Package 'toxplot'

April 3, 2018

Type Package

Title Batch Processing, Modeling and Visualizing the Dose-Response of High-Throughput Screening Bioassay

Version 0.1.1

Author Jun Wang

Maintainer Jun Wang <njekin@gmail.com>

Description A convenient interface to batch process high-throughput toxicology bioassay screening data. It's designed specifically for screening experiment that features a primary inhibition-type assay and a companion cytotoxicity assay. This package provides functions for data normalization, quality-control analysis, dose-response curve fitting (using the Hill model provided in the 'tcpl' package), visualization, and a unique toxicity-adjusted potency ranking system.

URL http://github.com/njekin/ToxPlot-R-Package

License GPL (>= 2)

Encoding UTF-8

LazyData true

Depends R (>= 3.3.0)

Imports dplyr (>= 0.5.0), tidyr (>= 0.5.0), tcpl (>= 1.0.0), ggplot2 (>= 2.2.0), ggthemes (>= 3.4.0)

Suggests knitr, rmarkdown, testthat, plotly, devtools,

RoxygenNote 6.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2018-04-03 07:56:35 UTC

R topics documented:

`auc_hill_tcpl` *function to calculate Area Under the Curve (AUC) of the hill model*

Description

function to calculate Area Under the Curve (AUC) of the hill model

Usage

```
auc_hill_tcpl(p, lower, upper)
```

Arguments

`p`	a vector containing the Hill model parameters: top, log AC50, hill coefficient
`lower`	lower boundary of x for integration
`upper`	upper boundary of x for integration

Value

calculated area under the curve (AUC) value

`demo_mc` *Example multi-concentration screening data*

Description

A dataset contains example data for a primary inhibition assay and the parallel cytotoxicity assay

Usage

```
demo_mc
```

Format

See vignette for description of the demo data.

demo_mc_norm *Normalized example multi-concentration screening data*

Description

A dataset contains normalized example data for a primary inhibition assay and the parallel cytotoxicity assay

Usage

```
demo_mc_norm
```

Format

See vignette for description of the demo data.

fit_curve_tcpl *fit dose-resopnse curve using tcpl hill model*

Description

Curve fitting using the tcplFit function in 'tcpl' package. Chemicals are modelled based on spid. If you want to model the same chemical (e.g. positive controls), then assign different spid to this chemical so the function can separate them out. Absolute IC20 and absolute IC50 are calculated as well.

Usage

```
fit_curve_tcpl(df, assay_info, prim_cutoff = 20, toxi_cutoff = 20)
```

Arguments

df	input data contain normalized assay readings
assay_info	predefined names for primary and cytotoxicity assays, use NULL if either one of the assay does not need to be modeled.
prim_cutoff	significance cutoff for primary assay (eg. 3sigma or 3bMAD)
toxi_cutoff	significance cutoff for cytotoxicity assay (eg. 3sigma or 3bMAD)

Value

A list object containing modeling results, the corresponding data for each chemical.

Examples

```
## fit curve with default significant threshold 20

demo_md <- fit_curve_tcpl(demo_mc_norm, assay_info =
list(prim_assay = "Primary", toxi_assay = "Cytotox"))

## start from raw data
# define assay
assay_info <- list(prim_assay = "Primary",toxi_assay = "Cytotox")
# data normalization
demo_mc_norm <- normalize_per_plate(demo_mc, nctrl = "DMSO")
# filter out two test chemicals
demo_mc_norm <- dplyr::filter(demo_mc_norm, spid %in% c("TP0001502B05", "TP0001502B01"))
# fit curve with default 20% threshold
demo_md <- fit_curve_tcpl(demo_mc_norm, assay_info)

## fit curve with specified significance threshold
demo_md <- fit_curve_tcpl(demo_mc_norm, assay_info, prim_cutoff = 25, toxi_cutoff = 25)
```

