Package 'Rcurvep'

January 7, 2021

Type Package

Title Concentration-Response Data Analysis using Curvep

Version 1.2.0

Description Provide an R interface for processing concentration-response datasets using Curvep, a response noise filtering algorithm. The algorithm was described in the publications (Sedykh A et al. (2011) <doi:10.1289/ehp.1002476> and Sedykh A (2016) <doi:10.1007/978-1-4939-6346-1_14>). Other parametric fitting approaches (e.g., Hill equation) are also adopted for ease of comparison. Also, methods for calculating the confidence interval around the activity metrics are also provided. The methods are based on the bootstrap approach to simulate the datasets (Hsieh J-H et al. <doi:10.1093/toxsci/kfy258>). The simulated datasets can be used to derive the baseline noise threshold in an assay endpoint. This threshold is critical in the toxicological studies to derive the point-of-departure (POD).

Language en-US

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Imports dplyr (>= 0.7), tibble, magrittr, tidyselect, boot, tidyr, purrr, rlang, stringr, ggplot2, Rdpack, methods

RdMacros Rdpack

Suggests testthat, knitr, rmarkdown, tcpl

VignetteBuilder knitr

RoxygenNote 7.1.1

Depends R (>= 3.5)

NeedsCompilation no

Author Jui-Hua Hsieh [aut, cre], Alexander Sedykh [aut]

Maintainer Jui-Hua Hsieh <juihua.hsieh@gmail.com>

Repository CRAN

Date/Publication 2021-01-07 19:10:02 UTC

R topics documented:

`cal_knee_point` *Calculate the knee point on the exponential-like curve*

Description

Currently two methods have been implemented to get the "keen-point" from the variance(y) - threshold(x) curve. One is to use the original y values to draw a straight line between the lowest x value (p1) to highest x value (p2). The knee-point is the x that has the longest distance to the line. The other one is to fit the data first then use the fitted responses to do the same analysis. Currently the first method is preferred.

Usage

```
cal_knee_point(d, xaxis, yaxis, p1 = NULL, p2 = NULL, plot = TRUE)
```

Arguments

`d`	A tibble.
`xaxis`	The column name in the `d` to be the x-axis in the exponential-like curve
`yaxis`	The column name in the `d` to be the y-axis in the exponential-like curve
`p1`	Default = NULL, or an integer value to manually set the first index of line.
`p2`	Default = NULL, or an integer value to manually set the last index of line.
`plot`	Default = TRUE, plot the diagnostic plot.

Value

A list with two components: stats and outcome.

- stats: a tibble, including pooled variance (pvar), fitted responses (y_exp_fit, y_lm_fit), distance to the line (dist2l)
- outcome: a tibble, including estimated BMRs (bmr)

Suffix in the *stats* and *outcome* tibble: *ori (original values),* exp(exponential fit). prefix in the *outcome* tibble: cor *(correlation between the fitted responses and the original responses), bmr* (benchmark response), qc (quality control).

See Also

`estimate_dataset_bmr()`

Examples

```
inp <- data.frame(
x = seq(5, 95, by = 5),
y = c(0.0537, 0.0281, 0.0119, 0.0109, 0.0062, 0.0043, 0.0043, 0.0042,
0.0041, 0.0043, 0.0044, 0.0044, 0.0046, 0.0051,
0.0055, 0.0057, 0.0072, 0.0068, 0.0035)
)

out <- cal_knee_point(inp,"x", "y", plot = FALSE)
plot(out)
```

---

combi_run_rcurvep *Run Curvep on datasets of concentration-response data with a combination of Curvep parameters*

Description

It simplifies the steps of `run_rcurvep()` by wrapping the `create_dataset()` in the function.

