

Package ‘calib’

July 18, 2009

Version 2.0.1

Date 07-07-09

Title Statistical tool for calibration of plate based bioassays

Author Daniel Samarov <daniel_samarov@bd.com>, Perry Haaland <perry_haaland@bd.com>, Elaine McVey <elaine_mcvey@bd.com>

Maintainer Perry Haaland <perry_d_haaland@bd.com>

Depends R (>= 2.7.0), stats, nlme, methods

Imports graphics

Suggests

Description Provides basic S4 data structures and routines for calibration of bioassays

License GPL-2

URL <http://www.bd.com>

biocViews Statistics, Infrastructure, CellBasedAssays

Repository CRAN

Date/Publication 2009-07-18 15:40:26

R topics documented:

calib	2
calib-class	4
calib.fit	5
calib.fit-class	7
coef-methods	9
coefficients-methods	9
ELISA	10
experimentData	10
fitted-methods	11

fpl.model	11
HPLC	12
lin.model	12
loq	13
mdc	13
plot-methods	14
print-methods	14
rdl	15
resid-methods	15
residuals-methods	16
RIA	16
RIAy0	17
show-methods	17
thpl.model	18
tpl.model	18

Index	20
--------------	-----------

calib	<i>Calibration function</i>
-------	-----------------------------

Description

Computes the calibration statistics

Usage

```
calib(x, y0, conf = 0.9, dilution = 1, samp.names, m = x@m, truth,
times, samp.units = "", dose.units = "", dose.name = "",
maxit = 1000, toler = 1e-05, rname = "response", extrap = FALSE, xname = x)
```

Arguments

x	Output from calib.fit.
y0	Points to be calibrated.
conf	A vector of mean response values to predict the unknown x.
dilution	Dilution factor.
samp.names	Names of the unknowns.
m	Number of replicates.
truth	Optional argument to provide true concentrations if they are known
times	...
samp.units	Names of the unknowns
dose.units	Units of dose
dose.name	Name of dose
maxit	Maximum number of iterations to use in optimization

<code>toler</code>	Tolerance for optimization step
<code>rname</code>	This is the name of the reponse variable
<code>extrap</code>	Option to extrapolate out of range values
<code>xname</code>	Names of concentrations

Value

<code>Estimated.x</code>	Predicted values of x (for example concentration)
<code>PredStdErr</code>	The predicted standard errors of the estimated x's
<code>inver.low</code>	The estimate of the lower confidence limit for the predicted x's using inverse estimation
<code>inver.up</code>	The estimate of the upper confidence limit for the predicted x's using inverse estimation
<code>wald.low</code>	The estimate of the lower confidence limit for the predicted x's using Wald estimation
<code>wald.up</code>	The estimate of the upper confidence limit for the predicted x's using Wald estimation
<code>avg.response</code>	y0 values

Author(s)

Perry Haaland, Elaine McVey, Daniel Samarov

References

Davidian and Haaland 1990

See Also

[calib-class](#), [calib-class](#), [calib-class](#), [calib-class](#)

Examples

```
data(HPLC)
attach(HPLC)
model <- calib.fit(Concentration, Response)
calib(model, Concentration)
```

`calib-class`*Object of class calib*

Description

This is an object generated by using the `calib` function on an object of class `calib.fit`. Contains results from performing calibration.

Slots

Estimated.x: Predicted values of x
PredStdErr: Predicted standard errors
inver.low: Estimated lower confidence intervals using inverse method
inver.up: Estimated upper confidence intervals using inverse method
wald.low: Estimated lower confidence interval using Wald intervals
wald.up: Estimated upper confidence interval using Wald intervals
waldl.low: Internal use
waldl.up: Internal use
avg.response: y0 values
dilution: Dilution factors
oor: Values which fall out of range of assay
lmdc: Internal use
row.names: Internal use
labels: Internal use
max.x: Internal use
extrap: Should values outside the concentration range be estimated
repeq: Internal use
rname: Internal use
conf.level: Values to be used for estimating confidence intervals of estimated concentrations
type: Model type used (fpl, thpl or lin)
mdc: Minimum detectable concentration values
truth: If true concentration values are provided in the calib function
times: Internal use
samp.names: Names of samples

Methods

plot Plot results from a calib object.
print Print results from a calib object.
show Same as print.
summary Same as print.

