Package ‘nlsMicrobio’

September 17, 2021

Version 0.0-3
Title Nonlinear Regression in Predictive Microbiology
Depends stats, nlstools
Suggests knitr
Description Data sets and nonlinear regression models dedicated to predictive microbiology.
License GPL-3
LazyLoad yes
LazyData yes
NeedsCompilation no
VignetteBuilder knitr
BuildVignettes true
Author Florent Baty [aut],
Marie-Laure Delignette-Muller [aut],
Aurelie Siberchicot [aut, cre]
Maintainer Aurelie Siberchicot <aurelie.siberchicot@univ-lyon1.fr>
Repository CRAN
Date/Publication 2021-09-17 11:30:02 UTC

R topics documented:

<table>
<thead>
<tr>
<th>R topics</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>competitioncurve</td>
<td>2</td>
</tr>
<tr>
<td>competitionmodels</td>
<td>3</td>
</tr>
<tr>
<td>growthcurve</td>
<td>5</td>
</tr>
<tr>
<td>growthmodels</td>
<td>6</td>
</tr>
<tr>
<td>ross</td>
<td>8</td>
</tr>
<tr>
<td>secondary</td>
<td>9</td>
</tr>
<tr>
<td>survivalcurve</td>
<td>11</td>
</tr>
<tr>
<td>survivalmodels</td>
<td>12</td>
</tr>
</tbody>
</table>

Index 15
**Growth kinetics of two competitive bacterial flora**

**Description**

Kinetics of simultaneous growth of Escherichia coli O157:H7 and ground beef background microflora in enrichment broth

**Usage**

```r
data(competition1)
data(competition2)
```

**Format**

Data frames with 3 columns (t: time, flora: 1 for the first flora and 2 for the second one, LOG10N: decimal logarithm of bacterial density)

**Source**

Two of the kinetics used in Vimont et al. (2006)

**References**


**Examples**

```r
data(competition1)
data(competition2)
def.par <- par(no.readonly = TRUE)
par(mfrow = c(1,2))
twocolors <- c("red","blue")
plot(competition1$t,competition1$LOG10N,col=twocolors[competition1$flora])
plot(competition2$t,competition2$LOG10N,col=twocolors[competition2$flora])
par(def.par)
```
Description

Formulas of primary growth models used in predictive microbiology to model the simultaneous growth of two competitive bacterial flora assuming a Jameson effect.

Usage

  jason_buchanan
  jason_baranyi
  jason_without_lag

Details

These models describe the simultaneous evolution of the decimal logarithm of the microbial counts of two flora (LOG10N) as a function of the time (t) and of the flora (flora) coded as 1 for counts of flora 1 and 2 for counts of flora 2. These three models assume independent lag and growth parameters for flora 1 and 2, except for the saturation which is supposed to be governed by the Jameson effect and modelled by a common parameter (tmax) which represents the time at which both flora stop to multiply. Modelling the simultaneous saturation by this way enables the model to be fitted by nls, as an analytical form of the model is available.

jason_buchanan is based on the model of Buchanan et al. (1997) for lag phase modelling and is characterized by seven parameters (LOG10N0_1, mumax_1, lag_1, LOG10N0_2, mumax_2, lag_2 and the common saturation time tmax). This model was described and used in Vimont et al. (2006).

jason_baranyi is based on the model of Baranyi and Roberts (1994) for lag phase modelling and is characterized by seven parameters (LOG10N0_1, mumax_1, lag_1, LOG10N0_2, mumax_2, lag_2 and the common saturation time tmax).

jason_without_lag is based on the exponential model without lag phase and is thus characterized by five parameters (LOG10N0_1, mumax_1, LOG10N0_2, mumax_2 and the common saturation time tmax).

Value

A formula

Author(s)

