Package 'LW1949'

November 3, 2015

Title An Automated Approach to Evaluating Dose-Effect Experiments Following Litchfield and Wilcoxon (1949)

Version 0.0.0.9008

Description LW1949 takes the manual approach to evaluating dose-effect experiments (Litchfield and Wilcoxon 1949) and automates the steps so that the computer can do the work.

Depends R (>= 3.2.2)

Imports MASS, mgcv, tcltk

Suggests magrittr, testthat, knitr

License GPL

LazyData TRUE

URL https://github.com/JVAdams/LW1949

VignetteBuilder knitr

RoxygenNote 5.0.0

NeedsCompilation no

Author Jean V. Adams [aut, cre]

Maintainer Jean V. Adams <jvadams@usgs.gov>

R topics documented:

`assessfit` *Assess Fit of Dose-Response Curve*

Description

Assess the fit of a dose-response curve using the chi-squared statistic. The curve is described by the intercept and slope of a straight line in the log dose vs. probit effect scale.

Usage

```
assessfit(params, DEdata, fit, simple = TRUE)
```

Arguments

`params`	A numeric vector of length two, with the estimated intercept and slope of the dose-effect relation on the log10 and probit scale. These parameters define the dose-response curve.
`DEdata`	A data frame of dose-effect data (typically, the output from `dataprep`) containing at least these four variables: dose, ntot, pfx, fxcateg.
`fit`	A model object that can be used to predict the corrected values (as proportions) from `distexpprop5`, the distance between the expected values (as proportions) and 0.5. Typically, the output from `gamtable1()`.
`simple`	A logical scalar indicating if the output should be restricted to just the P value, default TRUE.

Details

This function is used to find the dose-response curve that minimizes the chi-squared statistic measuring the distance between the observed and expected values of the response (the proportion affected). Following Litchfield and Wilcoxon (1949, steps B1 and B2), records with expected effects < 0.01% or > 99.99% are deleted, and other expected effects are "corrected" using the `correctval` function.

Value

If `simple=FALSE`, a list of length two. The first element, `chi`, is a numeric vector of length three: `chistat`, chi-squared statistic; `df`, degrees of freedom; and `pval`, P value. The second element, `contrib`, is a matrix of three numeric vectors the same length as `obsn`: `exp`, expected effects; `expcorr`, expected effects corrected; and `contrib`, contributions to the chi-squared.

If `simple=TRUE`, a numeric scalar, the chi-squared statistic (see details).

References

Litchfield, JT Jr. and F Wilcoxon. 1949. A simplified method of evaluating dose-effect experiments. Journal of Pharmacology and Experimental Therapeutics 96(2):99-113. [link].

See Also

chi2 and chisq.test.

Examples

```
conc <- c(0.0625, 0.125, 0.25, 0.5, 1)
numtested <- rep(8, 5)
nalive <- c(1, 4, 4, 7, 8)
mydat <- dataprep(dose=conc, ntot=numtested, nfx=nalive)
gamfit <- gamtable1()
assessfit(log10(c(0.125, 0.5)), mydat, gamfit, simple=FALSE)
```

---

chi2 *Chi-Squared Statistic*

Description

Calculate the chi-squared statistic from observed and expected counts.

Usage

```
chi2(obsn, expn)
```

Arguments

obsn	A numeric vector of observed counts.
expn	A numeric vector of expected counts.

Value

A list of length two. The first element is a numeric vector of length three: chistat, chi-squared statistic; df, degrees of freedom; and pval, P value. The second element is a numeric vector the same length as obsn, contributions to the chi-squared.

See Also

chisq.test.

Examples

```
chi2(c(10, 8, 3), c(7, 7, 7))
```

coefprobit *Calculate the Coefficients of a Probit Regression Fit*

Description

Calculate the coefficients from a fitted probit regression model with confidence intervals.

Usage

```
coefprobit(pfit, alpha = 0.05)
```

Arguments

pfit	An object of class glm representing a probit regression fit to dose-effect data, typically the result of a call to fitprobit. Dose should be the only independent variable in the model.
alpha	A numeric scalar, the significance level used to generate 100*(1 - alpha)% confidence limits, default 0.05.

