Package 'chemdeg'

June 25, 2024

Type Package

Title Analysis of Chemical Degradation Kinetic Data

Version 0.1.4

Description A collection of functions that have been developed to assist experimenter in modeling chemical degradation kinetic data. The selection of the appropriate degradation model and parameter estimation is carried out automatically as far as possible and is driven by a rigorous statistical interpretation of the results. The package integrates already available goodness-of-fit statistics for nonlinear models. In addition it allows data fitting with the nonlinear first-order multi-target (FOMT) model.

License GPL (>= 3)

URL https://github.com/migliomatte/chemdeg, https://migliomatte.github.io/chemdeg/

BugReports https://github.com/migliomatte/chemdeg/issues

Depends R (>= 2.10)

Imports graphics, MASS, methods, stats

Suggests knitr, rmarkdown, testthat (>= 3.0.0), tibble

VignetteBuilder knitr

Config/testthat/edition 3

Encoding UTF-8

LazyData true

RoxygenNote 7.3.1

NeedsCompilation no

Author Matteo Migliorini [aut, cre, cph], Roberto Chignola [aut]

Maintainer Matteo Migliorini <matteo.migliorini@univr.it>

Repository CRAN

Date/Publication 2024-06-25 08:30:02 UTC

Contents

`AICC` *Akaike Information Criterion With Correction*

Description

The function calculates the Akaike Information Criterion with correction for small samples size.

Usage

```
AICC(fit)
```

Arguments

`fit`	a 'nls'-object

Details

When the sample size is small, there is a substantial probability that AIC (see `stats::AIC()` for more details) will select models that have too many parameters, i.e. that AIC will overfit. AICc is AIC with a correction for small sample sizes.

The AICc is computed as follows:

$$AICc = AIC + \frac{2\,k\,(k+1)}{n-k-1}$$

where n denotes the sample size and k denotes the number of parameters. Thus , AICc is essentially AIC with an extra penalty term for the number of parameters. Note that as $n \to \infty$, the extra penalty term converges to 0, and thus AICc converges to AIC.

Value

Returns the AICc value

See Also

`stats::AIC()` for uncorrected AIC, `stats::BIC()`, `stats::sigma()` ,`chiquad_red()` for other goodness of fit indicators. `goodness_of_fit()`

Examples

```
t <- seq(0, 10, 1)
y <- 1 / (0.5 * exp(t) + 1) + stats::rnorm(length(t), 0, 0.05)

fit <- nls(y ~ 1 / (k * exp(t) + 1),
  data = list(t = t, y = y),
  start = list(k = 0.2)
)
AICC(fit)
```

---

chiquad_red *Reduced chi-squared*

Description

Function that returns the reduced chi-squared ($\chi^2_{red} = \chi^2/df$, where df are the degrees of freedom) value for a non-linear regression model (`nls` object). Reduced-chi squared is a goodness-of-fit measure. Values close to 1 indicates a good fit, while values $>> 1$ indicate poor fit and values < 1 indicate over-fitting. The function is calculated only with non-linear regression weighted on experimental error.

Usage

```
chiquad_red(fit)
```

Arguments

`fit`	`nls` object with weighted fit

Value

Returns the reduced chi-squared value

References

Philip R. Bevington, D. Keith Robinson, J. Morris Blair, A. John Mallinckrodt, Susan McKay (1993). *Data Reduction and Error Analysis for the Physical Sciences*

See Also

stats::dchisq() for chi-squared distribution; stats::AIC(), stats::BIC(), stats::sigma() (for RMSE), AICC() for other goodness-of-fit indicators. goodness_of_fit()

Examples

```
x <- c(1, 2, 3, 4, 5)
y <- c(1.2, 3.9, 8.6, 17.4, 26)
er <- c(0.5, 0.8, 0.5, 1.9, 1.2)
fit1 <- nls(y ~ k * x^2,
  data = list(x = x, y = y),
  start = list(k = 1),
  weights = 1 / er^2
)
chiquad_red(fit1)

fit2 <- nls(y ~ k * x^3,
  data = list(x = x, y = y),
  start = list(k = 1),
  weights = 1 / er^2
)
chiquad_red(fit2)
```

---

det_order *Determining reaction order and kinetic formula*

Description

The functions seeks to determine the reaction order and kinetic rate constant for chemical models that best fit degradation kinetic data. The input of the function is a data-frame organized as follows:

1. first columns, time data;
2. second columns, concentration data;
3. third column (optional, but highly recommended), experimental error

Usage

```
det_order(dframe)
```

Arguments

dframe	a data-frame with 2 or 3 columns, containing time, concentrations, and (optional) error data.

