = "cv" to calculate the coefficient of variation or any built-in R function (e.g., sd).
replicate	a character vector indicating which the column names that contain which factors to use. Defaults to c("line", "type"). Note that if the typeVector name was changed in createDataframe this should be reflected here.
overwrite	a logical value indicating whether to overwrite existing aggregate dataframe for the same project name. This allows you to save different dataframes averaging across different factors or using different variance measures
save	denotes whether to overwrite the existing .csv file or just update the .df in the R global environment. Defaults to TRUE.

Value

A dataframe "projectName.ag" is saved to the global environment and a .csv file "projectName_ag.csv" is exported to the "parameter_files" directory.

See Also

addType if there multiple factors in your experiment. Add whatever the new factor is called (default: "type2") to the replicate vector if this is appropriate.

Examples

```
## Not run:
aggregateData("myProject")
aggregateData("myProject", varFunc= "sd", replicate = c("line", "drugAmt"), overwrite = FALSE)

## End(Not run)
```

`calcMIC` *Convert RAD to MIC based on built-in or provided parameters or datasets*

Description

Used to convert RAD into MIC. In all cases the linear relationship is log2(MIC) regressed onto RAD^2. This conversion can be based on a) existing built-in data from a number of species/drug combinations, b) a user-supplied slope and intercept of log2(MIC) regressed on RAD^2 for the species/drug combination of interest, c) a user supplied file containing MIC information from lines previously analyzed by diskImageR for RAD, or d) a user supplied file containing both RAD and MIC information. Note that user-supplied data should be MIC and RAD, not log2(MIC) and RAD^2 (the function will do this authomatically).

Usage

```
calcMIC(projectName, type = "df", RAD = "20", height = 4, width = 6,
  addBreakpoints = TRUE, savePDF = TRUE, popUp = TRUE)
```

Arguments

`projectName`	the short name you have been using for the project.
`type`	specify whether the dataset to use is a dataframe with all data ("df") or an aggregated dataframe ("ag")
`RAD`	a numeric value the the critical level of the radius of inhibition (i.e., resistance) parameter to use for MIC. Currently only `RAD` = "80" (80% reduction in growth), `RAD` = "50" (50% reduction in growth), and `RAD` = "20" (20% reduction in growth) are supported [Default = "20"]
`height`	a numeric value indicating the height of the pdf file generated
`width`	a numeric value indicating the width of the pdf file generated
`addBreakpoints`	Indicates whether to add breakpoint lines to the standard curve plot (if the user has supplied data to generate a standard curve)
`savePDF`	a logical value indicating whether to save a PDF file or open a new quartz window. Defaults to TRUE (saves a pdf file).
`popUp`	a logical value indicating whether to pop up the figure after it has been created

Value

In all cases the function will return an updated .csv file that contains the MIC values that correspond to calculated RAD values in the directory "parameter_files" in the main project directory. If the user has supplied their own MIC data the function will also save the calculated model parameters into a separate file and will plot the linear relationship and line of best fit.

createDataframe *Dataframe creation*

Description

createDataframe saves the calculated resistance, perseverence and sensitivity estimates

Usage

```
createDataframe(projectName, clearHalo, diskDiam = 6, maxDist = 30,
  standardLoc = 2.5, removeClear = FALSE, nameVector = TRUE,
  typeVector = TRUE, typePlace = 2, typeName = "type")
```