Value

A numeric vector of length six, the intercept and slope of the dose-response curve, each with 100*(1 - alpha)% confidence limits.

Examples

```
toxdat <- data.frame(
 dose=c(0.05, 0.0625, 0.125, 0.25, 0.5, 1),
 ntot=rep(8, 6),
  nfx = c(0, 1, 4, 4, 6, 8))
myfit <- fitprobit(toxdat)
coefprobit(myfit)
```

constrain *Constrain Data to a Specified Range*

Description

Constrain data to a specified range, assigning values from the specified range to those outside the range, typically for graphing purposes.

Usage

```
constrain(x, xrange)
```

Arguments

x	A numeric vector of values to constrain.
xrange	A numeric vector of length two specifying the constraints, the minimum and maximum value for x.

Value

A numeric vector, the same length as `x`, in which the minimum constraint is assigned to values of `x` less than the minimum, and the maximum constraint is assigned to values of `x` greater than the maximum.

Examples

```
constrain(1:20, c(3, 19))
```

`correctval` *Predict the Corrected Proportional Effect*

Description

Given an expected proportional effect, calculate the "corrected" proportional effect using a model fit of Litchfield and Wilcoxon's (1949) Table 1.

Usage

```
correctval(val, fit)
```

Arguments

`val`	A numeric vector of expected effects (as proportions).
`fit`	A model object to be used to predict the "corrected" effects (as proportions) from `distexpprop5`, the distance between the expected effects (as proportions) and 0.5. Typically the output from `gamtable1()`.

Value

A numeric vector of corrected effects (as proportions), the same length as `val`.

References

Litchfield, JT Jr. and F Wilcoxon. 1949. A simplified method of evaluating dose-effect experiments. Journal of Pharmacology and Experimental Therapeutics 96(2):99-113. [link].

Examples

```
gamfit <- gamtable1()
correctval(c(0.37, 0.5, 0.63), gamfit)
```

`dataprep` *Prepare Data*

Description

Prepare dose-effect data for evaluation.

Usage

```
dataprep(dose, ntot, nfx)
```

Arguments

`dose`	A numeric vector of unique, chemical concentrations (see Details).
`ntot`	A numeric vector of the number of individuals that were tested at each dose.
`nfx`	A numeric vector of the number of individuals that were affected at each dose.

Details

The input data are expected to be summarized by dose. If duplicate doses are provided, an error will be thrown.

Value

A data frame with eight columns (ordered by dose and proportion affected), seven numeric vectors and one logical vector:

- `dose` = chemical concentrations.
- `ntot` = the number of individuals that were tested at each dose.
- `nfx` = the number of individuals that were affected at each dose.
- `rec` = the record number corresponding to the input vectors `dose`, `ntot`, `nfx`.
- `pfx` = the proportion of individuals that were affected at each dose.
- `log10dose` = log transformed dose, `log10(dose)`.
- `bitpfx` = probit transformed proportional affected, `probit(pfx)`.
- `fxcateg` = effects category: 0 for none affected, 100 for all affected, and 50 for other proportions affected.
- `LWkeep` = logical vector identifying records to keep for Litchfield and Wilcoxon (1949, step A1) method.

References

Litchfield, JT Jr. and F Wilcoxon. 1949. A simplified method of evaluating dose-effect experiments. Journal of Pharmacology and Experimental Therapeutics 96(2):99-113. [link].

Examples

```
conc <- c(0.0625, 0.125, 0.25, 0.5, 1)
numtested <- rep(8, 5)
nalive <- c(1, 4, 4, 7, 8)
dataprep(dose=conc, ntot=numtested, nfx=nalive)
```

estimable *Determine if a Dose-Effect Relation is Estimable*

Description

Determine if a dose-effect relation is estimable based on available data.

Usage

```
estimable(DEdata)
```

Arguments

DEdata	A data frame of dose-effect data (typically, the output from dataprep) containing at least two variables: dose, a numeric vector of chemical concentrations, and pfx, a numeric vector of proportional effects at each dose (see Details).

