Package 'ecotox'

July 22, 2025

Type Package

Title Analysis of Ecotoxicology

Version 1.4.4

Description A simple approach to using a probit or logit analysis to calculate
lethal concentration (LC) or time (LT) and the appropriate fiducial
confidence limits desired for selected LC or LT for
ecotoxicology studies (Finney 1971; Wheeler et al. 2006;
Robertson et al. 2007). The simplicity of 'ecotox' comes from the
syntax it implies within its functions which are similar to functions
like glm() and lm(). In addition to the simplicity of the syntax,
a comprehensive data frame is produced which gives the user a
predicted LC or LT value for the desired level and a suite of important
parameters such as fiducial confidence limits and slope.
Finney, D.J. (1971, ISBN: 052108041X);
Wheeler, M.W., Park, R.M., and Bailer, A.J. (2006) <doi:10.1897/05-320R.1>;
Robertson, J.L., Savin, N.E., Russell, R.M., and Preisler, H.K.
(2007, ISBN: 0849323312).

Depends R (>= 3.3.0)

Imports stats, tibble

Suggests covr, data.table, dplyr, ggplot2, Hmisc, openxlsx, readr,
testthat, tidyr

License GPL-3 | file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.1.2.9000

NeedsCompilation no

Author Benjamin L Hlina [aut, cre]

Maintainer Benjamin L Hlina <benjamin.hlina@gmail.com>

Repository CRAN

Date/Publication 2021-10-27 16:20:02 UTC

Contents

---

lamprey_time *Lethal time data for seasonal exposure to TFM for larval sea lamprey*

Description

This dataset contains the amount of responding/dead lamprey from 1 to 12 h of exposure to the piscicide, 3-trifluoromethyl-4-nitrophenol (TFM), for the calculated LC50s at each month for the 2011 seasonal larval sea lamprey TFM toxicity data.

Usage

```
lamprey_time
```

Format

A tibble containing 44 rows and 7 columns.

nominal.dose nominal TFM dose(mg/L)

dose averaged measured TFM dose (mg/L) over 12 h of exposures

month month in which toxicity test occurred

hour duration of exposure in hours

response number of lamprey that responded over 12 h exposure to TFM

survive number of lamprey that survived over 12 h exposure to TFM

total total number of lamprey that were exposed to TFM over 12 h

Details

For each month, new lamprey were collected from the deer creek, MI, and were randomly assigned to a tank for exposure to the piscicide, 3-trifluoromethyl-4-nitrophenol (TFM). The LC50 at each month was then calculated and the amount of responding/dead lamprey from 1 to 12 h was recorded.

Source

Hlina, B.L., Birceanu, O., Robinson, C.S., Dhiyebi, H., Wilkie, M.P. *In Reivew*. Seasonal Variation in the Sensitivity of Invasive Sea Lampreys to the Lampricide TFM: Importance of Energy Reserves and Temperature. North American Journal of Fisheries Management

lamprey_tox	*Lethal concentration data for seasonal exposure to TFM for larval sea lamprey*

Description

This dataset contains 2011 seasonal larval sea lamprey exposed to the piscicide, 3-trifluoromethyl-4-nitrophenol (TFM) toxicity data.

Usage

```
lamprey_tox
```

Format

A tibble containing 64 rows and 7 columns.

nominal_dose nominal TFM dose (mg/L)

tank tank assignment

month month in which toxicity test occurred

dose averaged measured TFM dose (mg/L) over 12 h of exposure

response number of lamprey that responded over 12 h exposure to TFM

survive number of lamprey that survived over 12 h exposure to TFM

total total number of lamprey that were exposed to TFM over 12 h

Details

For each month, new lamprey were collected from the Deer Creek, MI, and were randomly assigned to a tank for exposure to varying doses of the piscicide, 3-trifluoromethyl-4-nitrophenol (TFM).