Arguments

projectName	the short name in use for the project.
clearHalo	numeric value that indicates which picture should be used to represent a clear halo (i.e., the clear space beside the disk).
diskDiam	the diameter of the diffusion disk in mm, defaults to 6.
maxDist	a numeric value indicating the maximum distance away from the disk to be considered. Defaults to 30mm.
standardLoc	a numberic value indicating the location (on the disk) to use to standardize intensity across photographs. The position of standardLoc is a position that should theoretically have the same intensity in all photographs, i.e., the white of the disk. The defaul value (2.5mm) was chosen after testing of 6mm disks that contain some writing. If smaller disks are used standardLoc should be scaled appropriately. You can see where standardLoc falls in each photograph in plotRaw (the red dashed line). To suppress this standardization use standardLoc = FALSE.
removeClear	a logical value that indicates whether to remove the clear halo picture from the dataset (i.e., is this picture an experimental picture, or one solely included to use as a clear halo). Defaults to FALSE.
nameVector	either a logial value or a character vector. Supported values are nameVector = "TRUE" to assign the photograph name to the 'name' column, nameVector = "FALSE" to assign th photograph number to the 'name' column, or nameVector = a vector the same length as the number of photographs indicating the desired names.
typeVector	a logical value. typeVector = "TRUE" will add a 'type' vector to the dataframe using values found in the typePlace position of the photograph names (see IJMacro for more details) while typeVector = "FALSE" will not add a type column.
typePlace	a number that indicates the position of the photograph name to be stored as the 'type' vector'. Defaults to 2. For more details see IJMacro
typeName	a character string that indicates what to name the typeVector. Defaults to "type".

Details

A dataframe with 11 columns:

- **name:** determined by `nameVector`, either photograph names, photograph numbers, or a user-supplied list of names
- **line:** the first components of the `namesVector`; everything that comes before the first "_" in the photograph name
- **type:** the location within the `name` of the phograph type is supplied by `typePlace`. Use `addType` if more than one type column are desired.
- **RAD80, RAD50, RAD20:** resistance parameters, coresponding to the distance in mm of 80%, 50% and 20% reduction in growth
- **FoG80, FoG50, FoG20:** perseverence parameters, coresponding to the fraction of growth achieved above the 80%, 50% and 20% reduction in growth points
- **slope:** sensitivity parameter, indicating the slope at the midpoint, i.e., how rapidly the population changes from low growth to full growth

Value

A dataframe "projectName.df" is saved to the global environment and a .csv file "projectName_df.csv" is exported to the "parameter_files" directory.

Examples

```
## Not run:
createDataframe("myProject", clearHalo=1)
createDataframe("myProject", clearHalo=1, removeClear = TRUE, typeName = "drugAmt")

## End(Not run)
```

`IJMacro` *Run an imageJ analysis macro on the folder that contains the photograph to be analyzed*

Description

`IJMacro` is used to run the imageJ analysis component of diskImageR and then load in the acquired output from imageJ into R.

Usage

```
IJMacro(projectName, projectDir = NA, photoDir = NA, imageJLoc = NA,
  diskDiam = 6)
```

Arguments

projectName	the short name you want use for the project
projectDir	the path to the project directory where all analyses will be saved. If left as NA (the default) you will be able to specify the locaion through a pop-up box. (default=NA)
photoDir	the path to the directory where the photographs are to be analyzed. If left as NA (the default) you will be able to specify the locaion through a pop-up box. (default=NA)
imageJLoc	the absolute path to ImageJ (ImageJ) on your computer. Leave as NA (the default) if you have downloaded ImageJ to a standard location (Mac: /Applications/ImageJ.app or /Applications/ImageJ/ImageJ.app/; Windows: Program Files/ImageJ). If you wish to run imageJ from an alternative path use imageJLoc to specify the absolute path.
diskDiam	the diameter of the diffusion disk in mm, defaults to 6.

Details

Each photograph in the directory specified by photoDir is input into ImageJ, where the built-in 'find particles' macro is used to find the center of a drug diffusion disk of the size specified by diskDiam. Lines are drawn every 5 degrees out from the center of the disk, and the pixel intensity, which corresponds to cell density, is measured using the 'plot-profile' macro along each line. The results from all lines are saved into the "imageJ-out" directory in the specified projectDir. The average pixel intensity is then determined across all 72 lines for each photograph and saved to projectName.
Note that the photograph names can be fairly important downstream and should follow a fairly strict convention to be able to take advantage of some of the built-in functions. Photographs should be named "line_factor1_factor2_factor3_...".