Author(s)

Perry Haaland, Elaine McVey, Daniel Samarov

See Also

[calib.fit-class](#), [calib.fit-class](#), [calib.fit-class](#), [calib.fit-class](#),
[calib.fit-class](#), [calib.fit-class](#), [calib.fit-class](#)

Examples

```
showClass("calib")
```

calib.fit

General Standard Curve Fitting

Description

calib.fit utilizes two main model types, linear and logistic, for the purpose of standard curve fitting. It also incorporates several alternatives within each model type to allow for more flexible and reliable fitting.

Usage

```
calib.fit(x, y, b1start, b2start, b3start, b4start, calcDiagnostics = TRUE,
m, cv = 0.2, conf = 0.95, mx = 50, lof.calc = T, lowLim = 0.001,
type = c("log.fpl.pom", "fpl.pom", "log.fpl", "fpl", "log.tpl.pom", "tpl.pom",
"log.tpl", "tpl", "quad.pom", "log.thpl.pom", "thpl.pom", "log.thpl",
"thpl", "lin.pom"))
```

Arguments

x	Independent variable (for example dose)
y	Dependent variable (for example response)
b1start	Starting value for b1 in either the fpl, tpl or thpl models.
b2start	Starting value for b2 in either the fpl, tpl or thpl models.
b3start	Starting value for b3 in either the fpl, tpl or thpl models.
b4start	Starting value for b4 in either the fpl or tpl models.
calcDiagnostics	Should diagnostics (i.e. mdc, rdl and loq) be calculated. Default to TRUE.
m	Number of repeated measurements
cv	The acceptable coefficient of variation. The limits of quantitation are calculated with this constraint
conf	The confidence level used for the determining the prediction interval
mx	The maximum number of iterations used in the non-linear least-squares fit

lof.calc	Should the lack of fit statistics be calculated. The default is TRUE.
lowLim	If there are x values equal to zero what (small) positive value be used to approximate it. Defaults is 0.001.
type	Can take the values <code>log.fpl.pom</code> , the log parameterized four parameter logistic regression (<code>fpl</code>) fit with power of the mean (POM), <code>fpl.pom</code> , <code>fpl</code> fit with POM, <code>log.fpl</code> , log parameterized <code>fpl</code> , <code>fpl</code> , standard <code>fpl</code> , <code>thpl.pom</code> , three parameter logistic (<code>thpl</code>) regression fit with POM, <code>thpl</code> without POM <code>thpl</code> , log parameterized POM <code>thpl</code> , <code>log.thpl.pom</code> , log parameterized <code>thpl</code> , <code>log.thpl</code> , <code>lin.quad.pom</code> , linear regression with a quadrattic term fit by POM, <code>lin.pom</code> , linear regression fit by POM.

