Package: BayesGrowth (via r-universe)

October 29, 2026

Type Package

Title Estimate Fish Growth Using MCMC Analysis

Date 2023-11-13

Version 1.0.0

Maintainer Jonathan Smart <jonsmartphd@gmail.com>

Description Estimate fish length-at-age models using MCMC analysis with 'rstan' models. This package allows a multimodel approach to growth fitting to be applied to length-at-age data and is supported by further analyses to determine model selection and result presentation. The core methods of this package are presented in Smart and Grammer (2021) ``Modernising fish and shark growth curves with Bayesian length-at-age models". PLOS ONE 16(2): e0246734 <doi:10.1371/journal.pone.0246734>.

License GPL-3

URL https://github.com/jonathansmart/BayesGrowth

BugReports https://github.com/jonathansmart/BayesGrowth/issues

Depends R (>= 3.4.0)

SystemRequirements GNU make

Imports AquaticLifeHistory, bayesplot, dplyr, ggplot2, loo, methods, Rcpp (>= 0.12.0), RcppParallel (>= 5.0.1), rstan (>= 2.18.1), rstantools (>= 2.3.1.1), stats, tibble, tidybayes

Suggests knitr, rmarkdown,

VignetteBuilder knitr

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

Biarch true

LinkingTo BH (>= 1.66.0), Rcpp (>= 0.12.0), RcppEigen (>= 0.3.3.3.0), RcppParallel (>= 5.0.1), rstan (>= 2.18.1), StanHeaders (>= 2.18.0)

NeedsCompilation yes

Author Jonathan Smart [aut, cre, ctb, cph]
(<https://orcid.org/0000-0003-2070-3208>)

Config/pak/sysreqs make

Repository https://cran.r-universe.dev

Date/Publication 2023-11-22 02:39:57 UTC

RemoteUrl https://github.com/cran/BayesGrowth

RemoteRef HEAD

RemoteSha 4bed80d8a40df9a07329bbbae39a8d68f33a0db2

Contents

BayesGrowth 2
Calc_Gompertz_LAA 3
Calc_Logistic_LAA 3
Calc_VBGF_LAA 4
Calculate_MCMC_growth_curve 5
Compare_Growth_Models 6
Estimate_MCMC_Growth 8
example_data 10
Get_MCMC_parameters 11
Looic_example_results 12
MCMC_example_results 12

Index **13**

`BayesGrowth` *BayesGrowth*

Description

A package to estimate fish length-at-age models using MCMC analysis with 'rstan' models.

Estimate fish length-at-age models using MCMC analysis with 'rstan' models. This package allows a multimodel approach to growth fitting to be applied to length-at-age data and is supported by further analyses to determine model selection and result presentation. The core methods of this package are presented in Smart and Grammer (2021) "Modernising fish and shark growth curves with Bayesian length-at-age models". PLOS ONE 16(2): e0246734.

References

To cite the BayesGrowth package in publications, type citation('BayesGrowth'). The Stan software should also be referenced: Stan Development Team (2020). RStan: the R interface to Stan. R package version 2.19.3. <https://mc-stan.org>

Calc_Gompertz_LAA *Calc_Gompertz_LAA*

Description

Calc_Gompertz_LAA

Usage

```
Calc_Gompertz_LAA(Linf, k, L0, Age)
```

Arguments

Linf	A single value of asymptotic length for the Gompertz model
k	A single value of the growth completion parameter for the Gompertz model
L0	A single value of length-at-birth for the Gompertz model
Age	A single value or vector of ages to convert to length based on the Gompertz model

Value

A vector of length-at-ages

Examples

```
# Calculate length-at-age for a Gompertz growth curve with given parameters
# and age range
Age_Range <-  0:16
Calc_Gompertz_LAA(Linf = 1580, k = 0.21, L0 = 72, Age = Age_Range)
```

Calc_Logistic_LAA *Calc_Logistic_LAA*

Description

Calc_Logistic_LAA

Usage

```
Calc_Logistic_LAA(Linf, k, L0, Age)
```

Arguments

Linf	A single value of asymptotic length for the logistic model
k	A single value of the growth completion parameter for the logistic model
L0	A single value of length-at-birth for the logistic model
Age	A single value or vector of ages to convert to length based on the logistic model

