Package 'growthcurver'

October 13, 2022

Type Package

Title Simple Metrics to Summarize Growth Curves

Version 0.3.1

Date 2020-10-3

Description Fits the logistic equation to
microbial growth curve data (e.g., repeated absorbance measurements
taken from a plate reader over time). From this fit, a variety of
metrics are provided, including the maximum growth rate,
the doubling time, the carrying capacity, the area under the logistic
curve, and the time to the inflection point. Method described in
Sprouffske and Wagner (2016) <doi:10.1186/s12859-016-1016-7>.

LazyData TRUE

Depends R (>= 4.0)

Imports minpack.lm (>= 1.2), stats (>= 4.0), graphics (>= 4.0),
grDevices (>= 4.0)

URL https://github.com/sprouffske/growthcurver

BugReports https://github.com/sprouffske/growthcurver/issues

License GPL (>= 2)

Suggests testthat, knitr, dplyr, ggplot2, rmarkdown

VignetteBuilder knitr

RoxygenNote 7.1.1

NeedsCompilation no

Author Kathleen Sprouffske [aut, cre]

Maintainer Kathleen Sprouffske <sprouffske@gmail.com>

Repository CRAN

Date/Publication 2020-10-19 23:00:06 UTC

R topics documented:

---

gcfit   *Creates an object of class gcfit.*

Description

This is a constructor function for the "gcfit" class. This class is most often obtained as the return value when calling SummarizeGrowth.

Usage

```
gcfit(gc_vals, log_mod, data_t, data_n)
```

Arguments

gc_vals   An object of class gcvals that contains the summarized metrics from fitting the
                growth model to a set of experimental observations. This is where the fitness
                proxy parameters can be found. See gcvals for more information the informa-
                tion found in this object.

log_mod   An object of class nlsModel that contains the results of fitting the logistic growth
                model to the data

data_t   A numeric vector of times

data_n   A numeric vector of cell count or absorbance readings

Value

An object of class gcfit, which is a list of three objects, that combines the parameters (vals = gc_vals, model = log_mod, data = list(data_t, data_n))

---

gcvals	*Creates an object of type gcvals.*

Description

Constructor function for the "gcvals" class. This object is most often obtained when calling `SummarizeGrowth` (it is the first parameter in the `gcvals` object).

Usage

```
gcvals(
  k,
  k_se,
  k_p,
  n0,
  n0_se,
  n0_p,
  r,
  r_se,
  r_p,
  sigma,
  df,
  t_mid,
  dt,
  auc_l,
  auc_e,
  note
)
```

Arguments

k	The carrying capacity parameter
k_se	The standard error of the carrying capacity parameter
k_p	The p value of the carrying capacity parameter
n0	The initial population size
n0_se	The standard error of the initial population size
n0_p	The p value of the initial population size
r	The growth rate
r_se	The standard error of the growth rate
r_p	The p value of the growthrate
sigma	Residual standard error from non-linear least squares fit of the model to the data
df	Degrees of freedom
t_mid	The time at the inflection point of the logistic curve (occurs at half of the carrying capacity)

dt	The maximum doubling time, obtained by evaluating the the unrestrained growth of the population with growth rate r
auc_l	The area under the curve of the fitted logistic equation from time 0 to time t
auc_e	The area under the curve of the measurements.
note	Feedback on common problems with fitting the logistic curve to the data

Value

An object of class gcvals.

---

growthdata   *Simulated growth curve data*

Description

A dataset containing absorbance measurements over time of microbes growing in a plate reader for 1 day. The growth curves for a whole plate are included.

Usage

growthdata

Format

A data frame with 145 observations and 97 variables:

time time, in hours

A1 absorbance readings of well A1

A2 absorbance readings of well A2

A3 absorbance readings of well A3

A4 absorbance readings of well A4

A5 absorbance readings of well A5

A6 absorbance readings of well A6

A7 absorbance readings of well A7

A8 absorbance readings of well A8

A9 absorbance readings of well A9

A10 absorbance readings of well A10

A11 absorbance readings of well A11

A12 absorbance readings of well A12

B1 absorbance readings of well B1

B2 absorbance readings of well B2

B3 absorbance readings of well B3

B4 absorbance readings of well B4

B5 absorbance readings of well B5

B6 absorbance readings of well B6

B7 absorbance readings of well B7

B8 absorbance readings of well B8

B9 absorbance readings of well B9

B10 absorbance readings of well B10

B11 absorbance readings of well B11

B12 absorbance readings of well B12

C1 absorbance readings of well C1

C2 absorbance readings of well C2

C3 absorbance readings of well C3

C4 absorbance readings of well C4

C5 absorbance readings of well C5

C6 absorbance readings of well C6

C7 absorbance readings of well C7

C8 absorbance readings of well C8

C9 absorbance readings of well C9

C10 absorbance readings of well C10

C11 absorbance readings of well C11

C12 absorbance readings of well C12

D1 absorbance readings of well D1

D2 absorbance readings of well D2

D3 absorbance readings of well D3

D4 absorbance readings of well D4

D5 absorbance readings of well D5

D6 absorbance readings of well D6

D7 absorbance readings of well D7

D8 absorbance readings of well D8

D9 absorbance readings of well D9

D10 absorbance readings of well D10

D11 absorbance readings of well D11

D12 absorbance readings of well D12

E1 absorbance readings of well E1

E2 absorbance readings of well E2

E3 absorbance readings of well E3

E4 absorbance readings of well E4

E5 absorbance readings of well E5

E6 absorbance readings of well E6

E7 absorbance readings of well E7

E8 absorbance readings of well E8

E9 absorbance readings of well E9

E10 absorbance readings of well E10

E11 absorbance readings of well E11

E12 absorbance readings of well E12

F1 absorbance readings of well F1

F2 absorbance readings of well F2

F3 absorbance readings of well F3

F4 absorbance readings of well F4

F5 absorbance readings of well F5

F6 absorbance readings of well F6

F7 absorbance readings of well F7

F8 absorbance readings of well F8

F9 absorbance readings of well F9

F10 absorbance readings of well F10

F11 absorbance readings of well F11

F12 absorbance readings of well F12

G1 absorbance readings of well G1

G2 absorbance readings of well G2

G3 absorbance readings of well G3

G4 absorbance readings of well G4

G5 absorbance readings of well G5

G6 absorbance readings of well G6

G7 absorbance readings of well G7

G8 absorbance readings of well G8

G9 absorbance readings of well G9

G10 absorbance readings of well G10

G11 absorbance readings of well G11

G12 absorbance readings of well G12

H1 absorbance readings of well H1

H2 absorbance readings of well H2

H3 absorbance readings of well H3

H4 absorbance readings of well H4

H5 absorbance readings of well H5

H6 absorbance readings of well H6

H7 absorbance readings of well H7

H8 absorbance readings of well H8

H9 absorbance readings of well H9

H10 absorbance readings of well H10

H11 absorbance readings of well H11

H12 absorbance readings of well H12

NAtT   *Number of Cells at Time t*

Description

This function gives the number of cells or absorbance (N) at time t when the parameters to the logistic equation are K, N0, and r.

Usage

```
NAtT(k, n0, r, t)
```

Arguments

k	The carrying capacity
n0	The initial population size (absorbance or individuals)
r	The exponential "growth rate"
t	The time at which you want to know N

Value

The number of cells, or N, at time t

SummarizeGrowth   *Summarize Growth Curves*

Description

This function finds the parameters that describe the input data's growth. It does so by fitting the logistic curve to your growth curve measurements.

Usage

```
SummarizeGrowth(data_t, data_n, t_trim = 0, bg_correct = "min", blank = NA)
```

Arguments

`data_t`	A vector of timepoints (data_n must also be provided and be the same length).
`data_n`	A vector of cell counts or absorbance readings.
`t_trim`	Measurements taken after this time should not be included in fitting the curve. If stationary phase is variable, this may give you a better fit. A value of 0 means no trimming. Defaults to no trimming (0).
`bg_correct`	The background correction method to use. No background correction is performed for the default "none". Specifying "min" subtracts the smallest value in a column from all the rows in that column, and specifying "blank" subtracts the values from the blank vector from the data_n vector.
`blank`	A vector of absorbance readings from a blank well (typically contains only media) used for background correction. The corresponding blank value is subtracted from the data_n vector for each timepoint. Defaults to NA.

Details

The logistic curve equation is

$$N_t = \frac{N_0 K}{N_0 + (K - N_0)e^{-rt}}$$

where N_t is the number of cells (or the absorbance reading) at time t, N_0 is the initial cell count (or absorbance reading), K is the carrying capacity, and r is the growth rate.

The fitness proxies returned are the parameters of the logistic equation and the area under the curve (a measure that integrates the effects of N_0, K, and r). See `gcfit` for more documentation on these.

Value

An object of type gcfit containing the "fitness" proxies, as well as the input data and the fitted model.

