Package 'chemCal'

July 17, 2018

Version 0.2.1

Date 2018-07-17

Title Calibration Functions for Analytical Chemistry

Suggests MASS, knitr, testthat, investr

Description Simple functions for plotting linear
calibration functions and estimating standard errors for measurements
according to the Handbook of Chemometrics and Qualimetrics: Part A
by Massart et al. There are also functions estimating the limit
of detection (LOD) and limit of quantification (LOQ).
The functions work on model objects from - optionally weighted - linear
regression (lm) or robust linear regression ('rlm' from the 'MASS' package).

License GPL (>= 2)

LazyLoad yes

LazyData yes

VignetteBuilder knitr

Encoding UTF-8

BugReports http://github.com/jranke/chemCal/issues

URL https://pkgdown.jrwb.de/chemCal,

  https://cgit.jrwb.de/chemCal/about

RoxygenNote 6.0.1

NeedsCompilation no

Author Johannes Ranke [aut, cre, cph]
(<https://orcid.org/0000-0003-4371-6538>)

Maintainer Johannes Ranke <jranke@uni-bremen.de>

Repository CRAN

Date/Publication 2018-07-17 18:00:02 UTC

R topics documented:

---

chemCal-package   *Calibration functions for analytical chemistry*

Description

See ../DESCRIPTION

Details

There is a package vignette located in ../doc/chemCal.pdf.

Author(s)

Author and Maintainer: Johannes Ranke <jranke@uni-bremen.de>

---

calplot   *Plot calibration graphs from univariate linear models*

Description

Produce graphics of calibration data, the fitted model as well as confidence, and, for unweighted regression, prediction bands.

Usage

```
calplot(object, xlim = c("auto", "auto"), ylim = c("auto", "auto"),
  xlab = "Concentration", ylab = "Response", alpha=0.05, varfunc = NULL)
```

Arguments

object	A univariate model object of class `lm` or `rlm` with model formula y ~ x or y ~ x - 1.
xlim	The limits of the plot on the x axis.
ylim	The limits of the plot on the y axis.
xlab	The label of the x axis.
ylab	The label of the y axis.
alpha	The error tolerance level for the confidence and prediction bands. Note that this includes both tails of the Gaussian distribution, unlike the alpha and beta parameters used in `lod` (see note below).
varfunc	The variance function for generating the weights in the model. Currently, this argument is ignored (see note below).

Value

A plot of the calibration data, of your fitted model as well as lines showing the confidence limits. Prediction limits are only shown for models from unweighted regression.

Note

Prediction bands for models from weighted linear regression require weights for the data, for which responses should be predicted. Prediction intervals using weights e.g. from a variance function are currently not supported by the internally used function `predict.lm`, therefore, `calplot` does not draw prediction bands for such models.

It is possible to compare the `calplot` prediction bands with the `lod` values if the lod() alpha and beta parameters are half the value of the calplot() alpha parameter.

Author(s)

Johannes Ranke <jranke@uni-bremen.de> <http://www.uft.uni-bremen.de/chemie/ranke>

Examples

```
data(massart97ex3)
m <- lm(y ~ x, data = massart97ex3)
calplot(m)
```

---

din32645   *Calibration data from DIN 32645*

Description

Sample dataset to test the package.

Usage

```
data(din32645)
```

Format

A dataframe containing 10 rows of x and y values.

References

DIN 32645 (equivalent to ISO 11843), Beuth Verlag, Berlin, 1994

Dintest. Plugin for MS Excel for evaluations of calibration data. Written by Georg Schmitt, University of Heidelberg. Formerly available from the Website of the University of Heidelberg.

Currie, L. A. (1997) Nomenclature in evaluation of analytical methods including detection and quantification capabilities (IUPAC Recommendations 1995). Analytica Chimica Acta 391, 105 - 126.

