

# Package ‘BioRssay’

August 9, 2022

**Type** Package

**Title** Analyze Bioassays and Probit Graphs

**Version** 1.1.0

**Maintainer** Piyal Karunaratne <piyalkarumail@yahoo.com>

**Description** A robust framework for analyzing mortality data from bioassays for one or several strains/lines/populations.

**License** AGPL (>= 3)

**Imports** colorspace

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.2

**Depends** R (>= 3.5)

**Suggests** markdown, rmarkdown, knitr, testthat (>= 2.0.0), covr

**VignetteBuilder** knitr

**URL** <https://milesilab.github.io/BioRssay/>,  
<https://cran.r-project.org/package=BioRssay>

**BugReports** <https://github.com/milesilab/BioRssay/issues/>

**Config/testthat/edition** 2

**NeedsCompilation** no

**Author** Piyal Karunaratne [aut, cre] (<<https://orcid.org/0000-0002-1934-145X>>),  
Pascal Milesi [aut] (<<https://orcid.org/0000-0001-8580-4291>>),  
Pierrick Labbé [aut] (<<https://orcid.org/0000-0003-0806-1919>>)

**Repository** CRAN

**Date/Publication** 2022-08-09 14:40:10 UTC

## R topics documented:

bioassay . . . . .	2
model.signif . . . . .	2
mort.plot . . . . .	3
probit.trans . . . . .	4
resist.ratio . . . . .	5

<b>Index</b>	<b>8</b>
--------------	----------

---

bioassay	<i>Example bioassay data set</i>
----------	----------------------------------

---

### Description

Example bioassay data set

### Usage

```
data(bioassay)
```

### Format

An object of class `list` of length 2.

### Examples

```
data(bioassay)
head(bioassay$assay1)
```

---

model.signif	<i>Test the significance of dose-mortality response differences</i>
--------------	---

---

### Description

This function is used when comparing at least two strains. It tests whether the mortality-dose regressions are similar for different strains, using a likelihood ratio test (LRT). If there are more than two strains, it also computes pairwise tests, using sequential Bonferroni correction (Hommel, 1988) to account for multiple testing.

### Usage

```
model.signif(data)
```

### Arguments

`data` a data frame of probit transformed mortality data using the function `probit.trans`

## Details

A global LRT test assesses a strain's effect, by comparing two models, one with and one without this effect (i.e. comparing a model with several strains to a model where all the data originate from a single strain). If there are more than two strains, pairwise tests are computed, and p-values of significance are assessed using sequential Bonferroni correction (Hommel, 1988) to account for multiple testing.

Warning: We strongly encourage users to not use this function when the dose-mortality response for at least one strain significantly deviates from linearity (see `resist.ratio()` function for more details): in such cases the test cannot be interpreted.

## Value

a list with model outputs: a chi-square test if there are only two strains or if there are more than two strains, first an overall model assessment (i.e. one strain vs. all) and given overall model is significant, then a bonferroni test of significance from a pairwise model comparison.

## Author(s)

Pascal Milesi, Piyal Karunaratne, Pierrick Labbé

## Examples

```
data(bioassay)
transd<-probit.trans(bioassay$assay2)
data<-transd$str.data
model.signif(data)
```

---

mort.plot

*Plot dose-mortality response for each strain*

---

## Description

This function plots the probit-transformed mortalities (`probit.trans()` function) as a function of the log10 of the dose, the regressions predicted by the `resist.ratio()` function, with or without confidence levels, if the dose-mortality responses are linear (option).

## Usage

```
mort.plot(
  data,
  strains = NULL,
  plot.conf = TRUE,
  conf.level = 0.95,
  LD.value = c(25, 50, 95),
  test.validity = TRUE,
  legend.par = c("bottomleft"),
  ...
)
```

**Arguments**

data	a data frame of probit transformed mortality data using the function <code>probit.trans()</code>
strains	character. list of test strains to be plotted. If not provided, the function will plot all the strains in the data set.
plot.conf	logical. Whether to plot the confidence intervals for each strain, default TRUE
conf.level	numerical. The confidence interval to be plotted
LD.value	numerical. Level of lethal dose to be tested. default=c(25,50,95)
test.validity	logical. When TRUE (default), if a strain mortality-dose response fails the chi-square test for linearity in the <code>resist.ratio()</code> function, no regression will be plotted, only the observed data.
legend.par	multi-type. Arguments to be passed to the legend as in <a href="#">legend</a> . default position <code>bottomLeft</code> . If no legend desired use FALSE. Note: if <code>pch</code> , <code>lty</code> , and <code>col</code> are passed to the plot, they don't need to be passed to <code>legend()</code>
...	parameters to be passed on to graphics for the plot (e.g. <code>col</code> , <code>pch</code> )

**Value**

A plot of dose-mortality responses for bioassays

**Author(s)**

Piyal Karunaratne, Pascal Milesi, Pierrick Labbé

**Examples**

```
data(bioassay)
transd<-probit.trans(bioassay$assay2)
data<-transd$str.data
strains<-levels(data$strain)
mort.plot(data,strains)
```

---

`probit.trans`

*Probit-transform the data and apply Abbott's correction*

---

**Description**

This function applies probit transformation to the data, after applying Abbott's correction (see reference) when control groups (e.g. unexposed susceptible strain) show non-negligible mortality.