`hill_model` *hill mode in ToxCast tcpl package*

Description

hill mode in ToxCast tcpl package

Usage

```
hill_model(p, x)
```

Arguments

p	a vector containing the Hill model parameters: top, log AC50, hill coefficient
x	a vector of log concentrations

Value

calculated y value based on the x and model parameters

log_abs_ec *calculate absolute EC_anything based on tcpl hill model*

Description

calculate absolute EC_anything based on tcpl hill model

Usage

```
log_abs_ec(p, y)
```

Arguments

p	a vector containing the Hill model parameters: top, log AC50, hill coefficient
y	the y value

Value

calculated x value

normalize_per_plate *normalize per plate*

Description

normalize raw readings as percent of median vehicle control wells

Usage

```
normalize_per_plate(dt, nctrl = "DMSO")
```

Arguments

dt	data.frame contains essential columns with the raw data.
nctrl	the name (spid) of the vehicle/solvent control used for calculation

Value

data.frame with normalized value columns. 'nval_mean' column is the normalized value calculated using the mean of vehicle control wells, 'nval_median' column is the normalized value calculated using the median of vehicle control wells.

Examples

```
## normalize demo data
demo_mc_norm <- normalize_per_plate(demo_mc, nctrl = "DMSO")
```

`normalize_single_assay` *normalize per plate (from single assay data)*

Description

normalize raw readings as percent of median/mean vehicle control wells, per assay, per plate. This function is called by normalize_per_plate, should not be called directly by user. #'

Usage

```
normalize_single_assay(dt, nctrl)
```

Arguments

`dt`	data.frame contains essential columns with the raw data.
`nctrl`	the name (spid) of the vehicle/solvent control used for calculation

Value

data.frame with normalized value columns. 'nval_mean' column is the normalized value calculated using the mean of vehicle control wells, 'nval_median' column is the normalized value calculated using the median of vehicle control wells.

`plot_tcpl` *Plot dose-resonse curves based on the tcpl hill model*

Description

Produce the plot for the dose-response curves and data points for both primary and toxicity assay. The direction of the data and dose-resonse curves are presented as the original data, rather than the uptrend direction required by the 'tcpl' function. Plots are sorted by the ranking_score.

Usage

```
plot_tcpl(tcpl_models, rank_table = NULL, spid_chnm_table = NULL,
  notation = FALSE, cunit = "M")
```

Arguments

`tcpl_models`	the list object created by 'fit_curve_tcpl' function
`rank_table`	the data.frame output from 'rank_tcpl' function
`spid_chnm_table`	the spid, chnm, casn info table
`notation`	value can be TRUE or FALSE, determine whehter to show potency metrics on the plot
`cunit`	the unit of concentration, on default is "M" (molar).

Value

list of ggplot2 objects, each corresponding to one spid.

Examples

```
## produce plots without notations
demo_md <- fit_curve_tcpl(demo_mc_norm, assay_info =
list(prim_assay = "Primary", toxi_assay = "Cytotox"))
plots <- plot_tcpl(demo_md)


## start from raw data
# define assay
assay_info <- list(prim_assay = "Primary",toxi_assay = "Cytotox")
# data normalization
demo_mc_norm <- normalize_per_plate(demo_mc, nctrl = "DMSO")
# filter out two test chemicals
demo_mc_norm <- dplyr::filter(demo_mc_norm, spid %in% c("TP0001502B05", "TP0001502B01"))
# fit curve with default 20% threshold
demo_md <- fit_curve_tcpl(demo_mc_norm, assay_info)
# calculate TAA and Med_diff only
demo_rank <- rank_tcpl(demo_md, med_taa = NULL, med_med_diff = NULL)
#produce plots with notations
demo_plots <- plot_tcpl(demo_md, demo_rank, notation = TRUE)

##produce plots with notations, with changed concentration unit displayed on the plot
demo_plots <- plot_tcpl(demo_md, demo_rank, notation = TRUE, cunit = "uM")
```

`plot_tcpl_minimal` *Plot dose-resonse curves with minimal text annotation This funciton plots dose-response curve with minimal text annotation, no x and y axis label, 0 borders. Useful when need to present several plots together.*

Description

Plot dose-resonse curves with minimal text annotation This funciton plots dose-response curve with minimal text annotation, no x and y axis label, 0 borders. Useful when need to present several plots together.