Source

Hlina, B.L., Birceanu, O., Robinson, C.S., Dhiyebi, H., Wilkie, M.P. *In Reivew*. Seasonal Variation in the Sensitivity of Invasive Sea Lampreys to the Lampricide TFM: Importance of Energy Reserves and Temperature. North American Journal of Fisheries Management

---

LC_logit   *Lethal Concentration Logit*

Description

Calculates lethal concentration (LC) and its fiducial confidence limits (CL) using a logit analysis according to Finney 1971, Wheeler et al. 2006, and Robertson et al. 2007.

Usage

```
LC_logit(formula, data, p = NULL, weights = NULL,
         subset = NULL, log_base = NULL,
         log_x = TRUE, het_sig = NULL,
         conf_level = NULL, conf_type = NULL,
         long_output = TRUE)
```

Arguments

formula	an object of class `formula` or one that can be coerced to that class): a symbolic description of the model to be fitted. The details of model specification are given under Details.
data	an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment(formula), typically the environment from which `LC_logit` is called.
p	Lethal Concentration (LC) values for given p, example will return a LC50 value if p equals 50. If more than one LC value wanted specify by creating a vector. LC values can be calculated down to the 1e-16 of a percentage (e.g. LC99.99). However, the tibble produced can round to nearest whole number.
weights	vector of 'prior weights' to be used in the fitting process. Only needs to be supplied if you are taking the response / total for your response variable within the formula call of `LC_probit`. Otherwise if you use cbind(response, non-response) method you do not need to supply weights. If you do the model will be incorrect. If you don't supply weights there is a warning that will help you to make sure you are using one method or the other.See the following StackExchange post about differences cbind() function in R for a logistic regression.
subset	allows for the data to be subseted if desired. Default set to `NULL`.
log_base	default is `10` and will be used to calculate results using the anti of `log10()` given that the x variable has been `log10` transformed. If `FALSE` results will not be back transformed.
log_x	default is `TRUE` and will calculate results using the antilog of determined by `log_base` given that the x variable has been `log()` transformed. If `FALSE` results will not be back transformed.
het_sig	significance level from person's chi square goodness-of-fit test that is used to decide if a heterogeneity factor is used. `NULL` is set to 0.15.

conf_level	adjust confidence level as necessary or NULL set at 0.95.
conf_type	default is "fl" which will calculate fudicial confidence limits per Finney 1971. If set to "dm" the delta method will be used instead.
long_output	default is TRUE which will return a tibble with all 17 variables. If FALSE the tibble returned will consist of the p level, n, the predicted LC for given p level, lower and upper confidence limits.

Value

Returns a tibble with predicted LC for given p level, lower CL (LCL), upper CL (UCL), Pearson's chi square goodness-of-fit test (pgof), slope, intercept, slope and intercept p values and standard error, and LC variance.

References

Finney, D.J., 1971. Probit Analysis, Cambridge University Press, Cambridge, England, ISBN: 052108041X

Wheeler, M.W., Park, R.M., and Bailey, A.J., 2006. Comparing median lethal concentration values using confidence interval overlap or ratio tests, Environ. Toxic. Chem. 25(5), 1441-1444.10.1897/05-320R.1

Robertson, J.L., Savin, N.E., Russell, R.M. and Preisler, H.K., 2007. Bioassays with arthropods. CRC press. ISBN: 9780849323317