Details

A dose-effect relation is defined to be estimable if and only if there are at least two test records and there is some (non-zero) variability in both the doses and the proportional effects. The input data are expected to be summarized by dose. If duplicate doses are provided, an error will be thrown.

Value

A logical scalar indicating if a dose-effect relation is estimable.

Examples

```
conc <- c(0.0625, 0.125, 0.25, 0.5, 1)
numtested <- rep(8, 5)
nalive <- c(1, 4, 4, 7, 8)
mydat <- dataprep(dose=conc, ntot=numtested, nfx=nalive)
estimable(mydat)
nalive2 <- rep(4, 5)
mydat2 <- dataprep(dose=conc, ntot=numtested, nfx=nalive2)
estimable(mydat2)
```

fill *Fill in Missing Values*

Description

Fill in missing values in a vector, using the last recorded value.

Usage

```
fill(x, resetWhen = rep(FALSE, length(x)))
```

Arguments

x	A vector, can be character, numeric, or logical.
resetWhen	A logical vector, the same length as x, indicating elements that should not be filled in.

Details

Similar to na.locf in the zoo package, but works for "" in character vectors as well.

Value

A vector the same length as x, with all NAs or ""s replaced by the last value for the vector. Note that and missing values at the beginning of the vector will not be replaced.

References

This is a copy of the fill function from the [jvamisc] package.

Examples

```
numvec <- c(NA, 1:5, NA, NA, NA, 10:12, NA)
newgroup <- c(1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0)
fill(numvec)
fill(numvec, newgroup)

charvec <- c("", letters[1:5], "", "", "", letters[10:12], "")
fill(charvec)
```

---

fitlinear *Determine Linear Regression Coefficients from Dose-Effect Data*

Description

Determine coefficients (intercept and slope) from dose-effect data using simple linear regression on the log10 dose vs. probit effect scale.

Usage

```
fitlinear(DEdata, fit, constr = c(1e-04, 0.9999))
```

Arguments

DEdata	A data frame of dose-effect data (typically, the output from dataprep) containing at least three variables: log10dose, bitpfx, and LWkeep.
fit	A model object that can be used to predict the corrected values (as proportions) from distexpprop5, the distance between the expected values (as proportions) and 0.5. Typically the output from gamtable1().
constr	A numeric vector of length two, indicating the constraints (see constrain) applied to the proportional effects, default c(0.0001, 0.9999).

Value

A numeric vector of length two, the estimated intercept and slope.

Examples

```
conc <- c(0.0625, 0.125, 0.25, 0.5, 1)
numtested <- rep(8, 5)
nalive <- c(1, 4, 4, 7, 8)
mydat <- dataprep(dose=conc, ntot=numtested, nfx=nalive)
gamfit <- gamtable1()
fitlinear(mydat, gamfit)
```

`fitLW` *Apply Litchfield and Wilcoxon Evaluation of Dose-Effect Experiments*

Description

Automatically apply Litchfield and Wilcoxon's (1949) evaluation of dose-effect experiments.

Usage

```
fitLW(DEdata)
```

Arguments

`DEdata`	A data frame of dose-effect data (typically, the output from `dataprep`) containing at least eight variables: dose, ntot, nfx, pfx, log10dose, bitpfx, fxcateg, and LWkeep (see Details).

Details

The input data are expected to be summarized by dose. If duplicate doses are provided, an error will be thrown.

Value

A list of length three:

- `chi` = the chi-squared statistic with associated P value and degrees of freedom,
- `params` = the estimated intercept and slope of the dose-response curve on the log10 probit scale,
- `LWest` = the Litchfield Wilcoxon estimates of ED50 with 95% confidence intervals and the number of records with partial effects (npartfx) as well as other metrics used in their step-by-step approach (ED16, ED84, S with 95% confidence intervals, N', and fED50).

References

Litchfield, JT Jr. and F Wilcoxon. 1949. A simplified method of evaluating dose-effect experiments. Journal of Pharmacology and Experimental Therapeutics 96(2):99-113. [link].