Value

A .csv file is saved to the directory "imageJ_out" in the directory specified by projectDir. The average line for each photograph is saved to the list projectName in the global environment.

Important

There can not be any spaces or special characters in any of the folder names that are in the path that lead to either the main project directory or the photograph directory. If there are an error box titled "Macro Error" will pop up and the script will not run. The project name should ideally be fairly short (easy to type without typos!) and specific to the project. It must start with a letter, not a number or special character, but can otherwise be anything. The project name must always be specified with quotation marks around it (a surprisingly common error).

Examples

```
## Not run:
IJMacro("myProject")

## End(Not run)
```

maxLik *Maximum Likelihood Inference*

Description

maxLik uses maximum likelihood to find the logistic and double logistic equations that best describe the shape of the imageJ output data to then fit parameters that describe reistance, tolerance and sensitivity.

Usage

```
maxLik(projectName, clearHalo, diskDiam = 6, maxDist = 30,
  standardLoc = 2.5, ymax = 125, xplots = 5, height = 8, width = 8,
  FoG = 20, RAD = "all", needML = TRUE, popUp = TRUE,
  nameVector = TRUE, overwrite = TRUE, plotParam = TRUE, plotFoG = TRUE,
  savePDF = TRUE, plotSub = NA, plotCompon = FALSE)
```

Arguments

projectName	the short name in use for the project.
clearHalo	numeric value that indicates which picture should be used to represent a clear halo (i.e., the clear space beside the disk).
diskDiam	the diameter of the diffusion disk in mm, defaults to 6.
maxDist	a numeric value indicating the maximum distance away from the disk to be considered. Defaults to 30mm.
standardLoc	a numberic value indicating the location (on the disk) to use to standardize intensity across photographs. The position of standardLoc is a position that should theoretically have the same intensity in all photographs, i.e., the white of the disk. The defaul value (2.5mm) was chosen after testing of 6mm disks that contain some writing. If smaller disks are used standardLoc should be scaled appropriately. You can see where standardLoc falls in each photograph in plotRaw (the red dashed line). To suppress this standardization use standardLoc = FALSE.
ymax	a numeric value indicating the maximum y value plotted in each graph, does not influence maximum likelihood fitting
xplots	a numeric value indicating how many plots to plot in each row, does not influence maximum likelihood fitting
height	a numeric value indicating the height of the pdf file generated, does not influence maximum likelihood fitting
width	a numeric value indicating the width of the pdf file generated, does not influence maximum likelihood fitting
FoG	a numeric value indicating the the critical level of the area under the curve to plot, does not influence maximum likelihood fitting. Current only FoG = "80" (80% reduction in growth), FoG = "50" (50% reduction in growth), and FoG = "20" (20% reduction in growth) are supported. Note, the critical level for FoG can be different than that chosen for RAD.

RAD	a numeric value indicating the critical level of the radius of inhibition (i.e., resistance) parameter to plot, does not influence maximum likelihood fitting. Currently only RAD = "80" (80% reduction in growth), RAD = "50" (50% reduction in growth), RAD = "20" (20% reduction in growth), and RAD = "all" are supported.
needML	a logical value indicating whether the maximum likelihood results already exist in the global environment or not. If maxLik has already been run in this session then needML can be set to FALSE, which allows the user to replot the results without the need to rerun the time consuming maximum likelihood models. Defaults to TRUE.
popUp	a logical value indicating whether to pop up the figure after it has been created.
nameVector	either a logial value indicating whether to plot the photograph names above the graph or not or a vector the same length as the number of pictures containing the desired names. Defaults to TRUE.
overwrite	a logical value indicating whether to overwrite existing figures created on the same day for the same project name.defaults to TRUE.
plotParam	a logical value indicating whether to save plots containing, at minimum, the fitted logistic equatoin and specified RAD levels to plot, but may also include the FoG plotFoG = "TRUE" or the components of the logistic equation plotCompon = "TRUE". Defaults to TRUE.
plotFoG	a logical value indicating whether to plot the FoG or not. Defaults to TRUE
savePDF	a logical value indicating whether to save a PDF file or open a new quartz. Defaults to TRUE.
plotSub	allows you to plot only a subset of photographs - indicate with a vector the corresponding numeric indices of the data you wish to plot. Photographs are numbered alphabetically by name, and the photograph numbers can also be found by using the showNum option in plotRaw. Defaults to NA, which will plot data from all photographs. Note this does not affect the analysis component, all data is always analyzed.
plotCompon	plots the two terms of the double logistic equation. Defaults to FALSE