Usage

```
combi_run_rcurvep(
  d,
  n_samples = NULL,
  vdata = NULL,
  mask = 0,
  keep_sets = c("act_set", "resp_set", "fp_set"),
  ...
)
```

Arguments

d	Datasets with concentration-response data. Examples are zfishbeh and zfishdev.
n_samples	NULL (default) for not to simulate responses or an integer number to indicate the number of responses per concentration to simulate.
vdata	NULL (default) for not to simulate responses or a vector of numeric responses in vehicle control wells to use as error. This parameter only works when n_samples is not NULL; an experimental feature.
mask	Default = 0, for no mask (values in the mask column all 0). Use a vector of integers to mask the responses: 1 to mask the response at the highest concentration; 2 to mask the response at the second highest concentration, and so on. If mask column exists, the setting will be ignored.
keep_sets	The types of output to be reported. Allowed values: act_set, resp_set, fp_set. Multiple values are allowed. act_set is the must. • act_set: activity data • resp_set: response data • fp_set: fingerprint data
...	Curvep settings. See curvep_defaults() for allowed parameters. These can be used to overwrite the default values.

Value

An rcurvep object. It has two components: result, config The result component is also a list of output sets depending on the parameter, *keep_sets*. The config component is a *curvep_config* object.

Often used columns in the *act_set*: AUC (area under the curve), wAUC (weighted AUC), POD (point-of-departure), EC50 (Half maximal effective concentration), nCorrected (number of corrected points).

See Also

run_rcurvep()

Examples

```
data(zfishbeh)

# 2 simulated sample curves +
# using two thresholds +
# mask the response at the higest concentration
# only to output the act_set

out <- combi_run_rcurvep(
  zfishbeh,
  n_samples = 2,
  TRSH = c(5, 10),
  mask = 1,
```

```
  keep_sets = "act_set")

# create the zfishdev_act dataset


data(zfishdev_all)
zfishdev_act <- combi_run_rcurvep(
  zfishdev_all, n_samples = 100, keep_sets = c("act_set"),TRSH = seq(5, 95, by = 5),
  RNGE = 1000000, CARR = 20, seed = 300
)
```

`create_dataset` *Create concentration-response datasets that can be applied in the* `run_rcurvep()`

Description

The input dataset is created either by summarizing the response data or by simulating the response data.

Usage

```
create_dataset(d, n_samples = NULL, vdata = NULL)
```

Arguments

`d`	Datasets with concentration-response data. Examples are zfishbeh and zfishdev.
`n_samples`	NULL (default) for not to simulate responses or an integer number to indicate the number of responses per concentration to simulate.
`vdata`	NULL (default) for not to simulate responses or a vector of numeric responses in vehicle control wells to use as error. This parameter only works when n_samples is not NULL; an experimental feature.

Details

Curvep requires 1-to-1 concentration response relationship. For the dataset that does not meet the requirement, the following strategies are applied:

Summary (when n_samples = NULL):

- For dichotomous responses, percentage is reported (n_in/N*100).
- For continuous responses, median value of responses per concentration is reported.

Simulation (when n_samples is a positive integer):

- For dichotomous responses, bootstrap approach is used on the "n_in" vector to create a vector of percent response.
- For continuous responses, options are a) direct sampling; b) responses from the linear fit using the original data + error of responses based on the supplied vehicle control data

Value

The original dataset with a new column, sample_id (if n_samples is not NULL) or the summarized dataset with columns as zfishbeh.

See Also

run_rcurvep()

Examples

```
# datasets with continuous response data
data(zfishbeh)

## default
d <- create_dataset(zfishbeh)

## add samples
d <- create_dataset(zfishbeh, n_samples = 3)

## add samples and vdata
d <- create_dataset(zfishbeh, n_samples = 3, vdata = rnorm(100))

# dataset with dichotomous response data
data(zfishdev)

## default
d <- create_dataset(zfishdev)

## add samples
d <- create_dataset(zfishdev, n_samples = 3)
```

---

curvep *The Curvep function to process one set of concentration-response data*

Description

The relationship between concentration and response has to be 1 to 1. The function is the backbone of run_rcurvep() and combi_run_rcurvep().