Examples

```
m <- lm(y ~ x, data = din32645)
calplot(m)

## Prediction of x with confidence interval
prediction <- inverse.predict(m, 3500, alpha = 0.01)

# This should give 0.07434 according to test data from Dintest, which
# was collected from Procontrol 3.1 (isomehr GmbH) in this case
round(prediction$Confidence, 5)

## Critical value:
crit <- lod(m, alpha = 0.01, beta = 0.5)

# According to DIN 32645, we should get 0.07 for the critical value
# (decision limit, "Nachweisgrenze")
round(crit$x, 2)
# and according to Dintest test data, we should get 0.0698 from
round(crit$x, 4)

## Limit of detection (smallest detectable value given alpha and beta)
# In German, the smallest detectable value is the "Erfassungsgrenze", and we
# should get 0.14 according to DIN, which we achieve by using the method
# described in it:
lod.din <- lod(m, alpha = 0.01, beta = 0.01, method = "din")
round(lod.din$x, 2)

## Limit of quantification
# This accords to the test data coming with the test data from Dintest again,
# except for the last digits of the value cited for Procontrol 3.1 (0.2121)
loq <- loq(m, alpha = 0.01)
round(loq$x, 4)

# A similar value is obtained using the approximation
```

```
# LQ = 3.04 * LC (Currie 1999, p. 120)
3.04 * lod(m, alpha = 0.01, beta = 0.5)$x
```

---

inverse.predict   *Predict x from y for a linear calibration*

Description

This function predicts x values using a univariate linear model that has been generated for the pur-
pose of calibrating a measurement method. Prediction intervals are given at the specified confidence
level. The calculation method was taken from Massart et al. (1997). In particular, Equations 8.26
and 8.28 were combined in order to yield a general treatment of inverse prediction for univariate
linear models, taking into account weights that have been used to create the linear model, and at
the same time providing the possibility to specify a precision in sample measurements differing
from the precision in standard samples used for the calibration. This is elaborated in the package
vignette.

Usage

```
inverse.predict(object, newdata, ...,
  ws, alpha=0.05, var.s = "auto")
```

Arguments

object	A univariate model object of class lm or rlm with model formula y ~ x or y ~ x - 1.
newdata	A vector of observed y values for one sample.
...	Placeholder for further arguments that might be needed by future implementa-tions.
ws	The weight attributed to the sample. This argument is obligatory if object has weights.
alpha	The error tolerance level for the confidence interval to be reported.
var.s	The estimated variance of the sample measurements. The default is to take the residual standard error from the calibration and to adjust it using ws, if applica-ble. This means that var.s overrides ws.

Value

A list containing the predicted x value, its standard error and a confidence interval.

Note

The function was validated with examples 7 and 8 from Massart et al. (1997). Note that the
behaviour of inverse.predict changed with chemCal version 0.2.1. Confidence intervals for x values
obtained from calibrations with replicate measurements did not take the variation about the means
into account. Please refer to the vignette for details.

References

Massart, L.M, Vandenginste, B.G.M., Buydens, L.M.C., De Jong, S., Lewi, P.J., Smeyers-Verbeke, J. (1997) Handbook of Chemometrics and Qualimetrics: Part A, p. 200

Examples

```
# This is example 7 from Chapter 8 in Massart et al. (1997)
m <- lm(y ~ x, data = massart97ex1)
inverse.predict(m, 15)       #  6.1 +- 4.9
inverse.predict(m, 90)       # 43.9 +- 4.9
inverse.predict(m, rep(90,5)) # 43.9 +- 3.2

# For reproducing the results for replicate standard measurements in example 8,
# we need to do the calibration on the means when using chemCal > 0.2
weights <- with(massart97ex3, {
  yx <- split(y, x)
  ybar <- sapply(yx, mean)
  s <- round(sapply(yx, sd), digits = 2)
  w <- round(1 / (s^2), digits = 3)
})

massart97ex3.means <- aggregate(y ~ x, massart97ex3, mean)

m3.means <- lm(y ~ x, w = weights, data = massart97ex3.means)

inverse.predict(m3.means, 15, ws = 1.67)  # 5.9 +- 2.5
inverse.predict(m3.means, 90, ws = 0.145) # 44.1 +- 7.9
```

---

lod   *Estimate a limit of detection (LOD)*

Description

The decision limit (German: Nachweisgrenze) is defined as the signal or analyte concentration that is significantly different from the blank signal with a first order error alpha (one-sided significance test). The detection limit, or more precise, the minimum detectable value (German: Erfassungs-grenze), is then defined as the signal or analyte concentration where the probability that the signal is not detected although the analyte is present (type II or false negative error), is beta (also a one-sided significance test).