Examples

```
dose <- c(0.0625, 0.125, 0.25, 0.5, 1)
ntested <- rep(8, 5)
nalive <- c(1, 4, 4, 7, 8)
mydat <- dataprep(dose=dose, ntot=ntested, nfx=nalive)
mydat
fitLW(mydat)
```

`fitprobit` *Fit a Probit Regression to Dose-Effect Data*

Description

Fit a probit regression to dose-effect data, using the log10 of the dose as the response.

Usage

```
fitprobit(dat)
```

Arguments

`dat`	A data frame of toxicity data, including at least three variables: dose (the concentration of the tested chemical), ntot (the number of individuals tested), and nfx (the number of affected individuals).

Details

Only those rows with `dose > 0`, `ntot > 0`, and `nfx >= 0` are used in fitting the model.

Value

A an object of class `glm`.

Examples

```
toxdat <- data.frame(
 dose=c(0.05, 0.0625, 0.125, 0.25, 0.5, 1),
  ntot=rep(8, 6),
  nfx = c(0, 1, 4, 4, 6, 8))
fitprobit(toxdat)
```

fxcat *Define Effect Category*

Description

Define three effect categories, 0 for none affected, 100 for all affected, and 50 for other proportions affected.

Usage

```
fxcat(dat)
```

Arguments

dat	A data frame of toxicity data, including at least two variables: ntot (the number of individuals tested) and nfx (the number of affected individuals).

Value

An integer vector the same length as prob with categories of 0, 50, or 100.

Examples

```
toxdat <- data.frame(
  dose=c(0.0625, 0.125, 0.25, 0.5),
  ntot=rep(8, 4),
  nfx = c(0, 4, 6, 8))
cbind(toxdat, fxcat(toxdat))
```

gamtable1 *Fit a smooth GAM to Table 1 of Litchfield and Wilcoxon (1949)*

Description

Fit a smooth GAM function to replace looking up values in Table 1 of Litchfield and Wilcoxon (1949).

Usage

```
gamtable1()
```

Details

Note that for an expected value of 37 Table 1 gives a corrected value of 9.4, but for an expected value of 63 it gives a corrected value of 90.5. To ensure that both values add to 100, I used corrected values of 9.45 and 90.55. The expected and corrected values from Table 1 are then used to build a GAM model, which is used as input to the correctval function.

Value

A gamObject that can be used to predict the corrected values (as proportions) from distexpprop5, the distance bewteen the expected values (as proportions) and 0.5

References

Litchfield, JT Jr. and F Wilcoxon. 1949. A simplified method of evaluating dose-effect experiments. Journal of Pharmacology and Experimental Therapeutics 96(2):99-113. [link].

Examples

```
fit <- gamtable1()
summary(fit)
plot(fit)
```

---

invprobit *Convert Probit Scale to Proportions*

Description

Convert values on the probit scale to their proportions on the 0 to 1 scale.

Usage

```
invprobit(quan)
```

Arguments

quan	A numeric vector of probit quantiles.

Details

Simply calls pnorm(quan).

Value

A numeric vector of proportions the same length as quan.

Examples

```
invprobit(c(-3, -1, 0, 1, 3))
```

keeponly *Eliminate Consecutive Extreme Values*

Description

Generate the index for eliminating values beyond a given maximum number of consecutive extremes allowed.

Usage

```
keeponly(x, extremes = c(0, 100), nconsec = 2)
```

Arguments

x	A numeric vector, with no missing values.
extremes	A numeric vector of length two giving the boundary limits for x, default c(0, 100).
nconsec	An integer scalar, the maximum number of consecutive extreme values allowed, default 2.

Value

A logical vector for selecting all elements of orderedx without exceeding nconsec consecutive extreme values.

Examples

```
vec <- c(0, 0, 0, 4, 4, 4, 100, 100, 100, 100)
vec[keeponly(vec)]
# the original vector need not be ordered
vec <- c(100, 4, 100, 4, 0, 100, 0, 4, 0, 100)
keeponly(vec)
```

LW1949 *Automated Litchfield and Wilcoxon (1949) Evaluation of Dose-Effect Experiments*

Description

LW1949 is an automated approach to Litchfield and Wilcoxon's (1949) evaluation of dose-effect experiments. **LW1949** was first introduced by Adams et al. (*in preparation*).