Value

A vector of length-at-ages

Examples

```
# Calculate length-at-age for a logistic growth curve with given parameters
# and age range
Age_Range <- 0:16
Calc_Logistic_LAA(Linf = 1560, k = 0.36, L0 = 73, Age = Age_Range)
```

Calc_VBGF_LAA *Calc_VBGF_LAA*

Description

Calc_VBGF_LAA

Usage

```
Calc_VBGF_LAA(Linf, k, L0, Age)
```

Arguments

Linf	A single value of asymptotic length for the von Bertalanffy model
k	A single value of the growth completion parameter for the von Bertalanffy model
L0	A single value of length-at-birth for the von Bertalanffy model
Age	A single value or vector of ages to convert to length based on the von Bertalanffy model

Value

A vector of length-at-ages

Examples

```
# Calculate length-at-age for a von Bertalanffy growth curve with given parameters
# and age range
Age_Range <-  0:16
Calc_VBGF_LAA(Linf = 1630, k = 0.15, L0 = 71, Age = Age_Range)
```

Calculate_MCMC_growth_curve

Calculate_MCMC_growth_curve

Description

A 'stan.fit' object produced from Estimate_MCMC_Growth is converted to a dataframe and structured using requested quantiles. This function takes the list of MCMC results for multiple chains, restructures them into a dataframe and calculates quantiles around length-at-age estimates. The quantiles are produced using the tidybayes::mean_qi() function and this result is returned from the function. This can be conveniently plotted in a ggplot using the geom_lineribbon() function provided in the tidybayes package.

Usage

```
Calculate_MCMC_growth_curve(
  obj,
  Model = NULL,
  max.age = NULL,
  probs = c(0.5, 0.75, 0.95)
)
```

Arguments

obj	An output from the Estimate_MCMC_Growth function
Model	The model used in the Estimate_MCMC_Growth object. Either "VB", "Gom" or "Log".
max.age	The max age to estimate growth up until.
probs	The percentiles of the results to return. Can be a single value or a vector of values. A single quantile width is required rather than its range. For example, 50th percentiles would be width = .5 which would return a lower percentile at .25 and an upper percentile of .75.

Value

A tibble that has been formatted using tidybayes::mean_qi(). This includes variables: Age, LAA, .lower, .upper, .width, .point and .interval.

Examples

```
# load example data
data("example_data")
## Biological info - lengths in mm
max_size <- 440
max_size_se <- 5
birth_size <- 0
birth_size_se <- 0.001 # an se cannot be zero
```

```
# Use the function to estimate the rstan model
fit <- Estimate_MCMC_Growth(data = example_data,
                            Model = "VB" ,
                            iter = 5000,
                            Linf = max_size,
                            Linf.se = max_size_se,
                            L0 = birth_size,
                            sigma.max = 100,
                            L0.se = birth_size_se,
                            k.max = 1,
                            n_cores = 1)

# Use function to return a dataframe of model predictionsfor VB growth model
Calculate_MCMC_growth_curve(fit, Model = "VB" , max.age = max(example_data$Age))
```

Compare_Growth_Models *Compare_Growth_Models*

Description

Conduct growth model selection using 'Leave One Out' (LOO) cross validation analysis and Widely Applicable Information Criterion (WAIC)for three growth models: (von Bertalanffy, Gompertz and Logistic) using the same prior parameters for each.

Usage

```
Compare_Growth_Models(
  data,
  Linf = NULL,
  Linf.se = NULL,
  L0 = NULL,
  L0.se = NULL,
  k.max = NULL,
  sigma.max = NULL,
  iter = 10000,
  BurnIn = iter/2,
  n_cores = 1,
  controls = NULL,
  n.chains = 4,
  thin = 1,
  verbose = FALSE,
  stats = "LooIC"
)
```

