# Package 'AquaticLifeHistory'

January 10, 2026

**Type** Package

**Title** Life History Analysis Tools

**Version** 1.0.6

**Date** 2026-01-09

**Maintainer** Jonathan Smart <jonsmartphd@gmail.com>

**Description** Estimate aquatic species life history using robust techniques.
This package supports users undertaking two types of analysis: 1) Growth from
length-at-age data, and 2) maturity analyses for length and/or age data.
Maturity analyses are performed using generalised linear model approaches incorporating
either a binomial or quasibinomial distribution.
Growth modelling is performed using the multimodel approach presented by
Smart et al. (2016) ``Multimodel approaches in shark and ray growth studies:
strengths, weaknesses and the future'' <doi:10.1111/faf.12154>.

**Encoding** UTF-8

**LazyData** true

**Depends** R (>= 3.3)

**Imports** plyr, MASS, readr, broom, MuMIn, magrittr, rlist, minpack.lm,
dplyr, tidyr, ggplot2

**RoxygenNote** 7.2.3

**License** GPL-3

**Suggests** knitr, rmarkdown

**VignetteBuilder** knitr

**URL** https://github.com/jonathansmart/AquaticLifeHistory,
https://jonathansmart.github.io/AquaticLifeHistory/

**BugReports** https://github.com/jonathansmart/AquaticLifeHistory/issues

**NeedsCompilation** no

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

**Repository** CRAN

**Date/Publication** 2026-01-10 07:53:19 UTC

# Contents

---

AquaticLifeHistory          *Introduction to AquaticLifeHistory*

---

### Description

Estimate aquatic species life history using robust techniques. This package supports users undertaking two types of analysis: 1) Growth from length-at-age data, and 2) maturity analyses for length and/or age data.

Maturity analyses are performed using generalised linear model approaches incorporating either a binomial or quasibinomial distribution.

Growth modelling is performed using the multimodel approach presented by Smart et al. (2016) "Multimodel approaches in shark and ray growth studies: strengths, weaknesses and the future" <doi:10.1111/faf.12154>.

### Author(s)

Jonathan Smart

### References

To cite the AquaticLifeHistory package in publications, type citation('AquaticLifeHistory').

---

boot_data          *boot_data*

---

### Description

boot_data

### Usage

```
boot_data(.data, n = 100)
```

## Arguments

| | |
|---|---|
| `.data` | Length at Age data |
| `n` | number of bootstrap iterations |

## Value

a grouped_df produced by the group_by function in dplyr

---

Calculate_MMI *Calculate model averaged length-at-age estimates and parameters*

---

## Description

'Calculate_MMI' takes the outputs from an 'Estimate_Growth' function with plots = FALSE and returns the calculated model averaged parameters, SE and estimates based on AIC scores. It should be used if no candidate model has an AIC weight higher than 0.9. Use of this function should cite Smart et al (2016)

## Usage

```
Calculate_MMI(data)
```

## Arguments

| | |
|---|---|
| `data` | An output from the Estimate_Growth function with plots = FALSE |

## Value

A list with model averaged parameters and a dataframe of model averaged length-at-age estimates

## References

Smart et al. (2016) Multi model approaches in shark and ray growth studies: strengths, weaknesses and the future. Fish and Fisheries. 17: 955-971https://onlinelibrary.wiley.com/doi/abs/10.1111/faf.12154

## Examples

```
# load example data set
data("growth_data")
# Run function with three default model candidates and return results
# without plots. Use 100 bootstraps for testing and then increase to at
# least 1000 for actual model runs.
models <- Estimate_Growth(growth_data, plots = FALSE,  n.bootstraps = 100)
# Calculate the model average of the resulting growth estimates
Calculate_MMI(models)
```

---

Estimate_Age_Maturity    *Estimate age-at-maturity*

---

### Description

Age-at-maturity is estimated from binary maturity data using a logistic ogive. Two options are available depending on error structure. If binary data are used then a binomial error structure is required. If the user wishes to bin the data by age class then a quasi binomial error structure is needed with the data weighted by the sample size of each bin. This is handled automatically by the function.