Usage

```
curvep(
  Conc,
  Resp,
  Mask = NULL,
  TRSH = 15,
  RNGE = -100,
```

```
  MXDV = 5,
  CARR = 0,
  BSFT = 3,
  USHP = 4,
  TrustHi = FALSE,
  StrictImp = TRUE,
  DUMV = -999,
  TLOG = -24,
  ...
)
```

Arguments

Conc	Array of concentrations, e.g., in Molar units, can be log-transformed, in which case internal log-transformation is skipped.
Resp	Array of responses at corresponding concentrations, e.g., raw measurements or normalized to controls.
Mask	array of 1/0 flags indicating invalidated measurements (default = NULL).
TRSH	Base(zero-)line threshold (default = 15).
RNGE	Target range of responses (default = -100).
MXDV	Maximum allowed deviation from monotonicity (default = 5).
CARR	Carryover detection threshold (default = 0, analysis skipped if set to 0)
BSFT	For baseline shift issue, min.#points to detect baseline shift (default = 3, analysis skipped if set to 0).
USHP	For u-shape curves, min.#points to avoid flattening (default = 4, analysis skipped if set to 0).
TrustHi	For equal sets of corrections, trusts those retaining measurements at high concentrations (default = FALSE).
StrictImp	It prevents extrapolating over concentration-range boundaries; used for POD, ECxx etc (default = TRUE).
DUMV	A dummy value, default = -999.
TLOG	A scaling factor for calculating the wAUC, default = -24.
...	allow other parameters to pass

Value

A list with corrected concentration-response measurements and several calculated curve metrics.

- resp: corrected responses
- corr: flags for corrections
- ECxx: effective concentration values at various thresholds
- Cxx: concentrations for various absolute response levels
- Emax: maximum effective concentration, slope of the mid-curve (b/w EC25 and EC75)
- wConc: response-weighted concentration

- wResp: concentration-weighed response
- POD: point-of-departure (first concentration with response >TRSH)
- AUC: area-under-curve (in units of log-concentration X response)
- wAUC: AUC weighted by concentration range and POD / TLOG (-24)
- wAUC_pre: AUC weighted by concentration range and POD
- nCorrected: number of points corrected (basically, sum of flags in corr)
- Comments: warning and notes about the dose-response curve
- Settings: input parameters for this run

References

Sedykh A, Zhu H, Tang H, Zhang L, Richard A, Rusyn I, Tropsha A (2011-March). "Use of in vitro HTS-derived concentration-response data as biological descriptors improves the accuracy of QSAR models of in vivo toxicity." *Environmental health perspectives*, **119**, 364–370. doi: 10.1289/ehp.1002476, http://europepmc.org/articles/PMC3060000.

Sedykh A (2016). "CurveP Method for Rendering High-Throughput Screening Dose-Response Data into Digital Fingerprints." *Methods in molecular biology (Clifton, N.J.)*, **1473**, 135–141. doi: 10.1007/9781493963461_14.

See Also

`run_rcurvep()` and `combi_run_rcurvep()`

Examples

```
curvep(Conc = c(-8, -7, -6, -5, -4) , Resp = c(0, -3, -5, -15, -30))
```

`curvep_defaults` *Default parameters of Curvep*

Description

Default parameters of Curvep

Usage

```
curvep_defaults()
```

Value

A list of parameters with class as curvep_config.

- TRSH: (default = 15) base(zero-)line threshold
- RNGE: (default = -1000000, decreasing) target range of responses
- MXDV: (default = 5) maximum allowed deviation from monotonicity
- CARR: (default = 0) carryover detection threshold (analysis skipped if set to 0)
- BSFT: (default = 3) for baseline shift issue, min.#points to detect baseline shift (analysis skipped if set to 0)
- USHP: (default = 4) for u-shape curves, min.#points to avoid flattening (analysis skipped if set to 0)
- TrustHi: (default = TRUE)for equal sets of corrections, trusts those retaining measurements at high concentrations
- StrictImp: (default = TRUE) prevents extrapolating over concentration-range boundaries; used for POD, ECxx etc.
- DUMV: (default = -999) dummy value for inactive (not suggested to modify)
- TLOG: (default = -24) denominator for calculation wAUC (not suggested to modify)
- seed: (default = NA) can be set when bootstrapping samples