See Also

See the accompanying Vignette for an example of how to use and interpret SummarizeGrowth. https://CRAN.R-project.org/package=growthcurver/vignettes/Growthcurver-vignette. html.

See also `gcfit`.

Examples

```
# We can check that the parameters that are found are the same
# as we use to generate fake experimental data. To do so, let's first
# generate the "experimental" data using the logistic equation,
# e.g., absorbance readings from a single well in a plate reader over time.

k_in <- 0.5   # the initial carrying capacity
n0_in <- 1e-5 # the initial absorbance reading
r_in <- 1.2   # the initial growth rate
N <- 50       # the number of "measurements" collected during the growth
              # curve experiment
```

```
data_t <- 0:N * 24 / N   # the times the measurements were made (in hours)
data_n <- NAtT(k = k_in, n0 = n0_in, r = r_in, t = data_t) # the measurements

# Now summarize the "experimental" growth data that we just generated
gc <- SummarizeGrowth(data_t, data_n)

# Get the possible metrics for fitness proxies
gc$vals$r           # growth rate is a common choice for fitness
gc$vals$t_gen       # doubling time, or generation time, is also common
gc$vals$k
gc$vals$n0
gc$vals$auc_l
gc$vals$auc_e
gc$vals$t_mid

# Compare the data with the fit visually by plotting it
plot(gc)
```

SummarizeGrowthByPlate

Summarize Growth Curves

Description

This function finds the parameters that describe the input data's growth for a plate of growth curves.
It does so by fitting the logistic curve to your growth curve measurements.

Usage

```
SummarizeGrowthByPlate(
  plate,
  t_trim = 0,
  bg_correct = "min",
  plot_fit = FALSE,
  plot_file = "growthcurver.pdf"
)
```

Arguments

plate   A data.table with at least two columns. One column contains timepoints that
                 measurements were taken (e.g., hours) and must be named "time". An optional
                 column can be included called "blank" that contains the blank readings for back-
                 ground correction (make sure to select the "blank" bg_correct option if you pro-
                 vide a blank column). Each remaining column contains the absorbance readings
                 from a single well in a plate.

t_trim	Measurements taken after this time should not be included in fitting the curve. If stationary phase is variable, this may give you a better fit. A value of 0 means no trimming. Defaults to no trimming (0).
bg_correct	The background correction method to use. No background correction is performed for "none". Specifying "min" subtracts the smallest value in a column from all the rows in that column, and specifying "blank" subtracts the values from the blank vector from the data_n vector.
plot_fit	TRUE if you want to generate a pdf file that plots all columns provided in the plate along with the growthcurver's fit. The default value is FALSE, which generates no plots.
plot_file	The name of the file to save the plots to if you set plot_fit to TRUE. The default file is called "growthcurver.pdf".

Details

The logistic curve equation is

$$N_t = \frac{N_0 K}{N_0 + (K - N_0)e^{-rt}}$$

where N_t is the number of cells (or the absorbance reading) at time t, N_0 is the initial cell count (or absorbance reading), K is the carrying capacity, and r is the growth rate.

The fitness proxies returned are the parameters of the logistic equation and the area under the curve (a measure that integrates the effects of N_0, K, and r). See `gcfit` for more documentation on these.

This method expects that your data adhere to a particular format.

- The data are provided in a data.frame
- One column in the data.frame is named "time" and contains the time measurements (e.g., hours).
- Each remaining column contains the readings from a single well in a plate reader. The name of the column will be used to identify the sample in the output data.
- There are no missing values or non-numeric data in the data.frame.

Value

A data.table containing the summary metrics and residual error from the fit of the logistic curve to the data. The names of the input columns are used to identify each well (or sample).

See Also

See the accompanying Vignette for an example of how to use and interpret SummarizeGrowthBy-Plate. `https://CRAN.R-project.org/package=growthcurver/vignettes/Growthcurver-vignette.html`

Examples

```
#Get the summary metrics for the entire plate of sample data provided
#with the Growthcurver package
```

```
#First, load the example data provided with Growthcurver. Note that there is
#a column named "time" -- this is necessary for Growthcurver to know which
#column contains the time measurements. In this dataset, the repeated
#measurements from a single well in a plate are given in a column of data.

myPlate <- growthdata
names(myPlate)

#Next, do the analysis for all the columns.
summary_plate <- SummarizeGrowthByPlate(plate = myPlate)

#The output is a data frame that contains the information on the best
#fit for each column of data.
head(summary_plate)      # Use head to display just the first few rows
```

Index