Details

maxLik searches for the maximum likelihood (ML) parameter for a single logistic and double logistic equation using the pixel intensity information previously determined from IJMacro. The equations fit are single logistic ('ML'): y = asymA*exp(scalA(x-od50A))\(1+exp(scalA(x-od50A)))+N(0, sigma) double logistic ('ML2'): y = asymA*exp(scalA(x-od50A))\(1+exp(scalA(x-od50A)))+asymB*exp(scalB(x-od50B)))\(1+exp(scalB(x-od50B)))+N(0, sigma) where asymA and asymB are the two asymptotes, od50A and odB are the midpoints (of the two curves), scalA and scalB are the slopes at odA and odB divided by asymA/4 and asymB/4, respectively. Specifically, maxLik uses thesubplex method of optim, as implemented in find.mle. The single logistic is the essentially the same as the double, yet fits only a single asymptote, midpoint and slope. The results from the double logistic fit are used in createDataframe to find the resistance points as well as to fit the area under the curve and thus tolerance, the single logistic is used to determine the midpoint of the curve which is later used to find sensitivity, i.e., the slope at this midpoint.

Value

Two lists, ML and ML2 are saved to the global environment. A pdf file with one plot for each photograph is saved to visualize the results of curve fitting, zone of inhibition (resistance) and the area under the curve (tolerance).

Important

The photograph specified with `clearHalo` is extremely important to determine tolerance, as the intensity beside the disk for the chosen photograph is subtracted for all photographs. Choosing the photograph to be used for this purpose is the only subjective aspect of this pipeline; lighting and camera settings will determine the degre to which the hue of the plate backbground changes among different photographs. Care should be taken to ensure that plate background will be as similar as possible among different plates. Photographs are numbered alphabetically by name, and can also be found using `plotRaw`, showNum = TRUE. In many experiments a suitable strain will already be included, however a good practice is to always take a photograph of a blank plate with just the disk in the center to use for this purpose (and save it with a name like "a" so that it is always the first photograph in the list (i.e., 'clearHalo = 1'). The (non)results from this photograph can be later removed in the function 'createDataframe()'.

Warning

Depending on the number of photographs to be analyzed, 'maxLik()' can take a fair amount of time, upwards of an hour or more. This is due to the maximum likelihood fitting procedures, which determine the best fit parameters from multiple different starting values. The status is indicated by a series of dots (".") in the R console, with one dot per photograph. If for some reason the procedure gets halted in the middle of `maxLik()` (e.g., computer is shut down) as long as R remains open it should resume where it left off when the computer is reactivated.

References

Richard G. Fitzjohn (2012) Diversitree: comparative phylogenetic analyses of diversification in R. Methods in Ecology and Evolution. 3:1084-1092.

See Also

`saveMLParam` to save the parameter estimates for asym, od50, scal and sigma, as well as the log likelihood of the single and double logistic models.

Examples

```
## Not run:
maxLik("myProject", clearHalo=1)
maxLik("myProject", clearHalo=1, xplots = 2, height = 4, width = 6, needML = FALSE)

## End(Not run)
```

oneParamPlot *Used to plot a single parameter*

Description

This function creates a pdf figure of plots showing the results of the imageJ analysis for resistance (radius from the disk, RAD), sensitivity (slope) and tolerance (fraction of growth above RAD, FoG).

