Package 'mrdrc'

April 17, 2009

Title Model-robust concentration-response analysis

LazyLoad yes

LazyData yes

Version 1.0-2

Author Christian Ritz <ritz@life.ku.dk>, Mads Jeppe Tarp-Johansen,

Maintainer Christian Ritz <ritz@life.ku.dk>

Description Model-robust, semi-parametric modelling of continuous and quantal concentration/dose-response data.

License GPL (>= 2)

Depends R (>= 2.6.0), drc, locfit, EffectiveDose

Date 2009-02-17

URL http://www.r-project.org

Repository CRAN

Date/Publication 2009-02-17 09:35:34

R topics documented:

bin.mat 2
exp.a 3
pnsdrm 4

Index **8**

`bin.mat` *Quantal data from ECVAM*

Description

Quantal dose-response data provided by ECVAM.

Usage

```
data(bin.fer)

data(bin.mat)
```

Format

A data frame with 15 observations on the following 4 variables.

`exp` a factor with levels 20-09-2000, 01-05-2002, 09-05-2001 for `bin.mat` and 04-07-2001, 25-04-2002, 01-05-2002 for `bin.fer`

`conc` a numeric vector of concentrations applied

`matured` a numeric vector of the numbers of organisms matured

`fertilized` a numeric vector of the numbers of organisms fertilized

`total` a numeric vector of the total numbers of organisms

Details

Both datasets stem from experiments carried out under standard conditions, carried out on three different days.

Source

Data are kindly provided by European Centre for the Validation of Alternative Methods (ECVAM), EU Joint Research Centre, Ispra, Italy.

Examples

```
## Analysing bin.mat using a semi-parametric model
bin.mat.mrr1 <- pnsdrm(bin.mat[c(1,4,7,10,13),]$conc,
bin.mat[c(1,4,7,10,13),]$matured,
bin.mat[c(1,4,7,10,13),]$total, type = "binomial",
model = "semi-parametric", fct = LL.2())

bin.mat.mrr1
plot(bin.mat.mrr1)

## Non-parametric model
bin.mat.mrr2 <- pnsdrm(bin.mat[c(1,4,7,10,13),]$conc,
```

```
bin.mat[c(1,4,7,10,13),]$matured,
bin.mat[c(1,4,7,10,13),]$total, type = "binomial",
model = "non-parametric")

bin.mat.mrr2
plot(bin.mat.mrr2)

## Parametric model
bin.mat.mrr3 <- pnsdrm(bin.mat[c(1,4,7,10,13),]$conc,
bin.mat[c(1,4,7,10,13),]$matured,
bin.mat[c(1,4,7,10,13),]$total, type = "binomial",
model = "parametric", fct = LL.2())

bin.mat.mrr3
plot(bin.mat.mrr3)
```

`exp.a` *Continuous data from ECVAM*

Description

Continuous data provided by ECVAM from tests of compounds denoted "a", "b" and so on, within a specific test system.

Usage

```
data(exp.a)

data(exp.az)
```

Format

Data frames with 54 observations and 26 times 54 observations, respectively, on the following 2 variables.

`conc` a numeric vector of concentrations

`response` a numeric vector of response values

`exp` a character vector indicating the compound being tested.

Details

All datasets stem from experiments carried out under standard conditions, having replicates for all concentrations.

Source

Data are kindly provided by European Centre for the Validation of Alternative Methods (ECVAM), EU Joint Research Centre, Ispra, Italy.

Examples

```
## Fitting a semi-parametric model
exp.a.mrr1 <- pnsdrm(exp.a$x, exp.a$y, type = "continuous",
model = "semi-parametric", fct = list(LL.3(), LL.4()), respLev = c(10, 20, 50),
reference = NULL, level = 0.95, robust = FALSE)

## Showing the model fit
exp.a.mrr1
exp.a.mrr1$xtables

## Plotting the data and the fit
plot(exp.a.mrr1)
```

`pnsdrm` *Parametric, non-parametric or semi-parametric dose-response modelling*

Description

Parametric, non-parametric or semi-parametric dose-response modelling of both continuous and quantal data.