Examples

```
head(lamprey_tox)

# within the dataframe used, control dose, unless produced a value
# during experimentation, are removed from the dataframe,
# as glm cannot handle values of infinite. Other statistical programs
# make note of the control dose but do not include within analysis


# calculate LC50 and LC99 for May

m <- LC_logit((response / total) ~ log10(dose), p = c(50, 99),
        weights = total,
        data = lamprey_tox[lamprey_tox$nominal_dose != 0, ],
        subset = c(month == "May"))


# OR

m1 <- LC_logit(cbind(response, survive) ~ log10(dose), p = c(50, 99),
                data = lamprey_tox[lamprey_tox$nominal_dose != 0, ],
                subset = c(month == "May"))


# view calculated LC50 and LC99 for seasonal toxicity of a pisicide,
# to lamprey in 2011
```

```
m

# they are the same

m1

# dose-response curve can be plotted using 'ggplot2'
# Uncomment the below lines to run create plots

# library(ggplot2)

# lc_may <- subset(lamprey_tox, month %in% c("May"))

# p1 <- ggplot(data = lc_may[lc_may$nominal_dose != 0, ],
#                aes(x = log10(dose), y = (response / total))) +
# geom_point() +
# geom_smooth(method = "glm",
#                method.args = list(family = binomial(link = "logit")),
#                aes(weight = total), colour = "#FF0000", se = TRUE)

# p1

# calculate LC50s and LC99s for multiple toxicity tests, June, August, and September

j <- LC_logit((response / total) ~ log10(dose), p = c(50, 99),
        weights = total,
        data = lamprey_tox[lamprey_tox$nominal_dose != 0, ],
        subset = c(month == "June"))

a <- LC_logit((response / total) ~ log10(dose), p = c(50, 99),
        weights = total,
        data = lamprey_tox[lamprey_tox$nominal_dose != 0, ],
        subset = c(month == "August"))

s <- LC_logit((response / total) ~ log10(dose), p = c(50, 99),
        weights = total,
        data = lamprey_tox[lamprey_tox$nominal_dose != 0, ],
        subset = c(month == "September"))

# group results together in a dataframe to plot with 'ggplot2'

results <- rbind(m[, c(1, 3:8, 11)], j[,c(1, 3:8, 11)],
                 a[, c(1, 3:8, 11)], s[, c(1, 3:8, 11)])
results$month <- factor(c(rep("May", 2), rep("June", 2),
                        rep("August", 2), rep("September", 2)),
                      levels = c("May", "June", "August", "September"))


# p2 <- ggplot(data = results, aes(x = month, y = dose,
#                                  group = factor(p), fill = factor(p))) +
#   geom_col(position = position_dodge(width = 0.9), colour = "#000000") +
#   geom_errorbar(aes(ymin = LCL, ymax = UCL),
```

```
#                 size = 0.4, width = 0.06,
#                 position = position_dodge(width = 0.9))

# p2
```

LC_probit   *Lethal Concentration Probit*

Description

Calculates lethal concentration (LC) and its fiducial confidence limits (CL) using a probit analysis according to Finney 1971, Wheeler et al. 2006, and Robertson et al. 2007.

Usage

```
LC_probit(formula, data, p = NULL, weights = NULL,
          subset = NULL, log_base = NULL, log_x = TRUE,
          het_sig = NULL, conf_level = NULL, conf_type = NULL,
          long_output = TRUE)
```

Arguments

formula	an object of class `formula` or one that can be coerced to that class): a symbolic description of the model to be fitted. The details of model specification are given under Details.
data	an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment(formula), typically the environment from which `LC_probit` is called.
p	Lethal Concentration (LC) values for given p, example will return a LC50 value if p equals 50. If more than one LC value wanted specify by creating a vector. LC values can be calculated down to the 1e-16 of a percentage (e.g. LC99.99). However, the tibble produced can round to nearest whole number.
weights	vector of 'prior weights' to be used in the fitting process. Only needs to be supplied if you are taking the response / total for your response variable within the formula call of `LC_probit`. Otherwise if you use cbind(response, non-response) method you do not need to supply weights. If you do the model will be incorrect. If you don't supply weights there is a warning that will help you to make sure you are using one method or the other. See the following StackExchanges post about differences [cbind() function in R for a logistic regression](#) and [input format for binomial glm](#).
subset	allows for the data to be subseted if desired. Default set to NULL.
log_base	default is `10` and will be used to calculate results using the anti of `log10()` given that the x variable has been `log10` transformed. If `FALSE` results will not be back transformed.