Usage

```
oneParamPlot(projectName, type, param = "RAD20", ymin = 0, ymax = 100,
  width = 6, height = 4, xlabels = "line", xlabAngle = NA, order = NA,
  orderFactor = "line", overwrite = TRUE, savePDF = TRUE, popUp = TRUE,
  barplot = TRUE)
```

Arguments

projectName	the short name to be used for the project
type	specify whether the dataset to use is a dataframe with all data ("df") or an aggregated dataframe ("ag")
param	what parameter to plot (supported: "RAD20", "RAD50", "RAD80", "FoG20", "FoG50", "FoG80", "slope"), default = "RAD20"
ymin	a numeric value indicating the minimum y value plotted in each plot
ymax	a numeric value indicating the maximum y value plotted in each plot
width	a numeric value indicating the width of the pdf file generated
height	a numeric value indicating the height of the pdf file generated
xlabels	either a vector containing the desired x-axis labels, or a single value indicating the column name that contains the values to use (likely either the 'line' column or one of the type columns), default = "line".
xlabAngle	indicates whether to print the x axis labels on a angle, if a number is provided this will be the angle used. The defauilt is not to plot on an angle, default = NA.
order	can be either "factor" or "custom". If custom, supply a numberial vector the same length as the dataframe to indicate the desired order. If factor, supply the column name in `ordeFactor` to be used to factor.
orderFactor	if `order = "factor"` supply the column name to be used to factor.
overwrite	a logical value indicating whether to overwrite existing figures created on the same day for the same project name
savePDF	a logical value indicating whether to save a PDF file or open a new quartz window. Defaults to TRUE (saves a pdf file).
popUp	a logical value indicating whether to pop up the figure after it has been created
barplot	whether to plot values as a barplot (barplot = TRUE) or dotplot (barplot = FALSE), default = TRUE. Only possible when `type = "ag"`

Details

Basic parameter plotting functions to plot a single parameter. Input can be the dataframe from either `createDataframe` `type="df"` or from `aggregateData` `type=="ag"`.

Value

Either a pdf figure figure (projectName_RAD-FoG.pdf) saved to the 'figures' directory or a figure on screen

See Also

`twoParamPlot` for a similar figure with two parameters or `threeParamPlot` for a similar figure with three parameters

---

`plotRaw` *Used to plot the results of the imageJ analysis*

Description

`plotRaw` creates a pdf figure showing the results of the ImageJ analysis, with one plot for each photograph. This function is optional, and is primarily for visualization purposes.

Usage

```
plotRaw(projectName, ymin = 0, ymax = 250, xmin = 0, xmax = 40,
  xplots = 6, height = 4, width = 8, cexPt = 0.6, cexX = 0.8,
  cexY = 0.8, standardLoc = 2.5, nameVector = TRUE,
  plotStandardLoc = TRUE, showNum = FALSE, popUp = TRUE,
  overwrite = TRUE, savePDF = TRUE)
```

Arguments

`projectName`	the short name to be used for the project
`ymin`	a numeric value indicating the minimum y value plotted in each plot
`ymax`	a numeric value indicating the maximum y value plotted in each plot
`xmin`	a numeric value indicating the minimum x value plotted in each plot
`xmax`	a numeric value indicating the maximum x value plotted in each plot
`xplots`	a numeric value indicating how many plots to plot in each row
`height`	a numeric value indicating the height of the pdf file generated
`width`	a numeric value indicating the width of the pdf file generated
`cexPt`	a numeric value indicating the size to plot for points
`cexX`	a numeric value indicating the size of x-axis text
`cexY`	a numeric value indicating the size of y-axis text