Value

A `ord_res` object containing in a list the following information:

1. the phase space coordinates of transformed data;
2. the linear regression performed in the phase space;
3. a boolean variable indicating if the estimate of the degradation rate constant is statistically significant;
4. non-linear regression performed using a n^th^-order kinetic model (if n=0 the regression is linear);
5. the data-frame given as the input;
6. the estimated reaction order.

See Also

`results()` to print the results or `goodness_of_fit()` to visualize the major goodness-of-fit measures; `plot_ord()` to plot the regressions in both the phase and conventional spaces; `kin_regr()` to extract the best kinetic model that explain the data and `phase_space()` to extract the linear regression in the phase space.

Examples

```
t <- c(0, 4, 8, 12, 16, 20)
conc <- c(1, 0.51, 0.24, 0.12, 0.07, 0.02)
err <- c(0.02, 0.05, 0.04, 0.04, 0.03, 0.02)
dframe <- data.frame(t, conc)
res <- det_order(dframe)

class(res)



dframe2 <- data.frame(t, conc, err)
res2 <- det_order(dframe2)

res2[[5]] == dframe2
```

FOMT *First-Order Multi-Target model regression*

Description

The function performs a non-linear regression using the first-order multi-target model. The model equation is:

$$\frac{S}{S_0} = 1 - (1 - e^{-k\,t})^m$$

where S/S_0 is the fraction of surviving molecules, k is the average number of hits per time unit, m is the number of hits required to degrade the molecule, and t is time.

Usage

```
FOMT(dtframe)
```

Arguments

dtframe	A data-frame containing 2 or 3 columns: time, normalized concentration and error (optional), respectively

Details

The FOMT model has been proposed as an alternative to the Weibull equation that is commonly used when the time-dependent behavior of the data significantly deviates from that predicted by standard chemical models.

Value

Returns the results of the regression as a nls object.

See Also

FOMTm(), par_est_FOMT()

Examples

```
t <- c(0, 4, 8, 12, 16, 20)
conc <- c(1, 0.98, 0.99, 0.67, 0.12, 0.03)
err <- c(0.02, 0.05, 0.04, 0.04, 0.03, 0.02)
dframe <- data.frame(t, conc, err)
FOMT <- FOMT(dframe)
plot(dframe[[1]], dframe[[2]])
arrows(dframe[[1]], dframe[[2]] + dframe[[3]],
  dframe[[1]], dframe[[2]] - dframe[[3]],
  length = 0
)
newt <- seq(0, 21, by = 0.1)
lines(newt, predict(FOMT, newdata = list(t = newt)))

dframe1 <- data.frame(t, conc)
FOMT1 <- FOMT(dframe1)
plot(dframe1[[1]], dframe1[[2]])
lines(newt, predict(FOMT1, newdata = list(t = newt)))
summary(FOMT)
summary(FOMT1)
```

fomtdata *Total CQA with ascorbic acid (FOMT model)*

Description

Degradation data of 1.2 mM 5-caffeoylquinic acid (5-CQA) in the presence of 1.2 mM of ascorbic acid at 37°C. The data refer to total CQA concentration.

Usage

```
fomtdata
```

Format

A data frame with 8 rows and 2 columns:

time_h Time in hours

tCQA_AA Normalized concentration of total CQA measured at each time point

Source

Yusaku N. and Kuniyo I. (2013) *Degradation Kinetics of Chlorogenic Acid at Various pH Values and Effects of Ascorbic Acid and Epigallocatechin Gallate on Its Stability under Alkaline Conditions*, Journal of Agricultural and Food Chemistry, doi:10.1021/jf304105w, fig 2D, solid diamonds

FOMTm *First-Order Multi-Target model*

Description

Call the function to return the formula of the Single-Hit Multi-Target model (FOMT):

$$1 - (1 - e^{-k\,t})^n$$

Usage

```
FOMTm(t, k, m)
```

Arguments

t	time
k	average number of hits per time unit
m	minimum number of hits required to degrade the molecule

Value

Returns calculated values using the formula of FOMT model

It can be used inside the function nls as the RHS of the formula.