Florent Baty, Marie-Laure Delignette-Muller
References


Examples

```r
options(digits = 3)

### Example 1: fit of model jameson_buchanan
data(competition1)
nls1 <- nls(jameson_buchanan, competition1, 
            list(lag_1 = 2, mumax_1 = 1, LOG10N0_1 = 1, tmax = 12, 
                 lag_2 = 2, mumax_2 = 1, LOG10N0_2 = 4))
overview(nls1)

# Plot of theoretical curves with data
twocolors <- c("red","blue")
npoints <- 100
seq.t <- seq(0,max(competition1$t),length.out=npoints)
prednls1.1 <- predict(nls1,data.frame(t=seq.t,flora=rep(1,npoints)))
prednls1.2 <- predict(nls1,data.frame(t=seq.t,flora=rep(2,npoints)))
plot(competition1$t,competition1$LOG10N,col=twocolors[competition1$flora],xlab="t",ylab="LOG10N")
lines(seq.t,prednls1.1,col=twocolors[1])
lines(seq.t,prednls1.2,col=twocolors[2])

### Example 2 : fit of model jameson_baranyi
data(competition1)
nls2 <- nls(jameson_baranyi, competition1, 
            list(lag_1 = 2, mumax_1 = 1, LOG10N0_1 = 1, tmax = 12, 
                 lag_2 = 2, mumax_2 = 1, LOG10N0_2 = 4))
overview(nls2)
plotfit(nls2)

# Plot of theoretical curves with data
twocolors <- c("red","blue")
npoints <- 100
seq.t <- seq(0,max(competition1$t),length.out=npoints)
prednls2.1 <- predict(nls2,data.frame(t=seq.t,flora=rep(1,npoints)))
prednls2.2 <- predict(nls2,data.frame(t=seq.t,flora=rep(2,npoints)))
```

### Example 3: fit of model jameson_without_lag

data(competition2)
nls3 <- nls(jameson_without_lag, competition2, 
                    list(mumax_1 = 1, LOG10N0_1 = 1, tmax = 12, 
                         mumax_2 = 1, LOG10N0_2 = 4))

overview(nls3)
plotfit(nls3)

# Plot of theoretical curves with data
twocolors <- c("red","blue")
npoints <- 100
seq.t <- seq(0,max(competition2$t),length.out=npoints)
prednls3.1 <- predict(nls3, data.frame(t=seq.t,flora=rep(1,npoints)))
prednls3.2 <- predict(nls3, data.frame(t=seq.t,flora=rep(2,npoints)))
plot(competition2$t,competition2$LOG10N,col=twocolors[competition2$flora],xlab="t",ylab="LOG10N")
lines(seq.t,prednls3.1,col=twocolors[1])
lines(seq.t,prednls3.2,col=twocolors[2])

---

growthcurve  

**Bacterial kinetics data sets**

**Description**

Bacterial kinetics data sets

**Usage**

```
data(growthcurve1)
data(growthcurve2)
data(growthcurve3)
data(growthcurve4)
```

**Format**

Data frames with 2 columns (t: time, LOG10N: decimal logarithm of bacterial density)

**Source**

Data obtained by Florent Baty and Marie-Laure Delignette-Muller
growthmodels

Examples

data(growthcurve1)
data(growthcurve2)
data(growthcurve3)
data(growthcurve4)
def.par <- par(no.readonly = TRUE)
par(mfrow = c(2,2))
plot(growthcurve1)
plot(growthcurve2)
plot(growthcurve3)
plot(growthcurve4)
par(def.par)

growthmodels  Bacterial growth models

Description

Formulas of primary growth models commonly used in predictive microbiology

Usage

baranyi
baranyi_without_Nmax
baranyi_without_lag
buchanan
buchanan_without_Nmax
buchanan_without_lag
gompertz

Details

These models describe the evolution of the decimal logarithm of the microbial count (LOG10N) as a function of the time (t).

baranyi is the model of Baranyi and Roberts (1994) with four parameters (LOG10N0, mumax, lag, LOG10Nmax)

baranyi_without_Nmax is the model of Baranyi and Roberts (1994) with three parameters (LOG10N0, mumax, lag), without braking

baranyi_without_lag is the model of Baranyi and Roberts (1994) with three parameters (LOG10N0, mumax, LOG10Nmax), without lag

buchanan is the three-phase linear model proposed by Buchanan et al. (1997)

buchanan_without_Nmax is the two-phase linear model with three parameters (LOG10N0, mumax, lag), without braking
buchanan_without_lag is the two-phase linear model with three parameters (LOG10N0, mumax, LOG10Nmax), without lag

gompertz_m is the modified Gompertz model introduced by Gibson et al. (1988) and reparameterized by Zwietering et al. (1990)