Usage

```
lod(object, ..., alpha = 0.05, beta = 0.05, method = "default", tol = "default")
```

Arguments

object	A univariate model object of class `lm` or `rlm` with model formula `y ~ x` or `y ~ x - 1`, optionally from a weighted regression.
...	Placeholder for further arguments that might be needed by future implementations.
alpha	The error tolerance for the decision limit (critical value).
beta	The error tolerance beta for the detection limit.
method	The "default" method uses a prediction interval at the LOD for the estimation of the LOD, which obviously requires iteration. This is described for example in Massart, p. 432 ff. The "din" method uses the prediction interval at x = 0 as an approximation.
tol	When the "default" method is used, the default tolerance for the LOD on the x scale is the value of the smallest non-zero standard divided by 1000. Can be set to a numeric value to override this.

Value

A list containig the corresponding x and y values of the estimated limit of detection of a model used for calibration.

Note

- The default values for alpha and beta are the ones recommended by IUPAC. - The estimation of the LOD in terms of the analyte amount/concentration xD from the LOD in the signal domain SD is done by simply inverting the calibration function (i.e. assuming a known calibration function). - The calculation of a LOD from weighted calibration models requires a weights argument for the internally used `predict.lm` function, which is currently not supported in R.

References

Massart, L.M, Vandenginste, B.G.M., Buydens, L.M.C., De Jong, S., Lewi, P.J., Smeyers-Verbeke, J. (1997) Handbook of Chemometrics and Qualimetrics: Part A, Chapter 13.7.8

J. Inczedy, T. Lengyel, and A.M. Ure (2002) International Union of Pure and Applied Chemistry Compendium of Analytical Nomenclature: Definitive Rules. Web edition.

Currie, L. A. (1997) Nomenclature in evaluation of analytical methods including detection and quantification capabilities (IUPAC Recommendations 1995). Analytica Chimica Acta 391, 105 - 126.

See Also

Examples for `din32645`

Examples

```
m <- lm(y ~ x, data = din32645)
lod(m)
```

```
# The critical value (decision limit, German Nachweisgrenze) can be obtained
# by using beta = 0.5:
lod(m, alpha = 0.01, beta = 0.5)
```

---

loq   *Estimate a limit of quantification (LOQ)*

Description

The limit of quantification is the x value, where the relative error of the quantification given the calibration model reaches a prespecified value 1/k. Thus, it is the solution of the equation

$$L = kc(L)$$

where c(L) is half of the length of the confidence interval at the limit L (DIN 32645, equivalent to ISO 11843). c(L) is internally estimated by inverse.predict, and L is obtained by iteration.

Usage

```
loq(object, ..., alpha = 0.05, k = 3, n = 1, w.loq = "auto",
  var.loq = "auto", tol = "default")
```

Arguments

object	A univariate model object of class lm or rlm with model formula y ~ x or y ~ x - 1, optionally from a weighted regression. If weights are specified in the model, either w.loq or var.loq have to be specified.
alpha	The error tolerance for the prediction of x values in the calculation.
...	Placeholder for further arguments that might be needed by future implementations.
k	The inverse of the maximum relative error tolerated at the desired LOQ.
n	The number of replicate measurements for which the LOQ should be specified.
w.loq	The weight that should be attributed to the LOQ. Defaults to one for unweighted regression, and to the mean of the weights for weighted regression. See massart97ex3 for an example how to take advantage of knowledge about the variance function.
var.loq	The approximate variance at the LOQ. The default value is calculated from the model.
tol	The default tolerance for the LOQ on the x scale is the value of the smallest non-zero standard divided by 1000. Can be set to a numeric value to override this.

