

Package ‘aLBI’

May 7, 2026

Type Package

Title Estimating Length-Based Indicators for Fish Stock

Version 0.1.9

Maintainer Ataher Ali <ataher.cu.ms@gmail.com>

Description Provides tools for estimating length-based indicators from length frequency data to assess fish stock status and manage fisheries sustainably. Implements methods from Cope and Punt (2009) <doi:10.1577/C08-025.1> for data-limited stock assessment and Froese (2004) <doi:10.1111/j.1467-2979.2004.00144.x> for detecting overfishing using simple indicators. Key functions include:

FrequencyTable(): Calculate the frequency table from the collected and also the extract the length frequency data from the frequency table with the upper length_range. A numeric value specifying the bin width for class intervals. If not provided, the bin width is automatically calculated using Wang (2020) <doi:10.1016/j.fishres.2019.105474> formula.

FreqTM(): Creates a frequency distribution table for fish length data across multiple months using a consistent length class structure. The bin width is determined by either a custom value or Wang's formula, applied uniformly across all months. The function dynamically detects and renames columns to 'Month' and 'Length' from the input dataframe. The maximum observed length is included as part of the last class, with the upper bound set to the smallest multiple of the bin width greater than or equal to the maximum length. Months can be converted to dates using a configurable day and year, with dates assigned sequentially in 'day.month.year' format (e.g., 15.01.26).

FishPar(): Calculates length-based indicators (LBIs) proposed by Froese (2004) <doi:10.1111/j.1467-2979.2004.00144.x> such as the percentage of mature fish (Pmat), percentage of optimal length fish (Popt), percentage of mega spawners (Pmega), and the sum of these as Pobj. This function also estimates confidence intervals for different lengths, visualizes length frequency distributions, and provides data frames containing calculated values.

FishSS(): Makes decisions based on input from Cope and Punt (2009) <doi:10.1577/C08-025.1> and parameters calculated by FishPar() (e.g., Pobj, Pmat, Popt, LM_ratio) to determine stock status as target spawning biomass (TSB40) and limit spawning biomass (LSB25), and selectivity.

LWR(): Fits and visualizes length-weight relationships using linear regression, with options for log-transformation and customizable plotting.

Depends R (>= 4.0.0)

Imports dplyr, openxlsx, stats, graphics, grDevices, ggplot2, utils

Suggests testthat, knitr, rmarkdown, devtools, rlang, readxl

License GPL-3

Encoding UTF-8

LazyData true

URL <https://github.com/Ataher76/aLBI>

BugReports <https://github.com/Ataher76/aLBI/issues>

RoxygenNote 7.3.2

VignetteBuilder knitr

NeedsCompilation no

Author Ataher Ali [aut, cre],
Mohammed Shahidul Alam [aut]

Repository CRAN

Date/Publication 2026-01-11 00:30:02 UTC

Contents

CPdata	2
ExData	3
FishPar	4
FishSS	5
FreqTM	5
FrequencyTable	7
lenfreq01	8
lenfreq02	8
lenfreqM	9
LWdata	10
LWR	10
Index	13

CPdata

CPdata: Example dataset for aLBI package

Description

This dataset contains description of CPdata.

Usage

data(CPdata)

Format

A data frame with 11 columns:

A Probability values

B Probability values

C Probability values

D Probability values

E Probability values

F Probability values

G Probability values

H Probability values

I Probability values

J Probability values

Tx Target column compared with LM_ratio to pick probability values

Source

A decision table described by Cope and Punt (2009)

Examples

```
data(CPdata, package = "aLBI")
head(CPdata)
```

ExData

ExData: Example raw length dataset for aLBI package

Description

This dataset contains description of ExData.

Usage

```
data(ExData)
```

Format

A data frame with 1 column:

Length Sampled length data (cm)

Source

Data collected for fish stock assessment studies

Examples

```
data(ExData, package = "aLBI")
head(ExData)
```

 FishPar

Calculate Length-Based Indicators with Monte Carlo Simulation

Description

This function calculates length-based indicators using Monte Carlo simulation for length parameters and non-parametric bootstrap for Froese indicators. Plots are displayed in the plot panel, and PDFs and an Excel file of results are saved to the current working directory.

Usage

```
FishPar(data, resample = 1000, progress = FALSE, Linf = NULL, Linf_sd = 0.5, Lmat = NULL,
        Lmat_sd = 0.5)
```

Arguments

<code>data</code>	A data frame containing two columns: Length and Frequency.
<code>resample</code>	An integer indicating the number of Monte Carlo samples or bootstrap resamples (default: 1000).
<code>progress</code>	A logical value indicating whether to display a progress bar (default: FALSE).
<code>Linf</code>	A numeric value for the asymptotic length (optional). If provided, overrides the default $L_{max}/0.95$ calculation.
<code>Linf_sd</code>	A numeric value for the standard deviation of random variation added to <code>Linf</code> (default: 0.5). Only used if <code>Linf</code> is provided.
<code>Lmat</code>	A numeric value for the length at maturity (optional). If provided, overrides the default Monte Carlo estimation.
<code>Lmat_sd</code>	A numeric value for the standard deviation of random variation added to <code>Lmat</code> (default: 0.5). Only used if <code>Lmat</code> is provided.

Value

A list containing estimated length parameters, Froese indicators, and other metrics.

FishSS	<i>Assess Stock Status and Classify Fish selectivity</i>
--------	--

Description

This function assesses stock status and classifies fish selectivity based on the provided parameters.

Usage

```
FishSS(data, LM_ratio, Pmat, Popt, Pmega)
```

Arguments

data	A data frame containing the necessary columns for stock status calculation from Cope & Punt (2009)
LM_ratio	A numeric value representing the length at maturity ratio.
Pmat	A numeric value representing the percentage of mature fish.
Popt	A numeric value representing the percentage of optimally sized fish.
Pmega	A numeric value representing the percentage megaspawner.