Usage

```
pnsdrm(predictor, response, weights, type = c("continuous", "binomial"),
model = c("semi-parametric", "non-parametric", "parametric"),
fct = NULL, robust = FALSE, respLev = c(10, 20, 50),
reference = NULL, level = 0.95, logex = FALSE)

pnsdrm.calc(predictor, response, weights, type = c("continuous", "binomial"),
model = c("semi-parametric", "non-parametric", "parametric"),
fct = NULL, robust = FALSE, respLev = c(10, 20, 50),
reference = NULL, level = 0.95, logex = FALSE)
```

Arguments

`predictor`	numeric vector of concentrations/doses.
`response`	numeric vector of response values (proportions in case of quantal data).
`weights`	numeric vector of weights needed for quantal data.
`type`	character string specifying the type of response.
`model`	character string specifying the model to be fit.
`fct`	a built-in function or a list of built-in functions from the package 'drc'.
`robust`	logical specifying whether or not a robust approach should be used. Only for the semi-parametric approach.

respLev	numeric vector of requested ED level.
reference	optional reference value for the lower limit.
level	numeric specifying the confidence level.
logex	logical indicating whether or not a logarithmic x axis should be used.

Details

The parametric estimation is based on the model fitting function drm in the package 'drc'. The non-parametric estimation relies on the 'locfit' package.

The semi-parametric approach is mainly based on the development in Nottingham and Birch (2000), whereas the non-parametric approach uses on the package 'EffectiveDose' which implements the method introduced in Dette *et al* (2004).

plot and print methods are available.

Value

A list containing the requested ED values and additional information about the underlying model fit(s).

Note

The implementation of this function as well as all other functions in the package 'mrdrc' has been funded by European Centre for the Validation of Alternative Methods, EU Joint Research Centre under lot 3 of the project "Quality assessment and novel statistical analysis techniques for toxicological data".

Author(s)

Christian Ritz (wrapper functions) Mads Jeppe Tarp-Johansen (internal functions)

References

Dette, H., Neumeyer, N. and Pilz, K. F. (2004) A Note on Nonparametric Estimation of the Effective Dose in Quantal Bioassay, *J. Amer. Statist. Assoc.*, **100**, 503–510.

Nottingham, Q. and Birch, J. B. (2000) A Semiparametric Approach to Analysing Dose-Response Data, *Statist. Med.*, **19**, 389–404.

See Also

More examples are found in the help pages for bin.mat and exp.a.

