Package ‘envalysis’

April 17, 2020

Type Package

Title Miscellaneous Functions for Environmental Analyses

Version 0.4

Date 2020-04-17

Maintainer Zacharias Steinmetz <steinmetz-z@uni-landau.de>

Description Small toolbox for data analyses in environmental chemistry and ecotoxicology. Provides, for example, `calibration()` to calculate calibration curves and corresponding limits of detection (LODs) and quantification (LOQs) according to German DIN 32645:2008-11. `texture()` makes it easy to estimate soil particle size distributions from hydrometer measurements (ASTM D422-63(2007)e2).

URL https://github.com/zsteinmetz/envalysis

BugReports https://github.com/zsteinmetz/envalysis/issues

Encoding UTF-8

License GPL-3

LazyLoad yes

LazyData yes

VignetteBuilder knitr

Depends R (>= 3.5.0)

Imports drc, ggplot2

Suggests knitr, rmarkdown, testthat, MASS, data.table, tibble, soiltexture

RoxygenNote 7.1.0

NeedsCompilation no

Author Zacharias Steinmetz [aut, cre]

Repository CRAN

Date/Publication 2020-04-17 15:50:05 UTC
Categorize water drop penetration times

Description
This wrapper function categorizes water drop penetration times (WDPT) [s] according to the scale proposed by Bisdom et al. (1993).

Usage
bisdom(wdpt, ...)

Arguments
wdpt a numeric vector containing WDPT measurement data in seconds.
... arguments passed to findInterval.

References

See Also
findInterval for the generic function.

Examples
bisdom(c(2, 6, 20, NA, 3, 385))
calibration

Analytical calibration functions

Description

Defines a calibration object for the calculation of concentrations from measurement signals including estimations for the limit of detection (LOD) and limit of quantification (LOQ) in accordance with DIN 32645:2008-11.

The LOD is defined as the lowest quantity of a substance that can be distinguished from the absence of that substance (blank value) within a given confidence level (alpha). The LOQ is defined as the lowest quantity of a substance that can be quantified/distinguished from another sample given with respect to a defined confidence level (k).

Usage

calibration(formula, data = NULL, model = "lm", ...)

## S3 method for class 'calibration'
print(x, ...)  

## S3 method for class 'calibration'
plot(x, interval = "conf", level = 0.95, ...)  

lod(x, ...)

## S3 method for class 'calibration'
lod(x, alpha = 0.01, level = 0.05, ...)

loq(x, ...)

## S3 method for class 'calibration'
loq(x, alpha = 0.01, k = 3, level = 0.05, maxiter = 10, ...)

Arguments

formula  model formula providing the recorded signal intensities with respect to the nominal analyte concentrations in the form of signal ~ concentration or signal ~ concentration ~1: model formulae are currently restricted to those forms, however, the possibility to use log or sqrt transformed data will be implemented in the future.

data  an optional data frame containing the variables in the model.

model  model class to be used for fitting; currently, lm and rlm are supported.

...  further arguments passed to the sub method, i.e. the respective model environment (e.g. lm), plot, or print.

x  an object of class calibration with a model formula as shown above.
calibration

interval Type of interval calculation (can be abbreviated); see predict for details.
level tolerance/confidence level; see predict and confint for details.
alpha error tolerance for the detection limit (critical value).
k relative uncertainty for the limit of quantification (1/\beta).
maxiter a positive integer specifying the maximum number of iterations to calculate the LOQ.

Details

If the data supplied to calibration contain more than one blank value, i.e. measurements with a nominal concentration of zero, the LOD and LOQ are calculated from the deviation of the blank samples. This method is called "blank method" according to DIN 32645:2008-11 and supposed to be more accurate than the so-called "calibration method" which will be used for the estimation of LOD and LOQ when data does not contain zero concentration measurements.

Value

calibration returns an object of class "calibration". print calls the function parameters together with the respective LOD and LOQ. plot plots the respective calibration curve together with the measurement values. summary may be used to retrieve the model parameters to be found as a list item called "model".

References


See Also

icp, din32645

Examples

data(din32645)
din <- calibration(Area ~ Conc, data = din32645)
din
plot(din, interval = "confidence")
summary(din$model)

lod(din)
loq(din)
Hydrometer readings for a clay loam

Description
Data obtained with a 152H hydrometer in accordance with ASTM D422-63(2007)e2.

Format
A data frame containing 7 rows and 4 columns with information on:

- **time**: the measurement time in minutes
- **temperature**: the temperature of the soil suspension
- **reading**: the hydrometer reading at the bottom of the meniscus
- **blank**: a blank value obtained in 5 g/L sodium hexametaphosphate solution (composite correction)

References

See Also
texture

Calibration data from DIN 32645

Description
Sample data for the calibration of carbon in water.