### Usage

```
Estimate_Age_Maturity(
  data,
  error.structure = "binomial",
  n.bootstraps = 1000,
  display.points = FALSE,
  return = "parameters"
)
```

### Arguments

data
: A dataframe that includes age and a binary maturity status (immature = 0 and mature = 1). Columns should be named "Age" and "Maturity" but the function is robust enough to accept some reasonable variations to these

error.structure
: The distribution for the glm used to produce the logistic ogive. Must be either "binomial" for binary data or "quasi binomial" for binned maturity at age. Proportion mature at each age is automatically calculated within the function

n.bootstraps
: Number of bootstrap iterations required to produce 95% confidence intervals about the logistic ogive

display.points
: Should the raw data be plotted for the binomial model?

return
: Either:

    **parameters** The estimated logistic parameters and their standard error (A50 and A95)

    **estimates** The logistic ogive predictions with 95 percent confidence intervals (useful for creating ones own plots)

    **plot** a ggplot object of the logistic ogive.

### Value

Either:

**parameters** a dataframe of the estimated logistic parameters and their standard error (A50 and A95)

**estimates** a dataframe of logistic ogive predictions with 95 percent confidence intervals

**plot** a ggplot object of the logistic ogive

## Examples

```
# load example data set
data("maturity_data")
# Run function to estimate age-at-maturity parameters
Estimate_Age_Maturity(maturity_data)
# A plot can also be returned with bootstrapped CI's.  Use 100 bootstraps for
# testing and then increase to at least 1000 for actual model runs.
Estimate_Age_Maturity(maturity_data, return = "plot",n.bootstraps = 100)
```

---

Estimate_Growth *Estimate length-at-age parameters and growth curves for Elasmo-branchs*

---

## Description

A multi-model growth estimation approach is applied to length-at-age data. Three models can be applied which include the von Bertalanffy (VB), logistic (Log) and Gompertz (Gom) models. AIC values and weights are calculated. The outputs will return a list of model parameter estimates and will either print a plot to the screen or output the length-at-age estimates as part of the list.Use of this function should cite Smart et al. (2016).

## Usage

```
Estimate_Growth(
  data,
  models = c("VB", "Log", "Gom"),
  Birth.Len = NULL,
  correlation.matrix = FALSE,
  n.bootstraps = 1000,
  plots = TRUE,
  Max.Age = NULL,
  plot.legend = TRUE
)
```

## Arguments

| | |
|---|---|
| data | a data frame which includes 'Age' and 'Length - ideally with these names but the function will except some variation to these |
| models | a vector of models to be fitted. These can include" VB", "Log" and "Gom". A subset can also be used |
| Birth.Len | The length-at-birth to be used for two parameter models. If a value is provided, two parameter models are automatically run |

correlation.matrix

                Should the correlation matrix of parameters be returned? This is the only object returned if TRUE.

n.bootstraps    The number of bootstraps performed for model 95 confidence intervals

plots              Should plots be printed to the screen. If FALSE then the model estimates and CI's are returned as an additional output

Max.Age         Specify the max age for bootstrapped confidence intervals to be produced over. Default is the max age in the data.

plot.legend     Do you want a legend for the different models on the plot

## Value

Returns a list of parameter estimates with errors and AIC results. If plots is TRUE then a plot is printed to the screen. If plots is FALSE then the length-at-age estimates are returned as a list element

## References

Smart et al. (2016) Multi-model approaches in shark and ray growth studies: strengths, weaknesses and the future. Fish and Fisheries. 17: 955-971[https://onlinelibrary.wiley.com/doi/abs/](https://onlinelibrary.wiley.com/doi/abs/10.1111/faf.12154)
[10.1111/faf.12154](https://onlinelibrary.wiley.com/doi/abs/10.1111/faf.12154)

## Examples

```
# load example data set
data("growth_data")
# Run function with three default model candidates. Use 100 bootstraps for
# testing and then increase to at least 1000 for actual model runs.
Estimate_Growth(growth_data, n.bootstraps = 100)
```

---

Estimate_Len_Maturity   *Estimate length-at-maturity*

---

## Description

Length-at-maturity is estimated from binary maturity data using a logistic ogive. Two options are available depending on error structure. If binary data are used then a binomial error structure is required. If the user wishes to bin the data by length class then a quasi binomial error structure is needed with the data weighted by the sample size of each bin. This is handled automatically by the function.