See Also

FOMT(), par_est_FOMT(), stats::nls()

Examples

```
t <- seq(0, 100, by = 1)
k <- 0.1
n <- 200
y <- FOMTm(t, k, n)
plot(t, y, type = "l")
```

---

f_gen *Formula of an n-order model.*

Description

Given the reaction order n , the function returns the equation corresponding to that particular n^th^-order kinetic model. For $n \neq 1$:

$$y(t) = ((n-1)\,k\,t + y_0^{1-n}))^{\frac{1}{n-1}}$$

for $n = 1$:

$$y(t) = y_0\, e^{-k\,t}$$

Usage

```
f_gen(n)
```

Arguments

n	reaction order

Value

A formula object containing the equation of the selected *n*^th^ order kinetic model.

Examples

```
nc <- 2
f_gen(nc)

f_gen(1)
```

goodness_of_fit *Goodness-of-fit, non-linear regression*

Description

Function that returns the following goodness-of-fit statistics for non-linear regression: AIC, AICc, BIC, RMSE and reduced Chi-squared.

Usage

```
goodness_of_fit(fit)
```

Arguments

fit	a nls, lm or ord_res object

Details

The function returns the values of AIC, AICC, BIC, RMSE and reduced chi-squared (χ^2_{red}) for nls objects. If a linear model object is passed, the function returns its summary.

Given an ord_res object (output of the function det_order()), the function returns one of the results above depending on the model chosen to explain the data.

Because the chiquad_red() function returns the value only with weighted data, the χ^2_{red} will be returned only with weighted regressions.

Value

It returns a table with the values of AIC, AICc, BIC, RSME and reduced Chi squared. Single goodness-of-fit measures can be obtained as follows:

1. call standard R functions stats::AIC(), stats::BIC(), stats::sigma() for AIC, BIC and RMSE, respectively;
2. call chemdeg functions AICC() and chiquad_red() for AICc and reduced chi-squared, respectively.

See Also

stats::AIC(), AICC(), stats::BIC(), stats::sigma(), chiquad_red()

Examples

```
x <- c(1, 2, 3, 4, 5)
y <- c(1.2, 3.9, 8.6, 17.4, 26)
er <- c(0.5, 0.8, 0.5, 1.9, 1.2)
fit1 <- nls(y ~ k * x^2,
  data = list(x = x, y = y), start = list(k = 1),
  weights = 1 / er^2
)
goodness_of_fit(fit1)
```

kin_regr *Degradation kinetics*

Description

Returns from an ord_res object either the linear or the non-linear regression of the degradation kinetics data.

Usage

```
kin_regr(x)
```

Arguments

x	an ord_res object

Details

After the analysis in the phase space for the determination of the reaction order, det_order() performs either a linear or a non-linear regression of the kinetic data, depending on whether the reaction order is n=0 or n>0, respectively. To access the regression object call kin_degr.

Value

Returns either an nls or lm object based on the regression performed by the function det_order().

See Also

det_order(), phase_space(), results(), stats::lm()

Examples

```
t <- c(0, 4, 8, 12, 16, 20)
conc <- c(1, 0.51, 0.24, 0.12, 0.07, 0.02)
dframe <- data.frame(t, conc)
res <- det_order(dframe)

kin_regr(res)
```

ord1 *First order kinetic data*

Description

Synthetic data from a first-order kinetic model with k=0.7

Usage

```
ord1
```

Format

A data frame with 6 rows and 3 columns:

t time

concentration simulated concentration data at each time point

std.error simulated experimental error

par_est_FOMT *First-Order Multi-Target parameter starting values*

Description

`par_est_FOMT` estimates the starting values of the parameters of the first-order multi-target model from a data-set.