standardLoc	a numberic value indicating the location (on the disk) to use to standardize intensity across photographs. The position of standardLoc is a position that should theoretically have the same intensity in all photographs, i.e., the white of the disk. The defaul value (2.5mm) was chosen after testing of 6mm disks that contain some writing. If smaller disks are used standardLoc should be scaled appropriately.
nameVector	either a logial value indicating whether to plot the photograph names above the plot; a vector the same length as the number of photographs containing the desired names. Defaults to TRUE.
plotStandardLoc	a logical value indicating whether to draw a dashed vertical line at the standardization point that is used in `maxLik` and `createDataframe`
showNum	a logical value indicating whether to annotate each plot with the photograph number (determined alphabetically from the photograph names). This can be helpful for later functions that require the numerical place of a photograph with a clear halo.
popUp	a logical value indicating whether to pop up the figure after it has been created
overwrite	a logical value indicating whether to overwrite existing figures created on the same day for the same project name
savePDF	a logical value indicating whether to save a PDF file or open a new quartz window. Defaults to TRUE (saves a pdf file).

Value

A pdf file with one plot for each photograph is saved to visualize the results of imageJ analyses

Examples

```
## Not run:
plotRaw("myProject")
plotRaw("myProject", ymin = 50, ymax = 300, xplots=2, height=3, width=4, plotStandardLoc=FALSE)

## End(Not run)
```

`readExistingDF` *Read in an existing dataframe using the tcltk interface*

Description

Open an existing dataframe previously created with either `createDataframe` or `aggregateData` using tcltk interface.

Usage

```
readExistingDF(projectName)
```

Arguments

projectName	is the short name you want use for the project. Note, this can be different than the previous projectName used.

Value

projectName is saved to the global directory and can be used for aggregateData.

---

readInExistingIJ *Used to read in existing imageJ analyses*

Description

This function can be used to read in existing imageJ analyses following IJMacro. Running this function will prompt the user to select the main project folder and to select the directory that contains

Usage

```
readInExistingIJ(projectName, newList = list())
```

Arguments

projectName	the name to be used for project. This name should be short, and can be different than what was used originally for the imageJ analysis step.
newList	dummy variable

---

saveMLParam *Save maximum likelihood output*

Description

Saves the output of maximum likelihood functions - asym, od50, scal, sigma and lnLik.

Usage

```
saveMLParam(projectName)
```

Arguments

projectName	the short name in use for the project.

Value

A dataframe "projectName_ML.df" is saved to the global environment and a .csv file "projectName_ML.csv" is exported to the "parameter_files" directory.

Author(s)

Aleeza C. Gerstein

`threeParamPlot` *Used to plot the RAD, slope and FoG parameter results*

Description

This function creates a pdf figure of plots showing the results of the imageJ analysis for resistance (RAD), tolerance (FoG), and sensitivity (slope).

Usage

```
threeParamPlot(projectName, type, RAD = "RAD20", FoG = "FoG20",
  RADmin = 30, slopeMax = 160, tolMax = 100, width = 6, height = 4,
  xlabels = "line", xlabAngle = NA, order = NA, orderFactor = "line",
  overwrite = TRUE, savePDF = TRUE, popUp = TRUE)
```

Arguments

`projectName`	the short name to be used for the project
`type`	specify whether the dataset to use is a dataframe with all data ("df") or an aggregated dataframe ("ag")
`RAD`	specify the RAD (radius) parameter to be plotted ("RAD20", "RAD50" or "RAD80"), default = "RAD20".
`FoG`	specify the FoG (fraction of growth) parameter to be plotted ("FoG20", "FoG50" or "FoG80"), default = "FoG20".
`RADmin`	minimum distance from the disk for resistance plot (minimum y axis value), default = 30.
`slopeMax`	maximum y axis value for slope (sensitivity) plot
`tolMax`	maximum y axis value for tolerance plot. Note tolerance is coverted to a percent, default = 100.
`width`	a numeric value indicating the width of the pdf file generated
`height`	a numeric value indicating the height of the pdf file generated
`xlabels`	either a vector containing the desired x-axis labels, or a single value indicating the column name that contains the values to use (likely either the 'line' column or one of the type columns), default = "line".
`xlabAngle`	indicates whether to print the x axis labels on a angle, if a number is provided this will be the angle used. The defauilt is not to plot on an angle, default = NA.
`order`	can be either "factor" or "custom". If custom, supply a numberial vector the same length as the dataframe to indicate the desired order. If factor, supply the column name in `ordeFactor` to be used to factor.
`orderFactor`	if `order = "factor"` supply the column name to be used to factor.