Value
A formula

Author(s)
Florent Baty, Marie-Laure Delignette-Muller

References


Examples

# Example 1

data(growthcurve1)
nls1 <- nls(baranyi, growthcurve1,
 list(lag=4, mumax=1, LOG10N0 = 4, LOG10Nmax = 9))
nls2 <- nls(gompertz_m, growthcurve1,
 list(lag = 4, mumax = 1, LOG10N0 = 4, LOG10Nmax = 9))
nls3 <- nls(buchanan, growthcurve1,
 list(lag = 4, mumax = 1, LOG10N0 = 4, LOG10Nmax = 9))
def.par <- par(no.readonly = TRUE)
par(mfrow = c(2,2))
plotfit(nls1, smooth = TRUE)
plotfit(nls2, smooth = TRUE)
plotfit(nls3, smooth = TRUE)
par(def.par)
# Example 2

data(growthcurve2)
nls4 <- nls(baranyi_without_Nmax, growthcurve2, list(lag = 2, mumax = 0.4, LOG10N0 = 7.4))
nls5 <- nls(buchanan_without_Nmax, growthcurve2, list(lag = 2, mumax = 0.4, LOG10N0 = 7.4))
def.par <- par(no.readonly = TRUE)
par(mfrow = c(2,1))
plotfit(nls4, smooth = TRUE)
plotfit(nls5, smooth = TRUE)
par(def.par)

# Example 3

data(growthcurve3)
nls6 <- nls(baranyi_without_lag, growthcurve3, list(mumax = 1, LOG10N0 = 0, LOG10Nmax = 5))
nls7 <- nls(buchanan_without_lag, growthcurve3, list(mumax = 1, LOG10N0 = 0, LOG10Nmax = 5))
def.par <- par(no.readonly = TRUE)
par(mfrow = c(2,1))
plotfit(nls6, smooth = TRUE)
plotfit(nls7, smooth = TRUE)
par(def.par)

---

ross | Secondary growth curves

**Description**

A data frames describing the specific growth rate of *Escherichia coli* as a function of various environmental factors

**Usage**

data(ross)

**Format**

A data frame with five columns (author: the author of the paper from which the data was extracted, T: the temperature in Celsius, aw: the water activity, pH: the pH value, sqrtmumax: the square root of the maximum specific growth rate)

**Source**

Examples

data(ross)
def.par <- par(no.readonly = TRUE)
par(mfrow = c(2, 2))
plot(ross[c("T", "sqrtmumax")])
plot(ross[c("pH", "sqrtmumax")])
plot(ross[c("aw", "sqrtmumax")])
par(def.par)

Description

Formulas of secondary growth models commonly used in predictive microbiology

Usage

cpm_T
cpm_pH_4p
cpm_pH_3p
cpm_aw_3p
cpm_aw_2p
cpm_T_pH_aw

details

All the models describe the evolution of the square root of the maximum specific growth rate (sqrt-
mumax) as a function of one or more environmental factors among temperature (T), pH (pH) and
water activity (aw). Each model must be fitted to a data frame including at least two columns, the
last one named "sqrtmumax" and the first ones named "T", "pH" or "aw" according to the model.
cpm_T is the cardinal temperature model with inflection (Rosso et al., 1993) with four parameters
(Tmin, Topt, Tmax, muopt)
cpm_pH_4p is the cardinal pH model (Rosso et al., 1995) with four parameters (pHmin, pHopt, pHmax, muopt)
cpm_pH_3p is a symmetric cardinal pH model with three parameters (pHmin, pHopt, muopt), ob-
tained by fixing pHmax to 2pHopt-pHmin in the cpm_pH_4p model
cpm_aw_3p is the cardinal aw model (Rosso and Robinson, 2001) with three parameters (awmin, awopt, muopt)
cpm_aw_2p is a simplified cardinal aw model (Rosso and Robinson, 2001) with two parameters
(awmin, muopt) obtained by fixing awopt to 1 in the cpm_aw_3p model
cpm_T_pH_aw is the cardinal model based on the gamma concept (Pinon et al., 2004) with 9 parameters (Tmin, Topt, Tmax, pHmin, pHopt, pHmax, awmin, awopt, muopt)

Value
A formula

Author(s)
Florent Baty, Marie-Laure Delignette-Muller

References


Examples

data(ross)

# Example for the cpm_T model
d1 <- subset(ross, author == "salter" & aw == 0.997, select = c(T, sqrtmumax))
nls1 <- nls(cpm_T, d1, list(muopt = 1.7, Tmin = 4, Topt = 40, Tmax = 47))
plotfit(nls1, smooth = TRUE)
overview(nls1)