Value

<code>this-is-escaped-codenormal-bracket70bracket-normal:</code>	Estimates of the coefficients
<code>this-is-escaped-codenormal-bracket73bracket-normal:</code>	Estimates of the standard errors for the coefficients
<code>this-is-escaped-codenormal-bracket76bracket-normal:</code>	Standard deviation of model
<code>this-is-escaped-codenormal-bracket79bracket-normal:</code>	Unscaled variance-covariance matrix of the coefficients
<code>this-is-escaped-codenormal-bracket82bracket-normal:</code>	Whether or not POM model was used, TRUE or FALSE.
<code>this-is-escaped-codenormal-bracket85bracket-normal:</code>	The estimated value of POM parameter theta
<code>this-is-escaped-codenormal-bracket88bracket-normal:</code>	The residual degree of freedom
<code>this-is-escaped-codenormal-bracket91bracket-normal:</code>	The estimated fitted values
<code>this-is-escaped-codenormal-bracket94bracket-normal:</code>	The values of the residuals
<code>this-is-escaped-codenormal-bracket97bracket-normal:</code>	Which algorithm was used to do the optimization (i.e. ML, RML, EM, etc.)
<code>this-is-escaped-codenormal-bracket100bracket-normal:</code>	The number of iteration to convergence of the GLS fit
<code>this-is-escaped-codenormal-bracket103bracket-normal:</code>	Indicated whether the algorithm converged
<code>this-is-escaped-codenormal-bracket106bracket-normal:</code>	The x values
<code>this-is-escaped-codenormal-bracket109bracket-normal:</code>	The y values
<code>this-is-escaped-codenormal-bracket112bracket-normal:</code>	Indicated whether a log parameterization of the model used
<code>this-is-escaped-codenormal-bracket115bracket-normal:</code>	The number of repeated measures used in the model
<code>this-is-escaped-codenormal-bracket118bracket-normal:</code>	Coefficient of variation used

```

this-is-escaped-codenormal-bracket121bracket-normal:
    Minimum detectable concentration
this-is-escaped-codenormal-bracket124bracket-normal:
    Reliable detection limit
this-is-escaped-codenormal-bracket127bracket-normal:
    Limit of quantitation
this-is-escaped-codenormal-bracket130bracket-normal:
    The gradient matrix based off of the final parameter estimates
this-is-escaped-codenormal-bracket133bracket-normal:
    An slot returning information from a lack of fit (LOF) test.
this-is-escaped-codenormal-bracket136bracket-normal:
    The type of variance model used
this-is-escaped-codenormal-bracket139bracket-normal:
    Confidence level used
this-is-escaped-codenormal-bracket142bracket-normal:
    Whether, fpl, tpl, thpl, lin or quad was used.
this-is-escaped-codenormal-bracket145bracket-normal:
    The general model type used, fpl, thpl or lin

```

Author(s)

Perry Haaland, Elaine McVey, Daniel Samarov

References

Davidian and Haaland 1990

See Also

[calib-class](#), [calib-class](#), [calib-class](#), [calib-class](#), [calib-class](#), [calib-class](#), [calib-class](#), [calib-class](#), [calib-class](#), [calib-class](#), [calib-class](#), [calib-class](#)

Examples

```

data (HPLC)
attach (HPLC)
model <- calib.fit (Concentration, Response)

```

calib.fit-class *General Standard Curve Fitting*

Description

calib.fit utilizes two main model types, linear and logistic regression for the purpose of standard curve fitting. It also incorporates several alternatives within each model type to allow for more flexible and reliable fitting.

Slots

coefficients: Estimates of the coefficients
se.coefficients: Estimates of the standard errors for the coefficients
sigma: Standard deviation of model
cov.unscaled: Unscaled variance-covariance matrix of the coefficients
pom: Whether or not POM model was used, TRUE or FALSE.
theta: The estimated value of POM parameter theta
df.residual: The residual degree of freedom
fitted.values: The estimated fitted values
residuals: The values of the residuals
method: Which algorithm was used to do the optimization (i.e. ML, RML, EM, etc.)
kused: The number of iteration to convergence of the GLS fit
status: Indicated whether the algorithm converged
x: The x values
y: The y values
logParm: Indicated whether a log parameterization of the model used
m: The number of repeated measures used in the model
cv: Coefficient of variation used
mdc: Minimum detectable concentration
rdl: Reliable detection limit
loq: Limit of quantitation
gradient: The gradient matrix based off of the final parameter estimates
lof.test: An slot returning information from a lack of fit (LOF) test.
var.model: The type of variance model used
conf.level: Confidence level used
type: Whether, fpl, tpl, thpl, lin or quad was used.
rdlwarn: The general model type used, fpl, thpl or lin