Details

An example of how to use the functions in **LW1949** is given in this vignette [link]. Use `dataprep` to create a data frame with the results of a dose-effect experiment. Use `fitLW` and `fitprobit` to fit dose-effect relations. And use `plotDE` to plot the results.

U.S. Geological Survey (USGS) Computer Program **LW1949** version 0.0.0.9008. Written by Jean V. Adams, USGS - Great Lakes Science Center glsc.usgs.gov, Ann Arbor, Michigan, USA. Written in programming language R (R Core Team, 2015, www.R-project.org), version 3.2.2 (2015-08-14). Run on a PC with Intel(R) Core(TM) I7-4600m CPU, 2.90 GHz processor, 16.0 GB RAM, and Microsoft Windows 7 Enterprise operating system 2009 Service Pack 1. Source code is available from Jean V. Adams on GitHub, github.com/JVAdams/LW1949, *jvadams (at) usgs (dot) gov*.

Disclaimer: Although this program has been used by the USGS, no warranty, expressed or implied, is made by the USGS or the United States Government as to the accuracy and functioning of the program and related program material nor shall the fact of distribution constitute any such warranty, and no responsibility is assumed by the USGS in connection therewith.

References

Adams, JV, KS Slaght, and MA Boogaard. *In preparation*. An automated approach to Litchfield and Wilcoxon's evaluation of dose-effect experiments.

Litchfield, JT Jr. and F Wilcoxon. 1949. A simplified method of evaluating dose-effect experiments. Journal of Pharmacology and Experimental Therapeutics 96(2):99-113. [link].

`LWP` *User Friendly Evaluation of Dose-Effect Experiments using Litchfield-Wilcoxon and Probit Methods*

Description

User friendly evaluation of dose-effect experiments using automated Litchfield Wilcoxon (1949) and probit estimation methods. This function has been tailored for non-R users with input data set up in a particular way (see Details).

Usage

```
LWP(rawfile = NULL, descrcolz = 1:4, saveplots = TRUE,
  showplots = FALSE, saveresults = TRUE, showresults = TRUE,
  returnresults = FALSE)
```

Arguments

`rawfile`	A character scalar specifying the path of the input data as a csv file. If NULL, default, the user will be prompted to browse to a file using a menu.
`descrcolz`	A numeric vector, the column numbers to use as the description of the test, default 1:4.
`saveplots`	A logical scalar indicating if plots should be saved to a pdf file, default TRUE.
`showplots`	A logical scalar indicating if plots should be shown on screen, default FALSE.
`saveresults`	A logical scalar indicating if results should be saved to a csv file, default TRUE. The csv file is given the same name (plus the suffix "Smry") and is placed in the same directory as the input file.

showresults	A logical scalar indicating if results should be printed to the console, default TRUE. These results include the chi-squared statistic, degrees of freedom, and p-value for the Litchfield Wilcoxon method.
returnresults	A logical scalar indicating if results should be returned by the function, default FALSE.

Details

The input data must include at least these seven columns, with these names in the header row:

- `Test ID` = A character or numeric vector, the unique identifier for each test
- `Source` = A character vector, the source of the chemical
- `Batch` = A character or numeric vector, the batch of the chemical
- `Species` = A character vector, the species tested
- `TFM Conc. (mg/L)` = A numeric vector, the concentration of TFM in mg/L
- `No. Tested` = A numeric vector, the number of animals tested
- `No. Dead` = A numeric vector, the number of animals dead

The input data are expected to be summarized by dose. If duplicate doses are provided, an error will be thrown.

Value

If `returnresults=TRUE`, a data frame with 11 rows per test and 2 more columns than the input data. Three columns from the input data are not included (`TFM Conc. (mg/L)`, `No. Tested`, and `No. Dead`). Five columns are added: the parameter (`param`), the method used (`method`), the estimate (`estimate`), and the 95% confidence interval of the estimate (`lower95ci` and `upper95ci`)

References

Litchfield, JT Jr. and F Wilcoxon. 1949. A simplified method of evaluating dose-effect experiments. Journal of Pharmacology and Experimental Therapeutics 96(2):99-113. [link].