Arguments

data	A data.frame that contains columns named 'Age' and "Length'. The function can detect columns with similar names. If age and length columns cannot be determined then an error will occur. The dataset can have additional columns which will be ignored by the function
Linf	The prior for asymptotic length. Must be in the same unit (i.e. cm or mm) as the data. This should be based off of maximum size for the species.
Linf.se	The prior for normally distributed standard error around asymptotic length. Must be in the same unit (i.e. cm or mm) as the data. Cannot be zero.
L0	The prior for length-at-birth. Must be in the same unit (i.e. cm or mm) as the data. This should be based off of minimum size for the species.
L0.se	The prior for normally distributed standard error around length-at-birth. Must be in the same unit (i.e. cm or mm) as the data. Cannot be zero.
k.max	The maximum value to consider for the growth completion parameter 'k'. In the Gompertz and Logistic models, this parameter is often notated as 'g' instead of 'k'.
sigma.max	The maximum value to consider for sigma. This is the variance around the length-at-age residuals.
iter	How many MCMC iterations should be run? Default is 10000 but fewer can be useful to avoid longer run times when testing code or data
BurnIn	The number of iterations at the beginning of each chain to discard ('Burn in') to avoid biased values from starting values that do not resemble the target distribution. Default is iter/2.
n_cores	The number of cores to be used for parallel processing. It should be 1 core less than the maximum number available.
controls	A named list of parameters to control the rstan models behaviour.
n.chains	Number of MCMC chains to be run. Default is 4.
thin	The thinning of the MCMC simulations. Default is 1 which means no thinning occurs. Thinning is generally only necessary for complicated models as it increases run time.
verbose	TRUE or FALSE: flag indicating whether to print intermediate output from Stan on the console, which might be helpful for model debugging.
stats	Which statistics should be returned: LooIC, WAIC or both (both will return a list)

Value

A dataframe with the requested stats

Examples

```
# load example data
data("example_data")
## Biological info - lengths in mm
```

```
max_size <- 440
max_size_se <- 5
birth_size <- 0
birth_size_se <- 0.001 # an se cannot be zero

# Use the function to compare growth models with LooIC
Looic_example_results <- Compare_Growth_Models(data = example_data,
                                               stats = "LooIC",
                                               iter = 10000,
                                               n.chains = 4,
                                               BurnIn = 1000,
                                               thin = 1,
                                               n_cores = 1,
                                               Linf = max_size,
                                               Linf.se = max_size_se,
                                               L0 = birth_size,
                                               L0.se = birth_size_se,
                                               verbose = TRUE,
                                               sigma.max = 100,
                                               k.max = 1)
```

`Estimate_MCMC_Growth` *Estimate_MCMC_Growth*

Description

A wrapper function that creates a Stan MCMC model using the rstan package. The data and priors provided are combined into an rstan model that estimates a length-at-age model with a normal distribution. Three different growth models can be used: a von Bertalanffy model, Gompertz model or a logistic model. The prior on Linf and L0 are normally distributed and determined through the user providing a mean and se for each parameter. The growth completion parameter for any model (k) has a uniform prior which only requires an upper bound with the lower bound set at zero. Sigma is the residual variance of the data around the model and is set up in the same manner as 'k'. The growth estimates in the model are truncated to remain above zero so negative growth cannot occur.

Usage

```
Estimate_MCMC_Growth(
  data,
  Model = NULL,
  Linf = NULL,
  Linf.se = NULL,
  L0 = NULL,
  L0.se = NULL,
  k.max = NULL,
  sigma.max = NULL,
  iter = 10000,
  BurnIn = iter/2,
```

```
  n_cores = 1,
  controls = NULL,
  n.chains = 4,
  thin = 1,
  verbose = FALSE
)
```