Usage

```
plot_tcpl_minimal(tcpl_models, rank_table = NULL, spid_chnm_table = NULL,
  notation = FALSE, cunit = "M")
```

Arguments

tcpl_models	the list object created by 'fit_curve_tcpl' function
rank_table	the data.frame output from 'rank_tcpl' function
spid_chnm_table	the spid, chnm, casn info table
notation	value can be TRUE or FALSE, determine whehter to show potency metrics on the plot
cunit	the unit of concentration, on default is "M" (molar).

Value

list of ggplot2 objects, each corresponding to one spid.

Examples

```
## produce plots without notations
demo_md <- fit_curve_tcpl(demo_mc_norm, assay_info =
list(prim_assay = "Primary", toxi_assay = "Cytotox"))
plots_minimal <- plot_tcpl_minimal(demo_md)

## start from raw data
# define assay
assay_info <- list(prim_assay = "Primary",toxi_assay = "Cytotox")
# data normalization
demo_mc_norm <- normalize_per_plate(demo_mc, nctrl = "DMSO")
# filter out two test chemicals
demo_mc_norm <- dplyr::filter(demo_mc_norm, spid %in% c("TP0001502B05", "TP0001502B01"))
# fit curve with default 20% threshold
demo_md <- fit_curve_tcpl(demo_mc_norm, assay_info)
# calculate TAA and Med_diff only
demo_rank <- rank_tcpl(demo_md, med_taa = NULL, med_med_diff = NULL)
#produce plots with notations
demo_plots <- plot_tcpl_minimal(demo_md, demo_rank, notation = TRUE)

##produce plots with notations, with changed concentration unit displayed on the plot
demo_plots <- plot_tcpl_minimal(demo_md, demo_rank, notation = TRUE, cunit = "uM")
```

---

qc_per_plate *Quality-control metrics calculation*

Description

Calculate QC metrics, includin Z' score, CV of DMSO negative control, per assay plate.

Usage

```
qc_per_plate(d, assay_info, resp = "nval_median")
```

Arguments

d	data.frame contains essential columns with the raw data.
assay_info	assay_info list, contains names of primary and cytotox assay, names must match what are provided in the raw data, under the column 'assay'.
resp	response type, specify either 'nval_median' or 'nval_mean' for QC calculation

Value

three dataframe each representing negative control stats, positive control stats and QC metrics (CV_DMSO, Z' score, SSMD) for each assay plate

Examples

```
## calculate QC measures from demo data
assay_info <- list(prim_assay = "Primary",toxi_assay = "Cytotox")
demo_mc_norm <- normalize_per_plate(demo_mc, nctrl = "DMSO")
qc <- qc_per_plate(demo_mc_norm, assay_info)
```

---

rank_tcpl *function to calculate ranking score, TAA, med_diff, EC values based on tcpl hill model*

Description

calculate ranking score, TAA, med_diff, absolute EC values, AC50, based on the hill model in tcpl package

Usage

```
rank_tcpl(tcpl_models, spid_chnm_table = NULL, med_taa = NULL,
  med_med_diff = NULL)
```

Arguments

tcpl_models	the list object returned by 'fit_curve_tcpl' function
spid_chnm_table	a reference table with 'spid' and the corresponding chemical name 'chnm' column, and the CAS number 'casn' column.
med_taa	the median TAA value from reference chemical, if not supplied, then ranking score won't be calculated.
med_med_diff	the median Median-Difference from reference chemical, if not supplied, then ranking score won't be calculated.