See Also

curvep()

Examples

```
# display all default settings
curvep_defaults()

# customize settings
custom_settings <- curvep_defaults()
custom_settings$TRSH <- 30
custom_settings
```

`estimate_dataset_bmr` *Estimate benchmark response (BMR) for each dataset*

Description

Currently two methods have been implemented to get the "keen-point" from the variance(y) - threshold(x) curve. One is to use the original y values to draw a straight line between the lowest x value (p1) to highest x value (p2). The knee-point is the x that has the longest distance to the line. The other one is to fit the data first then use the fitted responses to do the same analysis. Currently the first method is preferred.

Usage

```
estimate_dataset_bmr(d, p1 = NULL, p2 = NULL, plot = TRUE)
```

Arguments

d	The rcurvep object with multiple samples and TRSHs. See combi_run_rcurvep() for an example.
p1	Default = NULL, or an integer value to manually set the first index of line.
p2	Default = NULL, or an integer value to manually set the last index of line.
plot	Default = TRUE, plot the diagnostic plot.

Details

The estimated BMR can be used in the calculation of POD. For example, if bmr = 25. For Curvep, `combi_run_rcurvep(zfishbeh,TRSH = 25)`.
For Hill fit, `summarize_fit_output(run_fit(zfishbeh,modls = "hill"),thr_resp = 25,extract_only = TRUE)`.

Value

A list with two components: stats and outcome.

- stats: a tibble, including pooled variance (pvar), fitted responses (y_exp_fit, y_lm_fit), distance to the line (dist2l)
- outcome: a tibble, including estimated BMRs (bmr)

Suffix in the *stats* and *outcome* tibble: *ori (original values),* exp(exponential fit). prefix in the *outcome* tibble: cor *(correlation between the fitted responses and the original responses), bmr* (benchmark response), qc (quality control).

See Also

cal_knee_point(), combi_run_rcurvep()

Examples

```
# no extra cleaning
data(zfishdev_act)
bmr_out <- estimate_dataset_bmr(zfishdev_act, plot = FALSE)
plot(bmr_out)

# if want to do extra cleaning...
actm <- summarize_rcurvep_output(zfishdev_act, clean_only = TRUE, inactivate = "CARRY_OVER")

bmr_out <- estimate_dataset_bmr(actm, plot = FALSE)
```

`fit_modls` *Fit one set of concentration-response data using types of models*

Description

A convenient function to fit data using available models and to sort the outcomes by AIC values.

Usage

```
fit_modls(Conc, Resp, Mask = NULL, modls = c("hill", "cnst"), ...)
```

Arguments

`Conc`	A vector of log10 concentrations.
`Resp`	A vector of numeric responses.
`Mask`	Default = NULL or a vector of 1 or 0. 1 is for masking the respective response.
`modls`	The model types for the fitting. Multiple values are allowed. Currently Hill model (hill) and constant model (cnst) are implemented. Default = c("hill", "cnst").
`...`	The named input configurations for replacing the default configurations. The input configuration needs to add model type as the prefix. For example, hill_pdir = -1 will set the Hill fit only to the decreasing direction.

Details

The backbone of fit using hill and cnst is based on the implementation from tcpl package. But the lower bound of ga is lower by log10(1/100).

Value

A list of components named by the models. The models are sorted by their AIC values. Thus, the first component has the best fit.

hill:

Fit output from Hill equation

- modl: model type, i.e., hill
- fit: fittable, 1 (yes) or 0 (no)
- aic: AIC value
- tp: model top, <0 means the fit for decreasing direction is preferred
- ga: ac50 (log10 scale)
- gw: Hill coefficient
- er: scale term for Student's t distribution

cnst:

Fit output from constant model

- modl: model type, i.e., cnst
- fit: fittable?, 1 or 0
- aic: AIC value
- er: scale term

See Also

tcpl::tcplObjHill(), tcpl::tcplObjCnst(), get_hill_fit_config()

Examples

```
concd <- c(-9, -8, -7, -6, -5, -4)
respd <- c(0, 2, 30, 40, 50, 60)
maskd <- c(0, 0, 0, 0, 0, 1)

# run hill only
fit_modls(concd, respd, modls = "hill")

# run hill only + increasing direction only
fit_modls(concd, respd, modls = "hill", hill_pdir = 1)

# run with mask at the highest concentration
fit_modls(concd, respd, maskd)
```

---

get_hill_fit_config *Get the default configurations for the Hill fit*

Description

The function gives the default settings by using one set of concentration-response data.