log_x	default is TRUE and will calculate results using the antilog of determined by log_base given that the x variable has been log() transformed. If FALSE results will not be back transformed.
het_sig	significance level from person's chi square goodness-of-fit test (pgof) that is used to decide if a heterogeneity factor is used. NULL is set to 0.15.
conf_level	adjust confidence level as necessary or NULL set at 0.95.
conf_type	default is "fl" which will calculate fudicial confidence limits per Finney 1971. If set to "dm" the delta method will be used instead.
long_output	default is TRUE which will return a tibble with all 17 variables. If FALSE the tibble returned will consist of the p level, n, the predicted LC for given p level, lower and upper confidence limits.

Value

Returns a tibble with predicted LC for given p level, lower CL (LCL), upper CL (UCL), Pearson's chi square goodness-of-fit test (pgof), slope, intercept, slope and intercept p values and standard error, and LC variance.

References

Finney, D.J., 1971. Probit Analysis, Cambridge University Press, Cambridge, England, ISBN: 052108041X

Wheeler, M.W., Park, R.M., and Bailey, A.J., 2006. Comparing median lethal concentration values using confidence interval overlap or ratio tests, Environ. Toxic. Chem. 25(5), 1441-1444.10.1897/05-320R.1

Robertson, J.L., Savin, N.E., Russell, R.M. and Preisler, H.K., 2007. Bioassays with arthropods. CRC press. ISBN: 9780849323317

Examples

```
head(lamprey_tox)

# within the dataframe used, control dose, unless produced a value
# during experimentation, are removed from the dataframe,
# as glm cannot handle values of infinite. Other statistical programs
# make note of the control dose but do not include within analysis

# calculate LC50 and LC99

m <- LC_probit((response / total) ~ log10(dose), p = c(50, 99),
               weights = total,
               data = lamprey_tox[lamprey_tox$nominal_dose != 0, ],
               subset = c(month == "May"))
# OR

m1 <- LC_probit(cbind(response, survive) ~ log10(dose), p = c(50, 99),
               data = lamprey_tox[lamprey_tox$nominal_dose != 0, ],
               subset = c(month == "May"))
```

```
# view calculated LC50 and LC99 for seasonal toxicity of a pisicide,
# to lamprey in 2011

m

# these are the same

m1

# dose-response curve can be plotted using 'ggplot2'

# library(ggplot2)

# lc_may <- subset(lamprey_tox, month %in% c("May"))

# p1 <- ggplot(data = lc_may[lc_may$nominal_dose != 0, ],
#               aes(x = log10(dose), y = (response / total))) +
#   geom_point() +
#   geom_smooth(method = "glm",
#               method.args = list(family = binomial(link = "probit")),
#               aes(weight = total), colour = "#FF0000", se = TRUE)

# p1

# calculate LC50s and LC99s for multiple toxicity tests, June, August, and September

j <- LC_probit((response / total) ~ log10(dose), p = c(50, 99),
        weights = total,
        data = lamprey_tox[lamprey_tox$nominal_dose != 0, ],
        subset = c(month == "June"))

a <- LC_probit((response / total) ~ log10(dose), p = c(50, 99),
        weights = total,
        data = lamprey_tox[lamprey_tox$nominal_dose != 0, ],
        subset = c(month == "August"))

s <- LC_probit((response / total) ~ log10(dose), p = c(50, 99),
        weights = total,
        data = lamprey_tox[lamprey_tox$nominal_dose != 0, ],
        subset = c(month == "September"))

# group results together in a dataframe to plot with 'ggplot2'

results <- rbind(m[, c(1, 3:8, 11)], j[,c(1, 3:8, 11)],
                 a[, c(1, 3:8, 11)], s[, c(1, 3:8, 11)])
results$month <- factor(c(rep("May", 2), rep("June", 2),
                          rep("August", 2), rep("September", 2)),
                        levels = c("May", "June", "August", "September"))

# p2 <- ggplot(data = results, aes(x = month, y = dose,
#                                  group = factor(p), fill = factor(p))) +
#   geom_col(position = position_dodge(width = 0.9), colour = "#000000") +
#   geom_errorbar(aes(ymin = LCL, ymax = UCL),
```

```
#                    size = 0.4, width = 0.06,
#                    position = position_dodge(width = 0.9))

# p2
```