## Usage

```
Estimate_Len_Maturity(
  data,
  error.structure = "binomial",
  n.bootstraps = 1000,
  bin.width = NA,
  display.points = FALSE,
  return = "parameters"
)
```

## Arguments

data
: A dataframe that includes length and a binary maturity status (immature = 0 and mature = 1). Columns should be named "Length" and "Maturity" but the function is robust enough to accept some reasonable variations to these

error.structure
: The distribution for the glm used to produce the logistic ogive. Must be either "binomial" for binary data or "quasi binomial" for binned maturity at length. Proportion mature at each length bin is automatically calculated within the function

n.bootstraps
: Number of bootstrap iterations required to produce 95% confidence intervals about the logistic ogive

bin.width
: The width of the length-class bins used for a quasi binomial logistic model. These should on the same unit as the length data. The y axis on any plots will automatically scale to the correct unit ("cm" or "mm")

display.points
: Should the raw data be plotted for the binomial model?

return
: Either:

  **parameters** The estimated logistic parameters and their standard error (L50 and L95)

  **estimates** The logistic ogive predictions with 95 percent confidence intervals (useful for creating ones own plots)

  **plot** a ggplot object of the logistic ogive.

## Value

Either:

**parameters** a dataframe of the estimated logistic parameters and their standard error (L50 and L95)

**estimates** a dataframe of logistic ogive predictions with 95 percent confidence intervals

**plot** a ggplot object of the logistic ogive. If binned length classes are used, this includes a bar plot of proportional maturity

## Examples

```
# load example data set

data("maturity_data")
# Run function to estimate length-at-maturity parameters
Estimate_Len_Maturity(maturity_data)
# A plot can also be returned with bootstrapped CI's. Use 100 bootstraps for
# testing and then increase to at least 1000 for actual model runs.
Estimate_Len_Maturity(maturity_data, return = "plot",n.bootstraps = 100)
```

---

growth_data                *Length-at-age data for blacktip sharks*

---

## Description

A data set containing the length-at-age data for common blacktip sharks (*Carcharhinus limbatus*) from Indonesia. This data was published in Smart et al. (2015).

## Usage

```
data(growth_data)
```

## Format

A data frame with 294 rows and 3 variables

## Details

- Age. Number of growth bands determined from vertebral analysis
- Length. Total Length in mm determined via back-calculation
- Sex. Females (F) or males (M)

## References

Smart et al (2015) Age and growth of the common blacktip shark Carcharhinus limbatus from Indonesia, incorporating an improved approach to comparing regional population growth rates. African Journal of Marine Science 37:177-188. https://www.tandfonline.com/doi/abs/10.2989/1814232X.2015.1025428

| maturity_data | *Age and length-at-maturity data for silky sharks* |
|---|---|

## Description

A data set containing the length-at-maturity and age-at-maturity data for female silky sharks (*Carcharhinus falciformis*) from Papua New Guinea. This data was published in Grant et al (2018)

## Usage

```
data(maturity_data)
```

## Format

A data frame with 284 rows and 4 variables

## Details

- Tag. Unique identifier for each Shark
- Age. Number of growth bands determined from vertebral analysis
- Length. Total Length in cm
- Maturity. Binary maturity status: immature = 0 and mature = 1

## References

Grant et al (2018) Life history characteristics of the silky shark (*Carcharhinus falciformis*) from the central west Pacific. Marine and Freshwater Research 69:562-573 http://www.publish.csiro.au/mf/MF17163

# Index