Value

The estimated limit of quantification for a model used for calibration.

Note

- IUPAC recommends to base the LOQ on the standard deviation of the signal where x = 0. -
The calculation of a LOQ based on weighted regression is non-standard and therefore not tested.
Feedback is welcome.

See Also

Examples for din32645

Examples

```
m <- lm(y ~ x, data = massart97ex1)
loq(m)

# We can get better by using replicate measurements
loq(m, n = 3)
```

---

massart97ex1 *Calibration data from Massart et al. (1997), example 1*

Description

Sample dataset from p. 175 to test the package.

Usage

```
data(massart97ex1)
```

Format

A dataframe containing 6 observations of x and y data.

Source

Massart, L.M, Vandenginste, B.G.M., Buydens, L.M.C., De Jong, S., Lewi, P.J., Smeyers-Verbeke,
J. (1997) Handbook of Chemometrics and Qualimetrics: Part A, Chapter 8.

| massart97ex3 | *Calibration data from Massart et al. (1997), example 3* |

Description

Sample dataset from p. 188 to test the package.

Usage

```
massart97ex3
```

Format

A dataframe containing 6 levels of x values with 5 observations of y for each level.

Source

Massart, L.M, Vandenginste, B.G.M., Buydens, L.M.C., De Jong, S., Lewi, P.J., Smeyers-Verbeke, J. (1997) Handbook of Chemometrics and Qualimetrics: Part A, Chapter 8.

Examples

```
# For reproducing the results for replicate standard measurements in example 8,
# we need to do the calibration on the means when using chemCal > 0.2
weights <- with(massart97ex3, {
  yx <- split(y, x)
  ybar <- sapply(yx, mean)
  s <- round(sapply(yx, sd), digits = 2)
  w <- round(1 / (s^2), digits = 3)
})

massart97ex3.means <- aggregate(y ~ x, massart97ex3, mean)

m3.means <- lm(y ~ x, w = weights, data = massart97ex3.means)

# The following concords with the book p. 200
inverse.predict(m3.means, 15, ws = 1.67)  # 5.9 +- 2.5
inverse.predict(m3.means, 90, ws = 0.145) # 44.1 +- 7.9

# The LOD is only calculated for models from unweighted regression
# with this version of chemCal
m0 <- lm(y ~ x, data = massart97ex3)
lod(m0)

# Limit of quantification from unweighted regression
loq(m0)

# For calculating the limit of quantification from a model from weighted
# regression, we need to supply weights, internally used for inverse.predict
# If we are not using a variance function, we can use the weight from
```

```
# the above example as a first approximation (x = 15 is close to our
# loq approx 14 from above).
loq(m3.means, w.loq = 1.67)
# The weight for the loq should therefore be derived at x = 7.3 instead
# of 15, but the graphical procedure of Massart (p. 201) to derive the
# variances on which the weights are based is quite inaccurate anyway.
```

---

rl95_toluene	*Toluene amounts by GC/MS as reported by Rocke and Lorenzato (1995)*

Description

Dataset reproduced from Table 4 in Rocke and Lorenzato (1995).

Format

A dataframe containing four replicate observations for each of the six calibration standards.

Source

Rocke, David M. und Lorenzato, Stefan (1995) A two-component model for measurement error in analytical chemistry. Technometrics 37(2), 176-184.

---

utstats14	*Example data for calibration with replicates from University of Toronto*

Description

Dataset read into R from http://www.chem.utoronto.ca/coursenotes/analsci/stats/files/example14.xls.

Format

A tibble containing three replicate observations of the response for five calibration concentrations.

Source

David Stone and Jon Ellis (2011) Statistics in Analytical Chemistry. Tutorial website maintained by the Departments of Chemistry, University of Toronto. http://www.chem.utoronto.ca/coursenotes/analsci/stats/index.html

Index