Usage

```
get_hill_fit_config(Conc, Resp, optimf = "tcplObjHill")
```

Arguments

Conc	A vector of log10 concentrations.
Resp	A vector of numeric responses.
optimf	The default optimized function is tcpl::tcplObjHill(). but can be changed to ObjHillnorm().

Value

A list of input configurations.

- theta: initial values of parameters for Hill equation: tp, ga, gw, er
- f: the object function
- ui: the bound matrix
- ci: the bound constraints

See Also

`tcpl::tcplObjHill()`, `fit_modls()`

---

merge_rcurvep_objs *Merge results from multiple rcurvep objects*

Description

Sometimes user may want to try multiple curvep setting and pick the one that can capture the shape (wAUC != 0). The highest absolute wAUC from the chemical-endpoint(-sample_id) pair will be picked.

Usage

```
merge_rcurvep_objs(...)
```

Arguments

...	rcurvep objects

Value

an updated rcurvep object with config = NULL

Examples

```
data(zfishbeh)

# combine default + mask
out1 <- combi_run_rcurvep(zfishbeh, TRSH = 10)
out2 <- combi_run_rcurvep(zfishbeh, TRSH = 10, mask = 1)
m1 <- merge_rcurvep_objs(out1, out2)

# use same set of samples to combine
out1 <- combi_run_rcurvep(zfishbeh, TRSH = 10, n_samples = 2, seed = 300)
out2 <- combi_run_rcurvep(zfishbeh, TRSH = 10, mask = 1, n_samples = 2, seed = 300)
m1 <- merge_rcurvep_objs(out1, out2)
```

plot.rcurvep_bmr *Plot BMR diagnostic curves*

Description

Plot BMR diagnostic curves

Usage

```
## S3 method for class 'rcurvep_bmr'
plot(x, ...)
```

Arguments

x	The rcurvep_bmr object from estimate_dataset_bmr().
...	Allowed values: n_in_page, number of endpoints in a page.

Value

A ggplot object.

Examples

```
data(zfishdev_act)
bmr_out <- estimate_dataset_bmr(zfishdev_act, plot = FALSE)
plot(bmr_out)
```

Rcurvep *Rcurvep: Concentration-Response Data Analysis using Curvep*

Description

Provide an R interface for processing concentration-response datasets using Curvep, a response noise filtering algorithm. The algorithm was described in the publications (see references below). Other parametric fitting approaches (e.g., Hill equation) are also adopted for ease of comparison. Also, methods for calculating the confidence interval around the activity metrics are also provided. The methods are based on the bootstrap approach to simulate the datasets. The simulated datasets can be used to derive the baseline noise threshold in an assay endpoint. This threshold is critical in the toxicological studies to derive the point-of-departure (POD).

Details

Different strategies are used to simulate the datasets:

- Curvep - bootstrapping the responses of replicates at each concentration
- Hill equation - bootstrapping the residuals and adding back to the fitted responses (by Hill) at each concentration

For Curvep the bootstrapping strategy is different depending on the type of datasets. Datasets can be grouped into three types:

1. dichotomous binary incidence data (e.g. mortality data from alternative animal model data)
2. continuous data with high number of replicates (e.g. alternative animal model data)
3. continuous data with low number of replicates (e.g. in vitro data)

Bootstrapping strategies:

1. bootstrap incidence out of total animals per concentration then calculate percentage of incidence
2. bootstrap replicate responses per concentration directly
3. bootstrap vehicle control responses and add back to the fitted responses by linear regression per concentration (experimental)

To learn more about Rcurvep start with the vignettes: `browseVignettes(package = "Rcurvep")`

References

Curvep:
Sedykh A, Zhu H, Tang H, Zhang L, Richard A, Rusyn I, Tropsha A (2011-March). "Use of in vitro HTS-derived concentration-response data as biological descriptors improves the accuracy of QSAR models of in vivo toxicity." *Environmental health perspectives*, **119**, 364–370. doi: 10.1289/ehp.1002476, http://europepmc.org/articles/PMC3060000.