Format
A data frame containing 20 rows and 2 columns with information on:

- **Conc**: nominal concentration in mg/L
- **Area**: measurement signal
References


See Also

calibration

envalysis | Miscellaneous Functions for Environmental Analyses

Description

Small toolbox for data analyses in environmental chemistry and ecotoxicology. Provides, for example, `calibration()` to calculate calibration curves and corresponding limits of detection (LODs) and quantification (LOQs) according to German DIN 32645:2008-11. `texture()` makes it easy to estimate soil particle size distributions from hydrometer measurements (ASTM D422-63(2007)e2).

Details

Version: 0.4
Licence: GPLv3

Author(s)

Zacharias Steinmetz (<steinmetz-z@uni-landau.de>)

References


icp | ICP-AES calibration data

Description

Sample data of the calibration of silver ions in four-fold replication using inductively coupled plasma atomic emission spectroscopy (ICP-AES).
Format

A data frame containing 16 rows and 6 columns with information on:
### Description

This function should behave just like `drm` from `drc`, with the main difference being that model objects are passed through the function instead of requiring the data to be present in `.GlobalEnv`. If you have trouble with this function, you can use `drc::mselect()` instead.

### Usage

```r
mselect(object, fctList = NULL, nested = FALSE,
         sorted = c("IC", "Res var", "Lack of fit", "no"),
         linreg = FALSE, icfct = AIC)
```

### Arguments

- **object**: an object of class `drc`.
- **fctList**: a list of dose-response functions to be compared.
- **nested**: logical; `TRUE` results in F tests between adjacent models (in `fctList`; only sensible for nested models).
- **sorted**: character string determining according to which criterion the model fits are ranked.
- **linreg**: logical indicating whether or not additionally polynomial regression models (linear, quadratic, and cubic models) should be fitted (they could be useful for a kind of informal lack-of-fit consideration for the models specified, capturing unexpected departures).
- **icfct**: function for supplying the information criterion to be used. `AIC` and `BIC` are two options.

### Details

For Akaike’s information criterion and the residual standard error: the smaller the better and for lack-of-fit test (against a one-way ANOVA model): the larger (the p-value) the better. Note that the residual standard error is only available for continuous dose-response data.

Log likelihood values cannot be used for comparison unless the models are nested.
**Value**

A matrix with one row for each model and one column for each criterion.

**Examples**

```r
library(drc)

ryegrass.m1 <- drm(rootl ~ conc, data = ryegrass, fct = LL.4())
mselect(ryegrass.m1, list(LL.3(), LL.5(), W1.3(), W1.4(), W2.4(), baro5()))
```

**neitzel2003**

*Calibration data from Neitzel, 2003*

**Description**

Artificial sample data for the verification of quantification limits.

**Format**

A data frame containing 20 rows and 2 columns with information on:

- **Conc**: nominal concentration
- **Meas**: measurement signal

**References**


**See Also**

calibration

---

**rmse**

*Root mean square error*

**Description**

This function computes the root mean square error (RMSE) of the two vectors `obs` and `sim`. `rel = FALSE` returns the absolute RMSE, `rel = TRUE` the relative one. If `na.rm` is `TRUE`, missing values are omitted before the computation proceeds.

**Usage**

```r
rmse(obs, sim, rel = F, na.rm = T)
```
Arguments

obs       a numeric vector containing observed values.
sim       a numeric vector containing simulated values.
rel       logical. If TRUE, the relative RMSE is calculated, if FALSE the absolute RMSE is returned.
na.rm     logical. Should missing values be removed?

Examples

rmse(c(0.12,0.59,NA), c(0.15,0.63,1.2))

---

se       Standard error and confidence interval

Description

This wrapper function computes the standard error (SE) or the confidence interval (CI) of the values in x. If na.rm is TRUE, missing values are removed before the computation proceeds.

Usage

se(x, na.rm = FALSE)
CI(x, level = 0.95, na.rm = FALSE)

Arguments

x       a numeric vector or an R object which is coercible to one by as.vector(x,"numeric").
na.rm   logical. Should missing values be removed?
level   the confidence level required.

See Also

sd for the standard deviation.

Examples

se(1:5)
CI(1:5)
signifig

Print significant figures

Description

This function reports the significant figures of a given mean together with its respective error term (e.g. confidence interval or standard deviation).

Usage

signifig(mean, error, data, signif.na = 2, style = "pm")

Arguments

- `mean`: a numeric vector or data frame object containing the averaged values.
- `error`: a numeric vector or data frame object containing the respective error terms.
- `data`: a data frame containing the specified columns. If empty, `mean` and `error` need to be given as numeric vectors.
- `signif.na`: an integer controlling to which significant digit the mean value should be rounded when no error data was given.
- `style`: a string specifying the output style to be used. The default style "pm" reports the results as "3 ± 6", while "par" results in outputs like "0.26 (0.02)". "siunitx" returns "0.26 (2)" which might be used together with xtable for automated LaTeX table outputs.