Value

a dataframe containing ranking metrics for each chemical (spid)

Examples

```
## start with normalized data
demo_md <- fit_curve_tcpl(demo_mc_norm, assay_info =
list(prim_assay = "Primary", toxi_assay = "Cytotox"))
demo_rank <- rank_tcpl(demo_md)


## start from raw data
# define assay
assay_info <- list(prim_assay = "Primary",toxi_assay = "Cytotox")
# data normalization
demo_mc_norm <- normalize_per_plate(demo_mc, nctrl = "DMSO")
# filter out two test chemicals
demo_mc_norm <- dplyr::filter(demo_mc_norm, spid %in% c("TP0001502B05", "TP0001502B01"))
# fit curve with default 20% threshold
demo_md <- fit_curve_tcpl(demo_mc_norm, assay_info)
# calculate TAA and Med_diff only
demo_rank <- rank_tcpl(demo_md, med_taa = NULL, med_med_diff = NULL)

## calculate ranking score with specified median TAA and meidan Med_Difference
demo_rank <- rank_tcpl(demo_md, med_taa = 150, med_med_diff = 92)
```

---

round_df *round digits of numbers*

Description

round numbers in a datafram to specified digits

Usage

```
round_df(df, digits)
```

Arguments

df	the dataframe input
digits	the specified number of digits

Value

a dataframe

save_plot_pdf *save plots in pdf*

Description

save ggplot2 plots generated in a list to a pdf file

Usage

```
save_plot_pdf(plot_list, filename)
```

Arguments

plot_list	the r list object contains all ggplot2 objects
filename	the output file name, including the file directory

Examples

```
## start from raw data
# define assay
assay_info <- list(prim_assay = "Primary",toxi_assay = "Cytotox")
# data normalization
demo_mc_norm <- normalize_per_plate(demo_mc, nctrl = "DMSO")
# filter out two test chemicals
demo_mc_norm <- dplyr::filter(demo_mc_norm, spid %in% c("TP0001502B05", "TP0001502B01"))
# fit curve with default 20% threshold
demo_md <- fit_curve_tcpl(demo_mc_norm, assay_info)
# calculate TAA and Med_diff only
demo_rank <- rank_tcpl(demo_md, med_taa = NULL, med_med_diff = NULL)
#produce plots with notations
demo_plots <- plot_tcpl_minimal(demo_md, demo_rank, notation = TRUE)

## save all the plots as pdf
# save_plot_pdf(demo_plots, ".\output plots\all_plots.pdf")

## save the 1st plot as pdf
# save_plot_pdf(demo_plots[1], ".\output plots\plot1.pdf")
```

summary_tcpl *function to summarize curve fitting results*

Description

function to summarize curve fitting results

Usage

```
summary_tcpl(tcpl_models, spid_chnm_table = NULL)
```

Arguments

tcpl_models	the list object returned by 'fit_curve_tcpl' function
spid_chnm_table	a reference table with 'spid' and the corresponding chemical name 'chnm' column, and the CAS number 'casn' column.

Value

a data.frame contains summarized metrics for each chemical (spid)

Examples

```
## supply models as the essential argument. spid_chnm_table is optional.
demo_md <- fit_curve_tcpl(demo_mc_norm, assay_info =
list(prim_assay = "Primary", toxi_assay = "Cytotox"))
demo_sum <- summary_tcpl(demo_md)


## start from raw data
# define assay
assay_info <- list(prim_assay = "Primary",toxi_assay = "Cytotox")
# data normalization
demo_mc_norm <- normalize_per_plate(demo_mc, nctrl = "DMSO")
# filter out two test chemicals
demo_mc_norm <- dplyr::filter(demo_mc_norm, spid %in% c("TP0001502B05", "TP0001502B01"))
# fit curve with default 20% threshold
demo_md <- fit_curve_tcpl(demo_mc_norm, assay_info)
# obtain summary table
demo_sum <- summary_tcpl(demo_md)
```

Index