Sedykh A (2016). "CurveP Method for Rendering High-Throughput Screening Dose-Response Data into Digital Fingerprints." *Methods in molecular biology (Clifton, N.J.)*, **1473**, 135–141. doi: 10.1007/9781493963461_14.

Bootstrap:
Hubbard TD, Hsieh J, Rider CV, Sipes NS, Sedykh A, Collins BJ, Auerbach SS, Xia M, Huang R, Walker NJ, DeVito MJ (2019-March). "Using Tox21 High-Throughput Screening Assays for the Evaluation of Botanical and Dietary Supplements." *Applied in vitro toxicology*, **5**, 10–25. doi: 10.1089/aivt.2018.0020, http://europepmc.org/articles/PMC6442399.

Hsieh J, Ryan K, Sedykh A, Lin J, Shapiro AJ, Parham F, Behl M (2019-January). "Application of Benchmark Concentration (BMC) Analysis on Zebrafish Data: A New Perspective for Quantifying Toxicity in Alternative Animal Models." *Toxicological sciences an official journal of the Society of Toxicology*, **167**, 92–104. doi: 10.1093/toxsci/kfy258, http://europepmc.org/articles/PMC6317423.

`run_fit` *Run parametric fits using types of models on concentration-response datasets*

Description

Confidence intervals of activity metrics can be obtained through bootstrap approach. The bootstrap samples are generated by adding the residuals (the difference between the original responses and the Hill fit) to the fitted response (strictly to Hill equation).

Usage

```
run_fit(
  d,
  modls = c("hill", "cnst"),
  keep_sets = c("fit_set", "resp_set"),
  n_samples = NULL,
  ...
)
```

Arguments

`d`	Datasets with concentration-response data. An example is zfishbeh. mask column is optional.
`modls`	The model types for the fitting. Multiple values are allowed. Currently Hill model (hill) and constant model (cnst) are implemented. Default = c("hill", "cnst").
`keep_sets`	Output datasets. Multiple values are allowed. Default values are fit_set and resp_set. fit_set is a must. • fit_set: a tibble with output from model fits • resp_set: a tibble with fitted response data from the winning model
`n_samples`	NULL (default) for no bootstrap samples are generated or number of samples to be generated from bootstrapping. When n_samples is not NULL, fit_modls = "hill" will be set automatically.
`...`	The named input configurations for replacing the default configurations. The input configuration needs to add model type as the prefix. For example, hill_pdir = -1 will set the Hill fit only to the decreasing direction.

Value

A list of named components: result and result_nested. The result component is also a list of output sets depending on the parameter, *keep_sets*. The result_nested component is a tibble with input data nested in a column, input, and output data nested in a column, output.

The prefix of the column names in the *fit_set* are the used models. The *win_modl* is the winning model.

See Also

fit_modls() for model fit information and the following analyses using summarize_fit_output(). for dichotomous response (see zfishdev), use create_dataset() first.

Examples

```
# default
fitd <- run_fit(zfishbeh)

# use only hill model and fit only to the decreasing direction, keep only the fit_set output
fitd <- run_fit(zfishbeh, modls = "hill", keep_sets = "fit_set", hill_pdir = -1)

# fit to the bootstrap samples
fitd <- run_fit(zfishbeh, n_samples = 2)
```

---

run_rcurvep *Run Curvep on datasets of concentration-response data*

Description

The concentration-response relationship per endpoint and chemical has to be 1-to-1. If not, use create_dataset() for pre-processing or use combi_run_rcurvep(), which has both pre-processing and more flexible parameter controls.