overwrite	a logical value indicating whether to overwrite existing figures created on the same day for the same project name
savePDF	a logical value indicating whether to save a PDF file or open a new quartz window. Defaults to TRUE (saves a pdf file).
popUp	a logical value indicating whether to pop up the figure after it has been created

Details

Basic parameter plotting functions for three parameter plots (RAD, FoG , slope). Input can be the dataframe from either createDataframe type="df" or from aggregateData type=="ag". The default is to plot tolerance as a barplot and RAD and slope as a dotplot; tolerance can also be plotted as a dotplot with barplot=FALSE though there is currently not support to plot either RAD or slope as a barplot in this framework.

Value

Either a pdf figure figure saved to the 'figures' directory ("projectName_RAD-slope-FoG.pdf" or a figure on screen

See Also

oneParamPlot for a similar figure with one parameter and twoParamPlot for a similar figure with two parameters

---

twoParamPlot *Used to plot the RAD and FoG results*

Description

This function creates a pdf figure of plots showing the results of the imageJ analysis for resistance (RAD) and tolerance (FoG).

Usage

```
twoParamPlot(projectName, type, RAD = "RAD20", FoG = "FoG20", RADmin = 30,
  tolMax = 100, width = 6, height = 4, xlabels = "line",
  xlabAngle = NA, order = NA, orderFactor = "line", overwrite = TRUE,
  savePDF = TRUE, popUp = TRUE, barplot = TRUE)
```

Arguments

projectName	the short name to be used for the project
type	specify whether the dataset to use is a dataframe with all data ("df") or an aggregated dataframe ("ag")
RAD	specify the RAD (radius) parameter to be plotted ("RAD20", "RAD50" or "RAD80"), default = "RAD20".

FoG	specify the FoG (fraction of growth) parameter to be plotted ("FoG20", "FoG50" or "FoG80"), default = "FoG20".
RADmin	minimum distance from the disk for resistance plot (minimum y axis value), default = 30.
tolMax	maximum y axis value for tolerance plot. Note tolerance is coverted to a percent, default = 100.
width	a numeric value indicating the width of the pdf file generated
height	a numeric value indicating the height of the pdf file generated
xlabels	either a vector containing the desired x-axis labels, or a single value indicating the column name that contains the values to use (likely either the 'line' column or one of the type columns), default = "line".
xlabAngle	indicates whether to print the x axis labels on a angle, if a number is provided this will be the angle used. The defauilt is not to plot on an angle, default = NA.
order	can be either "factor" or "custom". If custom, supply a numberial vector the same length as the dataframe to indicate the desired order. If factor, supply the column name in ordeFactor to be used to factor.
orderFactor	if order = "factor" supply the column name to be used to factor.
overwrite	a logical value indicating whether to overwrite existing figures created on the same day for the same project name
savePDF	a logical value indicating whether to save a PDF file or open a new quartz window. Defaults to TRUE (saves a pdf file).
popUp	a logical value indicating whether to pop up the figure after it has been created
barplot	whether to plot tolerance as a barplot (barplot = TRUE) or dotplot (barplot = FALSE), default = TRUE. Only possible when type = "ag"

Details

Basic parameter plotting functions to plot RAD and FoG parameter plots. Input can be the dataframe from either createDataframe type="df" or from aggregateData type=="ag". The default is to plot RAD as a dotplot and tolerance as a barplot, though tolerance can also be plotted as a dotplot with barplot=FALSE (currently there is not support to plot RAD as a barplot in this framework).

Value

Either a pdf figure figure (projectName_RAD-FoG.pdf) saved to the 'figures' directory or a figure on screen

See Also

oneParamPlot for a similar figure with one parameter or threeParamPlot for a similar figure with three parameters

Index