Examples

```
## Not run:
LWP()

## End(Not run)
```

`plotDE` *Plot Dose-Effect Experiments*

Description

Plot dose-effect experiments on the arithmetic scale.

Usage

```
plotDE(DEdata, xlab = "Dose", ylab = "Affected  (%)",
  xlim = range(DEdata$dose, na.rm = TRUE), ylim = c(0, 100), ref = c(0,
  50, 100), ...)
```

Arguments

DEdata	A data frame of dose-effect data (typically, the output from dataprep) containing at least five variables: dose, pfx, log10dose, bitpfx, fxcateg.
xlab	A character scalar, the title for the dose (x) axis, default "Dose".
ylab	A character scalar, the title for the affected (y) axis, default "Affected (%)".
xlim	A numeric vector of length two giving the x coordinate range for dose, default range(DEdata$dose, na.rm=TRUE).
ylim	A numeric vector of length two giving the y coordinate range for affected (%), default c(0.1, 99.9). Observed effects beyond this range will be plotted at the limits of this range using an open symbol.
ref	A numeric vector specifying horizontal reference lines to be added to the plot, default c(0, 50, 100).
...	Additional arguments to plot.

See Also

predLines, plotDELP, predLinesLP

Examples

```
dose <- c(0.0625, 0.125, 0.25, 0.5, 1)
ntested <- rep(8, 5)
nalive <- c(1, 4, 4, 7, 8)
mydat <- dataprep(dose=dose, ntot=ntested, nfx=nalive)
plotDE(mydat)
```

---

plotDELP *Plot Dose-Effect Experiments*

Description

Plot dose-effect experiments on the log10-probit scale.

Usage

```
plotDELP(DEdata, xlab = "Dose", ylab = "Affected  (%)",
  xlim = range(DEdata$dose[DEdata$dose > 0], na.rm = TRUE), ylim = c(0.1,
  99.9), ...)
```

Arguments

DEdata	A data frame of dose-effect data (typically, the output from dataprep) containing at least five variables: dose, pfx, log10dose, bitpfx, fxcateg.
xlab	A character scalar, the title for the dose (x) axis, default "Dose".
ylab	A character scalar, the title for the affected (y) axis, default "Affected (%)".
xlim	A numeric vector of length two giving the x coordinate range for dose, default range(DEdata$dose, na.rm=TRUE).
ylim	A numeric vector of length two giving the y coordinate range for affected (%), default c(0.1, 99.9). Observed effects beyond this range will be plotted at the limits of this range using an open symbol.
...	Additional arguments to plot.

See Also

predLinesLP, plotDE, predLines

Examples

```
dose <- c(0.0625, 0.125, 0.25, 0.5, 1)
ntested <- rep(8, 5)
nalive <- c(1, 4, 4, 7, 8)
mydat <- dataprep(dose=dose, ntot=ntested, nfx=nalive)
plotDELP(mydat)
```

---

predlinear *Determine the Effective Dose from a Linear Regression Fit*

Description

Determine the effective dose for a specified percent effect from the intercept and slope of a linear regression.

Usage

```
predlinear(pct, LWmod, simple = FALSE)
```

Arguments

pct	A numeric vector of effects (in percents) for which to estimate the effective dose(s).
LWmod	If simple=TRUE, a numeric vector of length two giving the intercept and slope of the linear relation between the dose (x, the concentration of the applied chemical on the log10 scale), and the proportion of affected individuals (y, on the probit scale, with 0s converted to 0.1% and 1s converted to 99.9%). If simple=FALSE, a list with the results of fitting a Litchfield and Wilcoxon model to dose-effect data, the output from fitLW.
simple	A logical scalar indicating whether to carry out a simple estimation of effective doses from the intercept and slope (TRUE), or an estimation of effective doses with confidence intervals from the Litchfield and Wilcoxon model (default, FALSE).

Details

Follows methods outlined in Litchfield and Wilcoxon (1949). Specifically, for the 95% confidence intervals, see page 105, and equation 13 in the Appendix (corresponding to Nomograph 4).

Value

A numeric vector the same length as pct giving the estimated dose at the specified percent effect.