Usage

```
run_rcurvep(
  d,
  mask = 0,
  config = curvep_defaults(),
  keep_sets = c("act_set", "resp_set", "fp_set"),
  ...
)
```

Arguments

d	Datasets with columns: endpoint, chemical, conc, and resp, mask (optional) Example datasets as zfishbeh. It is required that the baseline of responses in the resp column to be 0.
mask	Default = 0, for no mask (values in the mask column all 0). Use a vector of integers to mask the responses: 1 to mask the response at the highest concentration; 2 to mask the response at the second highest concentration, and so on. If mask column exists, the setting will be ignored.
config	Default configurations set by curvep_defaults().

keep_sets — The types of output to be reported. Allowed values: act_set, resp_set, fp_set. Multiple values are allowed. act_set is the must.

- act_set: activity data
- resp_set: response data
- fp_set: fingerprint data

... — Curvep settings. See curvep_defaults() for allowed parameters. These can be used to overwrite the default values.

Value

An rcurvep object. It has two components: result, config The result component is also a list of output sets depending on the parameter, *keep_sets*. The config component is a *curvep_config* object.

Often used columns in the *act_set*: AUC (area under the curve), wAUC (weighted AUC), POD (point-of-departure), EC50 (Half maximal effective concentration), nCorrected (number of corrected points).

See Also

create_dataset(), combi_run_rcurvep(), curvep_defaults().

Examples

```
data(zfishbeh)
d <- create_dataset(zfishbeh)

# default
out <- run_rcurvep(d)

# change TRSH
out <- run_rcurvep(d, TRSH = 30)

# mask response at highest and second highest concentration
out <- run_rcurvep(d, mask = c(1, 2))
```

---

summarize_fit_output *Summarize the results from the parametric fitting using types of models*

Description

The function first extracts the activity data based on the fit the supplied input parameters. In addition, summary of activity data (e.g., confidence interval, hit confidence) can be produced.

Usage

```
summarize_fit_output(
  d,
  thr_resp = 20,
  perc_resp = 10,
  ci_level = 0.95,
  extract_only = FALSE
)
```

Arguments

d	The output from the run_fit().
thr_resp	The response cutoff to calculate the potency. Default = NULL.
perc_resp	The percentage cutoff to calculate the potency. Default = NULL.
ci_level	The confidence level for the activity metrics. Default is = 0.95.
extract_only	Whether act_summary data should be produced. Default = FALSE.

Details

A tibble, act_set is generated. When (extract_only = FALSE), a tibble, act_summary is generated with confidence intervals of the activity metrics. The quantile approach is used to calculate the confidence interval. For potency activity metrics, if value is NA, highest tested concentration is used in the summary. For other activity metrics, if value is NA, 0 is used in the summary.

Value

A list of named components: result and result_nested (and act_summary). The result and result_nested are the copy from the output of run_fit(). An act_set is added under the result component. If (extract_only = FALSE), an act_summary is added.

See Also

run_fit()

Examples

```
# generate some fit outputs

## fit only
fitd1 <- run_fit(zfishbeh)

## fit + bootstrap samples
fitd2 <- run_fit(zfishbeh, n_samples = 3)


# only to extract the activity data
sumd1 <- summarize_fit_output(fitd1, extract_only = TRUE)
```

```
# calculate EC20 instead of default EC10
sumd1 <- summarize_fit_output(fitd1, extract_only = TRUE, perc_resp  = 20)

# calculate POD using a higher noise level (e.g., 40)
## this number depends on the response unit
sumd1 <- summarize_fit_output(fitd1, extract_only = TRUE, thr_resp  = 40)

# calculate confidence intervals based on the bootstrap samples
sumd2 <- summarize_fit_output(fitd2)
```

`summarize_rcurvep_output` *Clean and summarize the output of rcurvep object*

Description

Clean and summarize the output of rcurvep object

Usage

```
summarize_rcurvep_output(
  d,
  inactivate = NULL,
  ci_level = 0.95,
  clean_only = FALSE
)
```

Arguments

`d`	The rcurvep object from `combi_run_rcurvep()` and `run_rcurvep()`.
`inactivate`	A character string, default = NULL, to make the curve with this string in the Comments column as inactive. or a vector of index for the rows in the act_set that needs to be inactive
`ci_level`	Default = 0.95 (95 percent of confidence interval).
`clean_only`	Default = FALSE, only the 1st, 2nd task will be performed (see Details).