LT_logit   *Lethal Time Logit*

Description

Calculates lethal time (LT) and its fiducial confidence limits (CL) using a logit analysis according to Finney 1971, Wheeler et al. 2006, and Robertson et al. 2007.

Usage

```
LT_logit(formula, data, p = NULL, weights = NULL,
         subset = NULL, log_base = NULL, log_x = TRUE, het_sig = NULL,
         conf_level = NULL, long_output = TRUE)
```

Arguments

formula	an object of class `formula` or one that can be coerced to that class: a symbolic description of the model to be fitted.
data	an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment(formula), typically the environment from which `LT_logit` is called.
p	Lethal time (LT) values for given p, example will return a LT50 value if p equals 50. If more than one LT value wanted specify by creating a vector. LT values can be calculated down to the 1e-16 of a percentage (e.g. LT99.99).However, the tibble produced can and will round to nearest whole number.
weights	vector of 'prior weights' to be used in the fitting process. Only needs to be supplied if you are taking the response / total for your response variable within the formula call of `LC_probit`. Otherwise if you use cbind(response, non-response) method you do not need to supply weights. If you do the model will be incorrect. If you don't supply weights there is a warning that will help you to make sure you are using one method or the other. See the following StackExchange post about differences [cbind() function in R for a logistic regression]().
subset	allows for the data to be subseted if desired. Default set to NULL.
log_base	default is `10` and will be used to calculate results using the anti of `log10()` given that the x variable has been `log10` transformed. If `FALSE` results will not be back transformed.
log_x	default is `TRUE` and will calculate results using the antilog of determined by `log_base` given that the x variable has been `log()` transformed. If `FALSE` results will not be back transformed.

het_sig	significance level from person's chi square goodness-of-fit test that is used to decide if a heterogeneity factor is used. `NULL` is set to 0.15.
conf_level	Adjust confidence level as necessary or `NULL` set at 0.95.
long_output	default is `TRUE` which will return a tibble with all 17 variables. If `FALSE` the tibble returned will consist of the p level, n, the predicted LC for given p level, lower and upper confidence limits.

Value

Returns a tibble with predicted LT for given p level, lower CL (LCL), upper CL (UCL), LCL, Pearson's chi square goodness-of-fit test (pgof), slope, intercept, slope and intercept p values and standard error, and LT variance.

References

Finney, D.J., 1971. Probit Analysis, Cambridge University Press, Cambridge, England, ISBN: 052108041X

Wheeler, M.W., Park, R.M., and Bailey, A.J., 2006. Comparing median lethal concentration values using confidence interval overlap or ratio tests, Environ. Toxic. Chem. 25(5), 1441-1444.10.1897/05-320R.1

Robertson, J.L., Savin, N.E., Russell, R.M. and Preisler, H.K., 2007. Bioassays with arthropods. CRC press. ISBN: 9780849323317

Examples

```
head(lamprey_time)

results <- LT_logit((response / total) ~ log10(hour),
p = c(50, 99),
weights = total,
data = lamprey_time,
subset = c(month == "May"))

# view calculated LT50 and LT99 for seasonal
# toxicity of a piscicide, 3-trifluoromethyl-4-nitrophenol, to lamprey in 2011

results

# dose-response curve can be plotted using 'ggplot2'
```

LT_probit	*Lethal Time Probit*

Description

Calculates lethal time (LT) and its fiducial confidence limits (CL) using a probit analysis according to Finney 1971, Wheeler et al. 2006, and Robertson et al. 2007.