# Example for the cpm_pH_4p model

d2 <- subset(ross, author == "presser" & aw > 0.99, select = c(pH, sqrtmumax))
nls2 <- nls(cpm_pH_4p, d2, list(muopt = 0.5, pHmin = 4, pHopt = 6.5, pHmax = 9))
plotfit(nls2, smooth = TRUE)
overview(nls2)
# Example for the cpm_pH_3p model

d3 <- subset(ross, author == "presser" & aw == 0.997,
select = c(pH, sqrtmumax))
nls3 <- nls(cpm_pH_3p, d3, list(muopt = 0.5, pHmin = 4, pHopt = 6.5))
plotfit(nls3, smooth = TRUE)
overview(nls3)

# Example for the cpm_aw_3p model

d4 <- subset(ross, author == "mellefont", select = c(aw, sqrtmumax))
nls4 <- nls(cpm_aw_3p, d4, list(muopt = 0.6, awmin = 0.95, awopt = 0.995))
plotfit(nls4, smooth = TRUE)
overview(nls4)

# Example for the cpm_aw_2p model

d5 <- subset(ross, author == "mellefont", select = c(aw, sqrtmumax))
nls5 <- nls(cpm_aw_2p, d5, list(muopt = 0.6, awmin = 0.95))
plotfit(nls5, smooth = TRUE)
overview(nls5)

# Examples for the cpm_T_pH_aw model

d6 <- subset(ross, select = c(T, pH, aw, sqrtmumax))
nls6 <- nls(cpm_T_pH_aw, d6, list(muopt = 2, Tmin = 4, Topt = 40, Tmax = 49,
pHmin = 4, pHopt = 6.5, pHmax = 9, awmin = 0.95, awopt = 0.995))
def.par <- par(no.readonly = TRUE)
par(mfrow = c(2, 2))
plotfit(nls6, variable = 1)
plotfit(nls6, variable = 2)
plotfit(nls6, variable = 3)
overview(nls6)
par(def.par)

---

Bacterial survival data sets

**Description**

Bacterial kinetics data sets

**Usage**

data(survivalcurve1)
data(survivalcurve2)
data(survivalcurve3)
survivalmodels

Format

Data frames with 2 columns (t: time, LOG10N: decimal logarithm of bacterial density)

Source

Data obtained by Florent Baty and Marie-Laure Delignette-Muller

Examples

```r
data(survivalcurve1)
data(survivalcurve2)
data(survivalcurve3)
def.par <- par(no.readonly = TRUE)
par(mfrow=c(2,2))
plot(survivalcurve1, type="b")
plot(survivalcurve2, type="b")
plot(survivalcurve3, type="b")
par(def.par)
```

survivalmodels Bacterial survival models

Description

Formulas of primary survival models commonly used in predictive microbiology

Usage

```r
geeraerd
geeraerd_without_Nres
geeraerd_without_Sl
mafart
albert
trilinear
bilinear_without_Nres
bilinear_without_Sl
```

Details

These models describe the evolution of the decimal logarithm of the microbial count (LOG10N) as a function of the time (t).

g**eeraerd** is the model of Geeraerd et al. (2005) with four parameters (LOG10N0, kmax, Sl, LOG10Nres)

g**eeraerd_without_Nres** is the model of of Geeraerd et al. (2005) with three parameters (LOG10N0, kmax, Sl), without tail

g**eeraerd_without_Sl** is the model of of Geeraerd et al. (2005) with three parameters (LOG10N0,
kmax, Nres), without shoulder

mafart is the Weibull model as parameterized by Mafart et al. (2002) with three parameters (p, delta, LOG10N0)

albert is the modified Weibull model proposed by Albert and Mafart (2005) with four parameters (p, delta, LOG10N0, LOG10Nres)

trilinear is the three-phase linear model with four parameters (LOG10N0, kmax, SI, LOG10Nres)

bilinear_without_Nres is the two-phase linear model with three parameters (LOG10N0, kmax, SI), without tail

bilinear_without_SI is the two-phase linear model with three parameters (LOG10N0, kmax, LOG10Nres), without shoulder

Value

A formula

Author(s)