Details

The function can perform the following tasks:

1. add an column, hit, in the *act_set*
2. unhit (make result as inactive) if the Comments column contains a certain string
3. summarize the results

The curve is considered as "hit" if its responses are monotonic after processing by Curvep. However, often, if the curve is "INVERSE" (yet monotonic) is not considered as an active curve. By using the information in the Comments column, we can "unhit" these cases.

When (clean_only = FALSE, default), a tibble, act_summary is generated with confidence intervals of the activity metrics. The quantile approach is used to calculate the confidence interval. For potency activity metrics, if value is NA, highest tested concentration is used in the summary. For other activity metrics, if value is NA, 0 is used in the summary.

Value

A list of named components: result and config (and act_summary). The result and config are the copy of the input d (but with modifications if *inactivate* is not NULL). If (clean_only = FALSE), an *act_summary* is added.

Suffix meaning in column names in *act_summary*: med (median), cil (lower end confidence interval), ciu (higher end confidence interval) Often used columns in *act_summary*: n_curves (number of curves used in summary), hit_confidence (fraction of active in n_curves)

See Also

`combi_run_rcurvep()`, `run_rcurvep()`

Examples

```
data(zfishbeh)

# original datasets
out <- combi_run_rcurvep(zfishbeh, n_samples = NULL, TRSH = c(5, 10))
out_res <- summarize_rcurvep_output(out)


# unhit when comment has "INVERSE"
out <- summarize_rcurvep_output(out, inactivate = "INVERSE")

# unhit for certain rows in act_set
out <- summarize_rcurvep_output(out, inactivate = c(2,3))

# simulated datasets
out <- combi_run_rcurvep(zfishbeh, n_samples = 3, TRSH = c(5, 10))
out_res <- summarize_rcurvep_output(out)
```

`zfishbeh` *Subsets of concentration response datasets from zebrafish neurotoxicity assays*

Description

The datasets contain 11 toxicity endpoints and 2 chemicals. The responses have been normalized so that the baseline is 0.

Usage

```
zfishbeh
```

Format

A tibble with 2123 rows and 4 columns:

endpoint endpoint name

chemical chemical name + CASRN

conc concentrations in log10(M) format

resp responses after normalized using the vehicle control on each plate

Source

Biobide study S-BBD-0017/15

`zfishdev` *Subsets of concentration response datasets from zebrafish developmental toxicity assays*

Description

The datasets contain 4 toxicity endpoints and 3 chemicals.

Usage

```
zfishdev
```

Format

A tibble with 96 rows and 5 columns:

endpoint endpoint name + at time point measured

chemical chemical name + CASRN

conc concentrations in log10(M) format

n_in number of incidence

N number of embryos

Source

Biobide study S-BBD-00016/15

zfishdev_act — *Activity output based on simulated datasets using zfishdev_all dataset*

Description

The data is an rcurvep object from the `combi_run_rcurvep()`. See `combi_run_rcurvep()` for the code to reproduce this dataset.

Usage

```
zfishdev_act
```

Format

A list of two named components: result and config. The result component is a list with one component: act_set.

See Also

`estimate_dataset_bmr()`

zfishdev_all — *Full sets of concentration response datasets from zebrafish developmental toxicity assays*

Description

The datasets contain 4 toxicity endpoints and 32 chemicals.

Usage

```
zfishdev_all
```

Format

A tibble with 512 rows and 5 columns:

Source

Biobide study S-BBD-00016/15

See Also

zfishdev

Index