Usage

```
LT_probit(
  formula,
  data,
  p = NULL,
  weights = NULL,
  subset = NULL,
  log_base = NULL,
  log_x = TRUE,
  het_sig = NULL,
  conf_level = NULL,
  conf_type = NULL,
  long_output = TRUE
)
```

Arguments

formula	an object of class `formula` or one that can be coerced to that class: a symbolic description of the model to be fitted.
data	an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment(formula), typically the environment from which `LT_probit` is called.
p	Lethal time (LT) values for given p, example will return a LT50 value if p equals 50. If more than one LT value wanted specify by creating a vector. LT values can be calculated down to the 1e-16 of a percentage (e.g. LT99.99). However, the tibble produced can and will round to nearest whole number.
weights	vector of 'prior weights' to be used in the fitting process. Only needs to be supplied if you are taking the response / total for your response variable within the formula call of `LC_probit`. Otherwise if you use cbind(response, non-response) method you do not need to supply weights. If you do the model will be incorrect. If you don't supply weights there is a warning that will help you to make sure you are using one method or the other. See the following StackExchange post about differences cbind() function in R for a logistic regression.
subset	allows for the data to be subseted if desired. Default set to `NULL`.
log_base	default is `10` and will be used to calculate results using the anti of `log10()` given that the x variable has been `log10` transformed. If `FALSE` results will not be back transformed.
log_x	default is `TRUE` and will calculate results using the antilog of determined by `log_base` given that the x variable has been `log()` transformed. If `FALSE` results will not be back transformed.
het_sig	significance level from person's chi square goodness-of-fit test that is used to decide if a heterogeneity factor is used. `NULL` is set to 0.15.
conf_level	Adjust confidence level as necessary or `NULL` set at 0.95.
conf_type	default is `"fl"` which will calculate fudicial confidence limits per Finney 1971. If set to `"dm"` the delta method will be used instead.

long_output   default is `TRUE` which will return a tibble with all 17 variables. If `FALSE` the tibble returned will consist of the p level, n, the predicted LC for given p level, lower and upper confidence limits.

Value

Returns a tibble with predicted LT for given p level, lower CL (LCL), upper CL (UCL), LCL, Pearson's chi square goodness-of-fit test (pgof), slope, intercept, slope and intercept p values and standard error, and LT variance.

References

Finney, D.J., 1971. Probit Analysis, Cambridge University Press, Cambridge, England, ISBN: 052108041X

Wheeler, M.W., Park, R.M., and Bailey, A.J., 2006. Comparing median lethal concentration values using confidence interval overlap or ratio tests, Environ. Toxic. Chem. 25(5), 1441-1444.10.1897/05-320R.1

Robertson, J.L., Savin, N.E., Russell, R.M. and Preisler, H.K., 2007. Bioassays with arthropods. CRC press. ISBN: 9780849323317

Examples

```
head(lamprey_time)

results <- LT_probit((response / total) ~ log10(hour),
p = c(50, 99),
weights = total,
data = lamprey_time,
subset = c(month == "May"))

# view calculated LT50 and LT99 for seasonal
# toxicity of a piscicide, 3-trifluoromethyl-4-nitrophenol, to lamprey in 2011

results

# dose-response curve can be plotted using 'ggplot2'
```

---

ratio_test   *Ratio test*

Description

Calculates a ratio test to compare two LC or LT values from two separate probit or logit models. This function is based on the ratio test developed in Wheeler et al. 2006. 10.1897/05-320R.1 which has been suggested as a replacement to the common method of comparing confidence intervals to determine differences.