Usage

```
par_est_FOMT(x, y = NULL)
```

Arguments

x	A data-frame with time and concentration in the first and second columns,respectively. Alternatively, it could be an array of time and y an array of concentrations.
y	Optional, an array of concentrations. To be inserted only if x is an array.

Value

The function returns an array with the suggested initial values of parameters.

See Also

`FOMT()`, `FOMTm()`

Examples

```
t <- seq(0, 30, by = 6)
k <- 0.3
n <- 40
set.seed(100)
y <- FOMTm(t, k, n) * (1 + rnorm(length(t), 0, 0.05))

nlsFOMT <- nls(y ~ FOMTm(t, k, n),
  data = list(y = y, t = t),
  start = par_est_FOMT(t, y)
)
summary(nlsFOMT)
```

---

phase_space *Phase space, linearized model*

Description

Given an ord_res object, this function returns the linearized model that best fits the data in the phase space. ord_res object can be obtained using the function det_order().

Usage

```
phase_space(x)
```

Arguments

x	an ord_res object

Value

Returns a lm class object.

See Also

det_order(), kin_regr(), results(), stats::lm()

Examples

```
t <- c(0, 4, 8, 12, 16, 20)
conc <- c(1, 0.51, 0.24, 0.12, 0.07, 0.02)
dframe <- data.frame(t, conc)
res <- det_order(dframe)

phase_space(res)
```

plot_ord *Plot of phase space and kinetic curve*

Description

The function plots the results obtained from a `det_order()` function. Two plots are shown: one representing the transformed data in the phase space and the other the kinetic data in the conventional space along with their regression curves.

Usage

```
plot_ord(ord_res)
```

Arguments

`ord_res`	an 'ord_res' object

Value

Two plots. The first representing the transformed data in the phase space and the other the kinetic data in the conventional space along with their regression curves. Black line represent the best regression curve, whereas green lines show the fits with the reaction order chosen.

Examples

```
t <- c(0, 4, 8, 12, 16, 20)
conc <- c(1, 0.51, 0.24, 0.12, 0.07, 0.02)
dframe <- data.frame(t, conc)
res <- det_order(dframe)

plot_ord(res)
```

results *Summary of 'ord_res' object*

Description

Returns the results of the analyses performed by `det_order()` function.

Usage

```
results(object)
```

Arguments

`object`	an 'ord_res' object

Details

The function prints:

1. the linear regression performed in the phase space, together with the estimated *n* value and its 95% confidence interval
2. a brief conclusion on the results obtained in the phase space stating which reaction order should be preferred
3. the (non-)linear regression performed with parameters associated statistics. If a non-linear regression has been performed, the most common goodness-of-fit measures calculated with `goodness_of_fit()` are printed

Value

It prints a summary of the analysis in the phase space, the reaction order, and the regression results.

See Also

`det_order()`, `kin_regr()`, `phase_space()`

Examples

```
t <- c(0, 4, 8, 12, 16, 20)
conc <- c(1, 0.51, 0.24, 0.12, 0.07, 0.02)
err <- c(0.02, 0.05, 0.04, 0.04, 0.03, 0.02)
dframe <- data.frame(t, conc, err)
res <- det_order(dframe)

results(res)
```

---

urfa *Urfa pepper ascorbic acid degradation (2-nd order)*

Description

Data describing the degradation kinetics of ascorbic acid during dehydration of Urfa peppers. The peppers were treated with hot air at 55, 65 and 75 °C.

Usage

```
urfa
```

Format

A data frame with 8 rows and 4 columns:

time_min time in minutes

AA_55 normalized concentration of ascorbic acid of Urfa peppers dehydrated at 55°C

AA_65 normalized concentration of ascorbic acid of Urfa peppers dehydrated at 65°C

AA_75 normalized concentration of ascorbic acid of Urfa peppers dehydrated at 75°C

Source

Ş. Dağhan, A. Yildirim, F. Mehmet Yilmaz, H. Vardin and M. Karaaslan (2018) *The effect of temperature and method of drying on isot (Urfa pepper) and its Vitamin C degradation kinetics*, Italian Journal of Food Science, doi:10.14674/IJFS1070, fig 5, hot-air

Index