Examples

```
## Analysing deguelin (in the package 'drc')

## Semi-parametric model
deguelin.mrr1 <- pnsdrm(deguelin$dose, deguelin$r, deguelin$n, type = "binomial",
model = "semi-parametric", fct = LL.2())
```

```
deguelin.mrr1
plot(deguelin.mrr1)

## The same
gmFct <- getMeanFunctions(fname = "LL.2")
deguelin.mrr1b <- pnsdrm(deguelin$dose, deguelin$r, deguelin$n, type = "binomial",
model = "semi-parametric", fct = gmFct)
deguelin.mrr1b
plot(deguelin.mrr1b)

## The same again
deguelin.mrr1c <- pnsdrm(deguelin$dose, deguelin$r, deguelin$n, type = "binomial",
model = "semi-parametric", fct = list(LL2.2()))
deguelin.mrr1c
plot(deguelin.mrr1c)

deguelin.mrr1d <- pnsdrm(deguelin$dose, deguelin$r, deguelin$n, type = "binomial",
model = "semi-parametric", fct = W1.2())
deguelin.mrr1d
plot(deguelin.mrr1d)

## The same
gmFct <- getMeanFunctions(fname = "W1.2")
deguelin.mrr1e <- pnsdrm(deguelin$dose, deguelin$r, deguelin$n, type = "binomial",
model = "semi-parametric", fct = gmFct)
deguelin.mrr1e
plot(deguelin.mrr1e)

## Parametric models
deguelin.mrr2 <- pnsdrm(deguelin$dose, deguelin$r, deguelin$n, type = "binomial",
model = "parametric", fct = list(LL.2(), W1.2(), W2.2()))
deguelin.mrr2
plot(deguelin.mrr2)

## The same parametric models
deguelin.mrr2b <- pnsdrm(deguelin$dose, deguelin$r, deguelin$n, type = "binomial",
model = "parametric", fct = list(W2.2(), LL.2(), W1.2()))
deguelin.mrr2b
plot(deguelin.mrr2b)

## Non-parametric approach
deguelin.mrr3 <- pnsdrm(deguelin$dose, deguelin$r, deguelin$n, type = "binomial",
model = "non-parametric")
deguelin.mrr3
plot(deguelin.mrr3)

## Semi-parametric model with reference level 0.3
deguelin.mrr4 <- pnsdrm(deguelin$dose, deguelin$r, deguelin$n, type = "binomial",
model = "semi-parametric", fct = LL.2(), reference = 0.3)
deguelin.mrr4
plot(deguelin.mrr4)

## Semi-parametric models
```

```
deguelin.mrr5 <- pnsdrm(deguelin$dose, deguelin$r, deguelin$n, type = "binomial",
model = "semi-parametric", fct = list(LL.2(), W1.2(), W2.2()))
deguelin.mrr5
plot(deguelin.mrr5)

## Analysing ryegrass (in the package 'drc')

ryegrass.mrr1 <- pnsdrm(ryegrass$conc, ryegrass$rootl, type = "continuous",
model = "semi-parametric", fct = LL.5())
ryegrass.mrr1
plot(ryegrass.mrr1)
plot(ryegrass.mrr1, log = "x")

ryegrass.mrr2 <- pnsdrm(ryegrass$conc, ryegrass$rootl, type = "continuous",
model = "semi-parametric", fct = list(LL.3(), LL.4(), LL.5()))
ryegrass.mrr2
plot(ryegrass.mrr2)

ryegrass.mrr3 <- pnsdrm(ryegrass$conc, ryegrass$rootl, type = "continuous",
model = "parametric", fct = list(LL.3(), LL.4(), LL.5()))
ryegrass.mrr3
plot(ryegrass.mrr3)

ryegrass.mrr4 <- pnsdrm(ryegrass$conc, ryegrass$rootl, type = "continuous",
model = "semi-parametric", fct = list(L.4(), LL.4(), W1.4(), W2.4()))
ryegrass.mrr4
plot(ryegrass.mrr4)

## Analysing lettuce (in the package 'drc')

lettuce.mrr1 <- pnsdrm(lettuce$conc, lettuce$weight, type = "continuous",
model = "semi-parametric", fct = LL.3())
lettuce.mrr1
plot(lettuce.mrr1)

lettuce.mrr2 <- pnsdrm(lettuce$conc, lettuce$weight, type = "continuous",
model = "semi-parametric", fct = BC.4())
lettuce.mrr2
plot(lettuce.mrr2)

#lettuce.mrr3 <- pnsdrm(lettuce$conc, lettuce$weight, type = "continuous",
#model = "semi-parametric", fct = LL.3(), robust = TRUE)

#lettuce.mrr3
#plot(lettuce.mrr3)
```

Index

∗Topic **datasets**
bin.mat, 2
exp.a, 3
∗Topic **models**
pnsdrm, 4
∗Topic **nonlinear**
pnsdrm, 4

bin.fer *(bin.mat)*, 2
bin.mat, 2, *5*

drm, *5*

exp.a, 3, *5*
exp.az *(exp.a)*, 3

pns.plot1 *(pnsdrm)*, 4
pnsdrm, 4