**Usage**

```
probit.trans(dataf, conf = 0.05)
```

**Arguments**

`dataf` a data frame of mortality data containing four mandatory columns "strain", "dose", "total", "dead" (not necessarily in that order).

`conf` numerical. Threshold for the mortality in the controls above which the correction should be applied (default=0.05)

**Value**

Returns a list. `convrg`: with correction values and convergence (NULL if mortality in the controls is below `conf`.), `tr.data`: transformed data

**Author(s)**

Pascal Milesi, Piyal Karunaratne, Pierrick Labbé

**References**

Abbott, WS (1925). A method of computing the effectiveness of an insecticide. *J. Econ. Entomol.*;18:265-267.

**Examples**

```
data(bioassay)
transd<-probit.trans(bioassay)
```

---

<code>resist.ratio</code>	<i>Calculate lethal dosage, resistance ratios, and regression coefficients and tests for linearity</i>
---------------------------	--

---

**Description**

Using a generalized linear model (GLM, logit link function), this function computes the lethal doses for 25 of the population (LD25, LD50 and LD95, resp.), and their confidence intervals (LDmax and LDmin, 0.95 by default). See details for more info.

**Usage**

```
resist.ratio(
  data,
  conf.level = 0.95,
  LD.value = c(25, 50, 95),
  ref.strain = NULL,
  plot = FALSE,
  plot.conf = TRUE,
  test.validity = TRUE,
  legend.par = c("bottomright"),
  ...
)
```

**Arguments**

<code>data</code>	a data frame of probit-transformed mortality data using the function <code>probit.trans()</code>
<code>conf.level</code>	numerical. level for confidence intervals to be applied to the models (default 0.95)
<code>LD.value</code>	numerical. Level of lethal dose to be tested. default=c(25,50,95)
<code>ref.strain</code>	character. name of the reference strain if present (see details)
<code>plot</code>	logical. Whether to draw the plot. Default FALSE
<code>plot.conf</code>	logical. If plot=TRUE, whether to plot the 95 percent confidence intervals. Default TRUE
<code>test.validity</code>	logical. If plot=TRUE (default), the regression for a strain that failed the linearity test is not plotted
<code>legend.par</code>	arguments to be passed on to <code>legend()</code> as in <code>mort.plot()</code>
<code>...</code>	parameters to be passed on to graphics for the plot (e.g. <code>col</code> , <code>pch</code> )

**Details**

If a name is provided in `ref.strain=`, it will be used as the reference to compute the resistance ratios (RR). Alternatively, the function will look for a strain with the suffix "-ref" in the dataset. If this returns NULL, the strain with the lowest LD50 will be considered as reference.

In addition to LD values, the function in a nutshell uses a script modified from Johnson et al (2013), which allows taking the *g* factor into account ("With almost all good sets of data, *g* will be substantially smaller than 1.0 and seldom greater than 0.4." Finney, 1971) and the heterogeneity (*h*) of the data (Finney, 1971) to calculate the confidence intervals (i.e. a larger heterogeneity will increase the confidence intervals). It also computes the corresponding resistance ratios (RR), i.e. the ratios between a given strain and the strain with the lower LD50 and LD95, respectively for RR50 and RR95 (usually, it is the susceptible reference strain), with their 95 Robertson and Preisler (1992). Finally, it also computes the coefficients (slope and intercept, with their standard error) of the linear regressions) and tests for the linearity of the dose-mortality response using a chi-square test (Chi(p)) between the observed dead numbers (*data*) and the dead numbers predicted by the regression (the test is significant if the data is not linear, e.g. mixed populations).

**Value**

Returns a data frame with the various estimates mentioned above. If `plot=TRUE`, plots the mortality on a probit-transformed scale against the  $\log_{10}$  doses.

**Author(s)**

Pascal Milesi, Piyal Karunaratne, Pierrick Labbé

**References**

- Finney DJ (1971). Probitanalysis. Cambridge:Cambridge University Press. 350p.
- Hommel G (1988). A stage wise rejective multiple test procedure based on a modified Bonferroni test. *Biometrika* 75, 383-6.

Johnson RM, Dahlgren L, Siegfried BD, Ellis MD (2013). Acaricide, fungicide and drug interactions in honeybees (*Apis mellifera*). PLoS ONE 8(1): e54092.

Robertson, J. L., and H.K. Preisler. 1992. Pesticide bioassays with arthropods. CRC, Boca Raton, FL.

### **Examples**

```
data(bioassay)
transd<-probit.trans(bioassay$assay2)
data<-transd$tr.data
resist.ratio(data,plot=TRUE)
```

# Index

\* **datasets**

bioassay, [2](#)

bioassay, [2](#)

legend, [4](#)

model.signif, [2](#)

mort.plot, [3](#)

probit.trans, [4](#)

resist.ratio, [5](#)