Methods

coef Accessor function for coefficients
coefficients Accessor function for coefficients
fitted Accessor function for the fitted values
loq Accessor function for the limit of quantization
mdc Accessor function for the minimum detectable concentration
plot Plot methods for object of class calib.fit
print Print results from calibration curve fit.
rdl Accessor function the reliable detection limit
resid Accessor function for residuals
residuals Accessor function for residuals
show Same as print
summary Same as print

Author(s)

Perry Haaland, Elaine McVey, Daniel Samarov

See Also

`calib-class`, `calib-class`, `calib-class`, `calib-class`, `calib-class`, `calib-class`, `calib-class`, `calib-class`, `calib-class`, `calib-class`, `calib-class`, `calib-class`

Examples

```
showClass("calib.fit")
```

`coef-methods` *Method coef for 'calib' package*

Description

Returns coefficient information from a `calib.fit` object.

Methods

object = "ANY" Objects `coef` applies to other than `calib.fit` objects.

object = "calib.fit" `calib.fit` object.

`coefficients-methods`
Method coefficients for calib package

Description

Returns coefficient information from a `calib.fit` object.

Methods

object = "ANY" Objects `coefficients` applies to other than `calib.fit` objects.

object = "calib.fit" `calib.fit` object.

ELISA

Enzyme-Linked ImmunoSortbent Assay Data (ELISA)

Description

A data set containing with the results from an ELISA experiment

Usage

```
data(ELISA)
```

Format

A data frame with 16 observations on the following 2 variables.

Concentration Concentration analyte

Response Response of assay

Examples

```
data(ELISA)
```

experimentData

Experimental Dataset

Description

Dataset similar to ELISA dataset

Usage

```
data(experimentData)
```

Format

A data frame with 72 observations on the following 3 variables.

group Unique identifier

OD A numeric vector of respnses

dilution A numeric vector of dilutions

Examples

```
data(experimentData)
```

fitted-methods	<i>Method fitted for calib.fit object.</i>
----------------	--

Description

Returns fitted values from a `calib.fit` object.

Methods

object = "ANY" Objects other than `calib.fit` for which `fitted` can be used.

object = "calib.fit" `calib.fit` object.

<code>fpl.model</code>	<i>Four Parameter Logistic Model</i>
------------------------	--------------------------------------

Description

Internal function, calculates the response values for given set of `x` values and the associated `gradient`.

Usage

```
fpl.model(x, b1, b2, b3, b4, w = 1, logParm = TRUE)
```

Arguments

<code>x</code>	Independent variables.
<code>b1</code>	Value for b1.
<code>b2</code>	Value for b2.
<code>b3</code>	Value for b3.
<code>b4</code>	Value for b4.
<code>w</code>	Weight vector.
<code>logParm</code>	Should log parameterization be used.

Author(s)

Perry Haaland, Elaine McVey, Daniel Samarov

See Also

[calib.fit](#)

HPLC

High Performance Liquid Chromatography (HPLC) dataset

Description

A data set with the results from an HPLC experiment

Usage

```
data(HPLC)
```

Format

A data frame with 51 observations on the following 2 variables.

Concentration Concentration of analyte

Response Response of assay

Examples

```
data(HPLC)
```

lin.model

Linear or Quadratic model.

Description

Internal function, calculates the response values for given set of x values and the associated gradient.

Usage

```
lin.model(x, beta, w = 1, type)
```

Arguments

x	Independent variables.
beta	If <code>type = "lin"</code> , beta is a single value, if <code>type = "quad"</code> , beta is a two values.
w	Weight vector.
type	lin or quad.

Author(s)

Perry Haaland, Elaine McVey, Daniel Samarov

See Also[calib.fit](#)

`loq`*Limit of Quantization*

Description

Computes `loq` value.