References


Examples

signifig(mean = c(0.28,5), error = c(0.688, 8))

sorption

Sorption isotherms

Description

This function returns the concentration of a substance sorbed to a surface boundary after an equilibrium has established at constant temperature given the concentration(s) conc of the dissolved substance.
Usage

sorption(conc, par, type = "freundlich")

Arguments

conc a numeric vector containing the concentration(s) of the dissolved substance.
par a numeric vector specifying the function parameters, see examples for details and correct order.
type a character string indicating the type of sorption isotherm to be used: "linear" for the linear type, "freundlich" for the Freundlich isotherm, "langmuir" for the Langmuir isotherm, "BET" for the BET model according to Brunauer, Emmet, and Teller "redlich" for the Redlich-Peterson isotherm.

References


Examples

sorption(conc = 1:5, c(Kd = 2.5), type = "linear")
sorption(conc = 1:5, c(K = 4, n = 0.6), type = "freundlich")
sorption(conc = 1:5, c(KL = 2, qmax = 10), type = "langmuir")
sorption(conc = 1:5, c(K = 50, qmax = 10, Csat = 10), type = "BET")
sorption(conc = 1:5, c(A = 30, B = 0.8), type = "redlich")

Description

Calculates the particle size distribution and both DIN and USDA texture classes from a series of hydrometer readings according to ASTM D422-63(2007)e2.
conc = 50,
Gs = 2.65,
hydrometer = "auto",
model = "auto",
plot = F,
...
)

## S3 method for class 'texture'
print(x, ...)

## S3 method for class 'texture'
plot(x, ...)

Arguments

reading a numeric vector of data values providing the hydrometer readings at the bottom of the meniscus.

... further arguments to be passed to texture (currently not used), print, or plot.

formula an object of class 'formula' of the form reading ~ blank + time + temp.

data a data frame containing the variables in formula.

blank a numeric vector containing the blank readings taken in 5 g/L sodium hexametaphosphate solution (composite correction).

time a numeric vector containing the time passed since the beginning of the measurement in minutes.

temp an integer vector containing the measured temperature.

conc the concentration of the soil solution, default is 50 g/L as proposed in the ASTM guideline.

Gs specific gravity of the suspension.

hydrometer a character string specifying the hydrometer used; accepted values are "auto" for auto-detection (default), "151H", and "152H".

model string is passed to drm, "auto" chooses the best fitting model automatically.

plot logical; if TRUE the particle size distribution is plotted.

x an object of class "texture".

Value

texture returns an object of class "texture". The functions print and plot are available to retrieve the soil texture classes and the particle size distribution, respectively.

An object of class "texture" is a list containing the following components:

- **meta** Measurement meta data
- **distribution** data frame providing the particle size distribution
- **model** information on the fitted drm model
- **din** Main DIN texture classes
- **usda** Main USDA texture classes
References


Examples

data(clayloam)

## Default method
attach(clayloam)
texture(reading, blank, time, temperature)
detach(clayloam)

## Formula interface
texture(reading ~ blank + time + temperature, clayloam)

theme_publish  
ggplot2 theme for scientific publications

Description

Themes set the general aspect of the plot such as the color of the background, gridlines, the size and colour of fonts. This particular theme is based on the classic dark-on-light ggplot2 theme_bw and has been used for scientific publications.

Usage

theme_publish(base_size = 12, base_family = "", line_size = 0.25, ...)

Arguments

base_size  base font size  
base_family base font family  
line_size base line size for, e.g. for ticks and axes  
... further arguments to be passed to theme_bw

See Also

ggtheme
Examples

```r
require(ggplot2)
p <- ggplot(mtcars) + geom_point(aes(x = wt, y = mpg,
                    colour=factor(gear))) + facet_wrap(~ am)
p
p + theme_publish()
```
Index

*Topic data
  clayloam, 5
  din32645, 5
  icp, 6
  neitzel2003, 9
AIC, 8
BIC, 8
bisdom, 2
calibration, 3, 6, 8, 9
CI (se), 10
class, 4, 13
clayloam, 5
confint, 4
din32645, 4, 5
drm, 8, 13
envalysis, 6
findInterval, 2
formula, 13
ggtheme, 14
icp, 4, 6
lm, 3
lod (calibration), 3
loq (calibration), 3
mselect, 8
neitzel2003, 9
plot.calibration (calibration), 3
plot.texture (texture), 12
predict, 4
print.calibration (calibration), 3
print.texture (texture), 12
rlm, 3
rmse, 9
sd, 10
se, 10
signifig, 11
sorption, 11
texture, 5, 12
theme_publish, 14