Florent Baty, Marie-Laure Delignette-Muller

References


Examples

# Example 1

data(survivalcurve1)
nls1a <- nls(geeraerd, survivalcurve1, 
list(Sl = 5, kmax = 1.5, LOG10N0 = 7, LOG10Nres = 1))

nls1b <- nls(trilinear, survivalcurve1, 
list(Sl = 5, kmax = 1.5, LOG10N0 = 7, LOG10Nres = 1))

nls1c <- nls(albert, survivalcurve1, 
list(p = 1.2, delta = 4, LOG10N0 = 7, LOG10Nres = 1))
def.par <- par(no.readonly = TRUE)
par(mfrow = c(2,2))
survivalmodels

overview(nls1a)
plotfit(nls1a, smooth = TRUE)
overview(nls1b)
plotfit(nls1b, smooth = TRUE)
overview(nls1c)
plotfit(nls1c, smooth = TRUE)
par(def.par)

# Example 2

data(survivalcurve2)
nls2a <- nls(geeraerd_without_Nres, survivalcurve2,
list(Sl = 10, kmax = 1.5, LOG10N0 = 7.5))
nls2b <- nls(bilinear_without_Nres, survivalcurve2,
list(Sl = 10, kmax = 1.5, LOG10N0 = 7.5))
nls2c <- nls(mafart, survivalcurve2,
list(p = 1.5, delta = 8, LOG10N0 = 7.5))
def.par <- par(no.readonly = TRUE)
par(mfrow = c(2,2))
overview(nls2a)
plotfit(nls2a, smooth = TRUE)
overview(nls2b)
plotfit(nls2b, smooth = TRUE)
overview(nls2c)
plotfit(nls2c, smooth = TRUE)
par(def.par)

# Example 3

data(survivalcurve3)
nls3a <- nls(geeraerd_without_Sl, survivalcurve3,
list(kmax = 4, LOG10N0 = 7.5, LOG10Nres = 1))
nls3b <- nls(bilinear_without_Sl, survivalcurve3,
list(kmax = 4, LOG10N0 = 7.5, LOG10Nres = 1))
nls3c <- nls(mafart, survivalcurve3,
list(p = 0.5, delta = 0.2, LOG10N0 = 7.5))
def.par <- par(no.readonly = TRUE)
par(mfrow = c(2,2))
overview(nls3a)
plotfit(nls3a, smooth = TRUE)
overview(nls3b)
plotfit(nls3b, smooth = TRUE)
overview(nls3c)
plotfit(nls3c, smooth = TRUE)
par(def.par)
Index

* datasets
  - competitioncurve, 2
  - growthcurve, 5
  - ross, 8
  - survivalcurve, 11

* models
  - competitionmodels, 3
  - growthmodels, 6
  - secondary, 9
  - survivalmodels, 12

  albert (survivalmodels), 12
  baranyi (growthmodels), 6
  baranyi_without_lag (growthmodels), 6
  baranyi_without_Nmax (growthmodels), 6
  bilinear_without_Nres (survivalmodels), 12
  bilinear_without_Sl (survivalmodels), 12
  buchanan (growthmodels), 6
  buchanan_without_lag (growthmodels), 6
  buchanan_without_Nmax (growthmodels), 6
  competition1 (competitioncurve), 2
  competition2 (competitioncurve), 2
  competitioncurve, 2
  competitionmodels, 3
  cpm_aw_2p (secondary), 9
  cpm_aw_3p (secondary), 9
  cpm_pH_3p (secondary), 9
  cpm_pH_4p (secondary), 9
  cpm_T (secondary), 9
  cpm_T_pH_aw (secondary), 9
  geeraerd (survivalmodels), 12
  geeraerd_without_Nres (survivalmodels), 12
  mafart (survivalmodels), 12
  mafart (survivalmodels), 12
  ross, 8
  secondary, 9
  survivalcurve, 11
  survivalcurve1 (survivalcurve), 11
  survivalcurve2 (survivalcurve), 11
  survivalcurve3 (survivalcurve), 11
  survivalmodels, 12
  trilinear (survivalmodels), 12
  growthcurve1 (growthcurve), 5
  growthcurve2 (growthcurve), 5
  growthcurve3 (growthcurve), 5
  growthcurve4 (growthcurve), 5
  growthmodels, 6
  jameson_baranyi (competitionmodels), 3
  jameson_buchanan (competitionmodels), 3
  jameson_without_lag (competitionmodels), 3
  mafart (survivalmodels), 12
  ross, 8
  secondary, 9
  survivalcurve, 11
  survivalcurve1 (survivalcurve), 11
  survivalcurve2 (survivalcurve), 11
  survivalcurve3 (survivalcurve), 11
  survivalmodels, 12
  trilinear (survivalmodels), 12
  growthcurve, 5