Usage

```
ratio_test(model_1, model_2, percentage = NULL,
           type = NULL, compare = NULL, log_base = NULL, log_x = TRUE,
           obj_type = NULL, conf_type = NULL)
```

Arguments

model_1   first model used in the ratio test. Should be an object of either a probit or logit
               model created using the glm() function. See example.

model_2   second model used in the ratio test. Should be an object of either a probit or
               logit model created using the glm() function. See example.

percentage   either a single value or a vector for given LC or LT percentage desired to com-
               pare. Percentage is the same value used for the argument p in all LC_ and LT_
               functions. For example, 50 will return and compare LC50 values for the two
               models. If more than one LC value is desired specify by creating a vector. LC
               values can be calculated down to the 1e-16 of a percentage (e.g. LC99.99).
               However, the tibble produced can and will round to nearest whole number.

type   Link type needs to be specified to either "probit" which is default and will
               return and used in calculations for a probit model for the desired LCs or LTs.
               If specified to "logit" then ratio_test will return and calculate using a logit
               model for the desired LCs or LTs.

compare   Supply a character string to be used in the output letting the user know what
               models the LCs or LTs are being compared. Default output is "Model 1 - Model
               2". See example.

log_base   default is 10 and will be used to calculate results using the anti of log10() given
               that the x variable has been log10 transformed. If FALSE results will not be back
               transformed.

log_x   default is TRUE and will calculate results using the antilog of determined by
               log_base given that the x variable has been log() transformed. If FALSE results
               will not be back transformed.

obj_type   default is "list" which requires both model_1 and model_2 arguments to be
               model objects, in the form of a list, from glm() functions. Alternatively "df"
               can be used which will require both model_1 and model_2 arguments to be
               data.frame objects created when running either LC_ or LT_ functions.

conf_type   default is "fl" which if "df" is supplied will correct covariance values if h is
               above 1 as fudicial confidence limits use a heterogeneity factor, h, to correct
               variances when chi-square p value is less than 0.15. conf_type can also be
               "dm", delta method, which doesn't use a heterogeneity correction factor there-
               fore covariance will not be uncorrected. This argument is only needed if you are
               using the dataframe objects from LC_ or LT_ functions and have used the delta
               method in that analysis.

Value

A tibble with percentage for the LC or LT value desired for the above percentage argument,
dose_1 and dose_2 displayed calculated backtransformed or untransformed doses for the desired

LC or LT values. Standard Error (se), Z test statistic (test_stat) and p_value determined using Z test statistic as determined using formulas in Wheeler et al. 2006. .

References

Wheeler, M.W., Park, R.M., and Bailey, A.J., 2006. Comparing median lethal concentration values using confidence interval overlap or ratio tests, Environ. Toxic. Chem. 25(5), 1441-1444. 10.1897/05-320R.1

Examples

```
# view lamprey_tox data

head(lamprey_tox)

# using glm() to detemine LC values using probit model for May and June

m <- glm((response / total) ~ log10(dose),
         data = lamprey_tox[lamprey_tox$nominal_dose != 0, ],
         subset = c(month == "May"),
         weights = total,
         family = binomial(link = "probit"))


j <- glm((response / total) ~ log10(dose),
         data = lamprey_tox[lamprey_tox$nominal_dose != 0, ],
         subset = c(month == "June"),
         weights = total,
         family = binomial(link = "probit"))

# now that both May and June models have been made. use ratio_test to
# compare LC50 values or whatever LC values of interest.

ratios <- ratio_test(model_1 = m, model_2 = j, percentage = 50,
compare = "May - June")

# view ratio test results

ratios

# you can also use LC_probit to create the models and use ratio test

m_1 <- LC_probit((response / total) ~ log10(dose), p = c(50, 99),
weights = total,
data = lamprey_tox[lamprey_tox$nominal_dose != 0, ],
subset = c(month == "May"))



j_1 <- LC_probit((response / total) ~ log10(dose), p = c(50, 99),
weights = total,
data = lamprey_tox[lamprey_tox$nominal_dose != 0, ],
```

```
subset = c(month == "June"))


ratios_2 <- ratio_test(model_1 = m_1, model_2 = j_1, percentage = 50,
compare = "May - June", obj_type = "df")

ratios_2
```

Index