If simple=TRUE, a numeric vector the same length as pct with the estimated effective doses. If simple=FALSE, an n*4 numeric matrix with the given effects (pct), the effective doses (ED), and Litchfield and Wilcoxon's (1949) 95% confidence intervals for the effective doses (lower and upper). The number of rows of the matrix, n, is the length of pct.

References

Litchfield, JT Jr. and F Wilcoxon. 1949. A simplified method of evaluating dose-effect experiments. Journal of Pharmacology and Experimental Therapeutics 96(2):99-113. [link].

Examples

```
predlinear(c(16, 50, 84, 99.9), c(1.700875, 2.199559), simple=TRUE)

dose <- c(0.0625, 0.125, 0.25, 0.5, 1)
ntested <- rep(8, 5)
nalive <- c(1, 4, 4, 7, 8)
mydat <- dataprep(dose=dose, ntot=ntested, nfx=nalive)
fLW <- fitLW(mydat)
predlinear(c(25, 50, 99.9), fLW)
```

---

predLines *Add Litchfield and Wilcoxon Predictions to a Plot*

Description

Add predictions from a Litchfield and Wilcoxon model fit to a plot of the results of a dose-effect experiment on the arithmetic scale.

Usage

```
predLines(fit)
```

Arguments

fit	A list of length three containing the result of a Litchfield and Wilcoxon model fit, typically the output from fitLW.

Value

A solid fitted line is added to the plot. Dashed lines are added to the plot representing the **horizontal** 95 for the predicted dose to elicit a given percent affected.

See Also

plotDE, plotDELP, predLinesLP

Examples

```
dose <- c(0.0625, 0.125, 0.25, 0.5, 1)
ntested <- rep(8, 5)
nalive <- c(1, 4, 4, 7, 8)
mydat <- dataprep(dose=dose, ntot=ntested, nfx=nalive)
plotDE(mydat)
myfit <- fitLW(mydat)
predLines(myfit)
```

predLinesLP *Add Litchfield and Wilcoxon Predictions to a Plot*

Description

Add predictions from a Litchfield and Wilcoxon model fit to a plot of the results of a dose-effect experiment on the log10-probit scale.

Usage

```
predLinesLP(fit)
```

Arguments

fit	A list of length three containing the result of a Litchfield and Wilcoxon model fit, typically the output from fitLW.

Value

A solid fitted line is added to the plot. Dashed lines are added to the plot representing the **horizontal** 95 for the predicted dose to elicit a given percent affected.

See Also

plotDELP, plotDE, predLines

Examples

```
dose <- c(0.0625, 0.125, 0.25, 0.5, 1)
ntested <- rep(8, 5)
nalive <- c(1, 4, 4, 7, 8)
mydat <- dataprep(dose=dose, ntot=ntested, nfx=nalive)
plotDELP(mydat)
myfit <- fitLW(mydat)
predLinesLP(myfit)
```

predprobit *Determine the Effective Dose from a Probit Regression Fit*

Description

Determine the effective dose for a specified percent effect from a fitted probit regression model.

Usage

```
predprobit(pct, pfit, alpha = 0.05, logbase = 10)
```

Arguments

pct	A numeric scalar of the effect (as a percent) for which to estimate the effective dose.
pfit	An object of class glm representing a probit regression fit to dose-effect data, typically the result of a call to fitprobit.
alpha	A numeric scalar, the significance level used to generate 100*(1 - alpha)% confidence limits, default 0.05.
logbase	A numeric or logical scalar, the base of the log transformation used for dose in pfit, default 10. Use logbase=FALSE, if the dose was not log transformed.

Value

A numeric vector of length three, the effective dose and the lower and upper 100*(1 - alpha)% confidence limits.

Examples

```
toxdat <- data.frame(
 dose=c(0.05, 0.0625, 0.125, 0.25, 0.5, 1),
 ntot=rep(8, 6),
 nfx = c(0, 1, 4, 4, 6, 8))
myfit <- fitprobit(toxdat)
predprobit(50, myfit)
```

---

prettylog *Pretty Breakpoints on Log Scale*

Description

Compute a sequence of "round" values which cover the range of x on the log scale.