Usage

```
loq(x, ...)
```

Arguments

<code>x</code>	An object of class <code>calib.fit</code>
<code>...</code>	NA

Value

<code>value</code>	Limit of quantization
--------------------	-----------------------

Author(s)

Perry Haaland, Elaine McVey, Daniel Samarov

Examples

```
data(ELISA)
attach(ELISA)
model <- calib.fit(Concentration, Response)
loq(model)
```

`mdc`*Minimum Detectable Concentration*

Description

Returns the minimum detectable concentration based on the model fitted by `calib.fit`

Usage

```
mdc(x, ...)
```

Arguments

`x` An object of class `calib.fit`
`...` NA

Details

Calculates the minimum detectable concentrations

Value

value mdc value

Author(s)

Perry Haaland, Elaine McVey, Daniel Samarov

Examples

```
data(HPLC)
attach(HPLC)
model <- calib.fit(Concentration, Response)
mdc(model)
```

plot-methods *Plot method*

Description

Plotting method for `calib.fit` and `calib` Provides plotting methods

Methods

`x = "ANY", y = "ANY"` ~~describe this method here
`x = "calib", y = "missing"` ~~describe this method here
`x = "calib.fit", y = "missing"` ~~describe this method here

print-methods *Method print for calib.fit object.*

Description

Prints output relevant to a `calib.fit` or `calib` object.

Methods

`x = "ANY"` Objects other than `calib.fit` and `calib` for which `print` can be used.
`x = "calib"` `calib` object.
`x = "calib.fit"` `calib.fit` object.

rdl *Reliable Detection Limit*

Description

Methods for function rdl

Usage

```
rdl(x, ...)
```

Arguments

x	An object of class <code>calib.fit</code>
...	NA

Value

value	Reliable detection limit
-------	--------------------------

Author(s)

Perry Haaland, Elaine McVey, Daniel Samarov

Examples

```
data(ELISA)
attach(ELISA)
model <- calib.fit(Concentration, Response)
rdl(model)
```

resid-methods *Method resid for package calib*

Description

Methods for function `resid` in Package ‘calib’

Methods

object = "ANY" ~~describe this method here

object = "calib.fit" ~~describe this method here

residuals-methods *Methods for Function residuals in Package calib*

Description

Returns residuals from an object `calib.fit`.

Methods

object = "ANY" Objects other than `calib.fit` object.

object = "calib.fit" `calib.fit` object.

RIA

Radioimmunoassay

Description

Dataset from the Davidian et al. 1996.

Usage

```
data(RIA)
```

Format

The format is: `chr "RIA"`

References

Dataset from the Davidian et al. 1996.

Examples

```
data(RIA)
```

RIAy0

Radio Immunoassay

Description

Dataset from the Davidian et. al 1996 paper.

Usage

```
data(RIAy0)
```

Format

The format is: chr "RIAy0"

Source

Dataset from the Davidian et al. 1996 paper.

References

Davidian et al. 1996

Examples

```
data(RIAy0)
```

show-methods

Methods for Function show in Package 'calib'

Description

Methods for function show in Package 'calib'

Methods

```
object = "ANY" NA  
object = "calib" calib object.  
object = "calib.fit" calib.fit object.  
object = "classRepresentation" NA  
object = "genericFunction" NA  
object = "MethodDefinition" NA  
object = "MethodWithNext" NA  
object = "ObjectsWithPackage" NA  
object = "signature" NA  
object = "traceable" NA
```

`thpl.model`*Three Parameter Logistic Model*

Description

Internal function, calculates the response values for given set of `x` values and the associated `gradient`.

Usage

```
thpl.model(x, b1, b2, b3, w = 1, logParm = TRUE)
```

Arguments

<code>x</code>	Independent variables.
<code>b1</code>	Value for <code>b1</code> .
<code>b2</code>	Value for <code>b2</code> .
<code>b3</code>	Value for <code>b3</code> .
<code>w</code>	Weight vector.
<code>logParm</code>	Should log parameterization be used.