Arguments

data	A data.frame that contains columns named 'Age' and "Length'. The function can detect columns with similar names. If age and length columns cannot be determined then an error will occur. The dataset can have additional columns which will be ignored by the function
Model	Which growth model should be run? Must be one of "VB", "Gom" or "Log" for von Bertalanffy, Gompertz or Logistic models, respectively
Linf	The prior for asymptotic length. Must be in the same unit (i.e. cm or mm) as the data. This should be based off of maximum size for the species.
Linf.se	The prior for normally distributed standard error around asymptotic length. Must be in the same unit (i.e. cm or mm) as the data. Cannot be zero.
L0	The prior for length-at-birth. Must be in the same unit (i.e. cm or mm) as the data. This should be based off of minimum size for the species.
L0.se	The prior for normally distributed standard error around length-at-birth. Must be in the same unit (i.e. cm or mm) as the data. Cannot be zero.
k.max	The maximum value to consider for the growth completion parameter 'k'. In the Gompertz and Logistic models, this parameter is often notated as 'g' instead of 'k'.
sigma.max	The maximum value to consider for sigma. This is the variance around the length-at-age residuals.
iter	How many MCMC iterations should be run? Default is 10000 but fewer can be useful to avoid longer run times when testing code or data
BurnIn	The number of iterations at the beginning of each chain to discard ('Burn in') to avoid biased values from starting values that do not resemble the target distribution. Default is iter/2.
n_cores	The number of cores to be used for parallel processing. It should be 1 core less than the maximum number available.
controls	A named list of parameters to control the rstan models behaviour.
n.chains	Number of MCMC chains to be run. Default is 4.
thin	The thinning of the MCMC simulations. Default is 1 which means no thinning occurs. Thinning is generally only necessary for complicated models as it increases run time.
verbose	TRUE or FALSE: flag indicating whether to print intermediate output from Stan on the console, which might be helpful for model debugging.

Value

An object of class 'stanfit' from the rstan package.

Examples

```
# load example data
data("example_data")
## Biological info - lengths in mm
max_size <- 440
max_size_se <- 5
birth_size <- 0
birth_size_se <- 0.001 # an se cannot be zero

# Use the function to estimate the rstan model
fit <- Estimate_MCMC_Growth(data = example_data,
                            Model = "VB" ,
                            iter = 5000,
                            Linf = max_size,
                            Linf.se = max_size_se,
                            L0 = birth_size,
                            sigma.max = 100,
                            L0.se = birth_size_se,
                            k.max = 1,
                            n_cores = 1)
```

---

example_data *Example length-at-age data*

Description

A dataset used as examples in the vignettes and for users to test code with

Usage

```
data(example_data)
```

Format

A data frame with 509 rows and 2 variables

Details

- Age. Number of growth bands determined from vertebral analysis
- Length. Total Length in mm determined via back-calculation

Get_MCMC_parameters *Get_MCMC_parameters*

Description

Get parameter summary statistics from the outputs of a Estimate_MCMC_Growth object. It is simplified set of results than is returned from summary(obj).

Usage

```
Get_MCMC_parameters(obj)
```

Arguments

obj	An output from the Estimate_MCMC_Growth function

Value

A data.frame with the posterior distributions for each parameter. These include the mean, Standard error of the mean, Standard deviation of the mean, median, 95th percentiles, effective sample sizes and Rhat.

Examples

```
# load example data
data("example_data")
## Biological info - lengths in mm
max_size <- 440
max_size_se <- 5
birth_size <- 0
birth_size_se <- 0.001 # an se cannot be zero

# Use the function to estimate the rstan model
fit <- Estimate_MCMC_Growth(data = example_data,
                            Model = "VB" ,
                            iter = 5000,
                            Linf = max_size,
                            Linf.se = max_size_se,
                            L0 = birth_size,
                            sigma.max = 100,
                            L0.se = birth_size_se,
                            k.max = 1,
                            n_cores = 1)

# Use function to return a dataframe of model results
Get_MCMC_parameters(fit)
```

Looic_example_results *Example MCMC model comparison outputs outputs*

Description

The results of an MCMC model comparison used as examples in the vignettes and for users to test code with

Usage

```
data(Looic_example_results)
```

Format

An dataframe with results of LooIC

MCMC_example_results *Example MCMC model outputs*

Description

The results of an MCMC model used as examples in the vignettes and for users to test code with

Usage

```
data(MCMC_example_results)
```

Format

An 'rstan' model with the class stan.fit

Index

∗ **datasets**
example_data, 10
Looic_example_results, 12
MCMC_example_results, 12

BayesGrowth, 2

Calc_Gompertz_LAA, 3
Calc_Logistic_LAA, 3
Calc_VBGF_LAA, 4
Calculate_MCMC_growth_curve, 5
Compare_Growth_Models, 6

Estimate_MCMC_Growth, 8
example_data, 10

Get_MCMC_parameters, 11

Looic_example_results, 12

MCMC_example_results, 12