Usage

```
prettylog(x, lead = c(1, 5), extra = 5)
```

Arguments

x	A numeric vector.
lead	An integer vector giving the desired lead digits of pretty values on the log scale, default c(1, 5).
extra	An integer scalar giving the desired number of additional non-log scale values to include, default 5.

Value

A numeric vector of pretty values covering the range of x on the log scale.

References

This is a copy of the prettylog function from the [jvamisc] package.

Examples

```
vals <- rlnorm(100, 6)
summary(vals)
prettylog(vals, 1, 0)
prettylog(vals, 1)
prettylog(vals, c(1, 2, 5))
```

`probit` *Convert Proportions to the Probit Scale*

Description

Convert proportions to the probit scale.

Usage

```
probit(prob)
```

Arguments

`prob`	A numeric vector of proportions.

Details

Simply calls `qnorm(prob)`.

Value

A numeric vector the same length as `prob` with quantiles on the probit scale.

Examples

```
probit(c(0.001, 0.01, 0.1, 0.5, 0.9, 0.99, 0.999))
```

`relPotency` *Relative Potency of Two Toxins*

Description

Estimate of relative potency of two toxins using Litchfield and Wilcoxon's (1949) approach to evaluating dose-effect experiments.

Usage

```
relPotency(ED50nS1, ED50nS2, vec = FALSE)
```

Arguments

ED50nS1	Either the list output from fitLW (vec = FALSE) or a numeric vector of length four (vec = TRUE) with the estimated ED50, fED50, S, and fS from a Litchfield and Wilcoxon fit to dose-effect data for the first toxin.
ED50nS2	Either the list output from fitLW (vec = FALSE) or a numeric vector of length four (vec = TRUE) with the estimated ED50, fED50, S, and fS from a Litchfield and Wilcoxon fit to dose-effect data for the second toxin.
vec	A logical scalar indicating whether the inputs ED50nS1 and ED50nS2 are both numeric vectors (TRUE) or both lists (FALSE, the default).

Details

The ratios reported (both for slope and potency) have the first toxin in the numerator and the second toxin in the denominator, but the test results (both for parallelism and relative potency) are based on the ratios of the larger values over the smaller values.

No relative potency is estimated if the two dose-effect curves differ significantly from parallelism (with 95% confidence).

Value

A list with two elements, SR with three elements:

- r = a numeric vector of length three with the estimated slope ratio with 95% confidence limits,
- f = a numeric scalar with the f of the slope ratio, and
- parallel = a logical scalar indicating whether the two curves differ significantly from parallelism (FALSE).

and PR with one (just difPotency if parallel=FALSE) or three (if parallel=TRUE) elements:

- r = a numeric vector of length three with the estimated potency ratio with 95% confidence limits,
- f = a numeric scalar with the f of the potency ratio, and
- difPotency = a logical scalar indicating whether the two toxins differ significantly in potency (FALSE).

References

Litchfield, JT Jr. and F Wilcoxon. 1949. A simplified method of evaluating dose-effect experiments. Journal of Pharmacology and Experimental Therapeutics 96(2):99-113. [link].

Examples

```
# Example starting from raw tox data
dose <- c(0.0625, 0.125, 0.25, 0.5, 1)
ntested <- rep(8, 5)
nalive1 <- c(1, 4, 4, 7, 8)
mydat1 <- dataprep(dose=dose, ntot=ntested, nfx=nalive1)
nalive2 <- c(0, 1, 2, 6, 6)
mydat2 <- dataprep(dose=dose, ntot=ntested, nfx=nalive2)
fit1 <- fitLW(mydat1)
fit2 <- fitLW(mydat2)
relPotency(fit1, fit2)
```

```
# Example from Litchfield and Wilcoxon (1949)
# comparing Tagathen and Pyribenzamine
relPotency(c(0.18, 1.72, 2.20, 1.60), c(0.60, 1.60, 2.34, 1.57), vec=TRUE)

# Example in which curves differ significantly from parallelism.
relPotency(c(0.18, 1.72, 2.20, 1.60), c(0.60, 1.60, 4.34, 1.57), vec=TRUE)
```

Index