Author(s)

Perry Haaland, Elaine McVey, Daniel Samarov

See Also

[calib.fit](#)

`tpl.model`*Two Parameter Logistic Model*

Description

Internal function, calculates the response values for given set of `x` values and the associated `gradient`.

Usage

```
tpl.model(x, b1, b2, b3, b4, w = 1, logParm = TRUE)
```

Arguments

x	Independent variables.
b1	Value for b1.
b2	Value for b2.
b3	Value for b3.
b4	Value for b4.
w	Weight vector.
logParm	Should log parameterization be used.

Author(s)

Perry Haaland, Elaine McVey, Daniel Samarov

See Also

[calib.fit](#)

Index

*Topic **classes**

calib-class, 3
calib.fit-class, 7

*Topic **datasets**

ELISA, 10
experimentData, 10
HPLC, 12
RIA, 16
RIAy0, 17

*Topic **methods**

coef-methods, 9
coefficients-methods, 9
fitted-methods, 11
loq, 13
mdc, 13
plot-methods, 14
print-methods, 14
rdl, 15
resid-methods, 15
residuals-methods, 16
show-methods, 17

*Topic **models**

calib, 2
calib.fit, 5
fpl.model, 11
lin.model, 12
thpl.model, 18
tpl.model, 18

calib, 2, 14
calib-class, 3, 7, 9
calib-class, 3
calib.fit, 3, 5, 11, 13, 14, 18, 19
calib.fit-class, 4
calib.fit-class, 7
coef, ANY-method (*coef-methods*), 9
coef, calib.fit-method
(*calib.fit-class*), 7
coef-methods, 9

coefficients, calib.fit-method
(*calib.fit-class*), 7
coefficients-methods, 9

ELISA, 10
experimentData, 10

fitted, ANY-method
(*fitted-methods*), 11
fitted, calib.fit-method
(*calib.fit-class*), 7
fitted-methods, 11
fpl.model, 11

HPLC, 12

lin.model, 12
loq, 13
loq, calib.fit-method
(*calib.fit-class*), 7

mdc, 13
mdc, calib.fit-method
(*calib.fit-class*), 7

plot, ANY, ANY-method
(*plot-methods*), 14
plot, calib, missing-method
(*calib-class*), 3
plot, calib.fit, missing-method
(*calib.fit-class*), 7
plot-methods, 14
print, ANY-method (*print-methods*),
14
print, calib-method (*calib-class*),
3
print, calib.fit-method
(*calib.fit-class*), 7
print-methods, 14
rdl, 15

`rdl`, `calib.fit`-method
 (*calib.fit-class*), 7
`resid`, ANY-method (*resid-methods*),
 15
`resid`, `calib.fit`-method
 (*calib.fit-class*), 7
`resid-methods`, 15
`residuals`, ANY-method
 (*residuals-methods*), 16
`residuals`, `calib.fit`-method
 (*calib.fit-class*), 7
`residuals-methods`, 16
`RIA`, 16
`RIAy0`, 17

`show`, ANY-method (*show-methods*), 17
`show`, `calib`-method (*calib-class*), 3
`show`, `calib.fit`-method
 (*calib.fit-class*), 7
`show`, `classRepresentation`-method
 (*show-methods*), 17
`show`, `genericFunction`-method
 (*show-methods*), 17
`show`, `MethodDefinition`-method
 (*show-methods*), 17
`show`, `MethodWithNext`-method
 (*show-methods*), 17
`show`, `ObjectsWithPackage`-method
 (*show-methods*), 17
`show`, `signature`-method
 (*show-methods*), 17
`show`, `traceable`-method
 (*show-methods*), 17
`show-methods`, 17
`summary`, `calib`-method
 (*calib-class*), 3
`summary`, `calib.fit`-method
 (*calib.fit-class*), 7

`thpl.model`, 18
`tpl.model`, 18