Value

A list containing the selected columns, the target value, the closest value index, the calculated stock status, and the fish classification.

Examples

```
utils::data("CPdata", package = "aLBI")
FishSS(CPdata, 0.75, 100, 30, 25)
```

FreqTM	<i>FreqTM Generate Frequency Distribution Table for Fish Length Data Across Months</i>
--------	--

Description

Creates a frequency distribution table for fish length data across multiple months using a consistent length class structure. The bin width is determined by either a custom value or Wang's formula, applied uniformly across all months. The function dynamically detects and renames columns to 'Month' and 'Length' from the input dataframe. The maximum observed length is included as part of the last class, with the upper bound set to the smallest multiple of the bin width greater than or equal to the maximum length. Months can be converted to dates using a configurable day and year, with dates assigned sequentially in 'day.month.year' format (e.g., 15.01.26).

Usage

```

FreqTM(
  data,
  bin_width = NULL,
  Lmax = NULL,
  output_file = "FreqTM_Output.xlsx",
  date_config = list(day = 1, year = 2025)
)

```

Arguments

<code>data</code>	A data frame containing columns for months and lengths (names can vary, e.g., 'Month', 'Length', or any other names).
<code>bin_width</code>	Numeric value specifying the bin width for class intervals. If NULL (default), bin width is calculated using Wang's formula.
<code>Lmax</code>	Numeric value for the maximum observed fish length. Required only if 'bin_width' is NULL and Wang's formula is used. Defaults to NULL.
<code>output_file</code>	Character string specifying the output Excel file name. Defaults to "FreqTM_Output.xlsx".
<code>date_config</code>	A list with elements 'day' (default 1) and 'year' (default 2025) to set the day and year for converting month names to dates. The day must be between 1 and 31.

Value

A data frame with columns 'Length' (upper bound of each class) and monthly frequency columns as dates in 'day.month.year' format, where each row represents a length class and its frequency across months.

Examples

```

# Load required packages
library(dplyr)
library(openxlsx)

# Generate sample data with custom column names
set.seed(123)
sample_data <- data.frame(
  Time = rep(c("Aug", "Sep"), each = 100),
  Size = runif(200, min = 5, max = 20)
)

# Create frequency table with automatic bin width and default date config
result <- FreqTM(data = sample_data, output_file = tempfile(fileext = ".xlsx"))

# Create frequency table with custom bin width and specific date
result <- FreqTM(data = sample_data, bin_width = 2,
  date_config = list(day = 15, year = 2026),
  output_file = tempfile(fileext = ".xlsx"))

```

FrequencyTable	<i>FrequencyTable Generate a Frequency Distribution Table for Fish Length Data</i>
----------------	--

Description

Creates a frequency distribution table for fish length data using either a custom bin width or Wang's formula for automatic bin width calculation. The bin width is rounded to the nearest integer if calculated. The results are saved to an Excel file and returned as a list of data frames.

Usage

```
FrequencyTable(
  data,
  bin_width = NULL,
  Lmax = NULL,
  output_file = "FrequencyTable_Output.xlsx"
)
```

Arguments

data	A numeric vector or data frame containing fish length measurements. If a data frame is provided, the first numeric column is used.
bin_width	Numeric value specifying the bin width for class intervals. If NULL (default), bin width is calculated using Wang's formula.
Lmax	Numeric value for the maximum observed fish length. Required only if 'bin_width' is NULL and Wang's formula is used. Defaults to NULL.
output_file	Character string specifying the output Excel file name. Defaults to "FrequencyTable_Output.xlsx".

Value

A list containing two data frames:

lfqTable Frequency table with length ranges and their frequencies.

lfreq Table with upper limits of bins and their frequencies.

Examples

```
# Load required package
library(dplyr)

# Generate random fish length data
set.seed(123)
fish_lengths <- runif(200, min = 5, max = 70)

# Create frequency table with automatic bin width
```

```
FrequencyTable(data = fish_lengths, output_file = tempfile(fileext = ".xlsx"))  
  
# Create frequency table with custom bin width and output file  
FrequencyTable(data = fish_lengths, bin_width = 5, output_file = tempfile(fileext = ".xlsx"))
```

lenfreq01

lenfreq01: Example dataset for aLBI package

Description

This dataset contains description of lenfreq01.

Usage

```
data(lenfreq01)
```

Format

A data frame with 2 columns:

Frequency Observed individuals in each length class

Length Upper value of each length class (cm)

Source

Data collected for fish stock assessment studies

Examples

```
data(lenfreq01, package = "aLBI")  
head(lenfreq01)
```

lenfreq02

lenfreq02: Example dataset for aLBI package

Description

This dataset contains description of lenfreq02.

Usage

```
data(lenfreq02)
```

Format

A data frame with 2 columns:

Frequency Observed individuals in each length class

LengthClass Upper value of each length class (cm)

Source

Data collected for fish stock assessment studies

Examples

```
data(lenfreq02, package = "aLBI")
head(lenfreq02)
```

lenfreqM

lenfreqM: Example length-weight dataset for aLBI package

Description

This dataset contains length data of multiple months

Usage

```
data(lenfreqM)
```

Format

A data frame with 2 columns: First one is month and second one is length

Months Name of the Sampling Months

Length Measured lengths of the sampled fish

Source

Data collected for fish stock assessment studies

Examples

```
data(lenfreqM, package = "aLBI")
head(lenfreqM)
```

LWdata

LWdata: Example length-weight dataset for aLBI package

Description

This dataset contains length and weight measurements for fish.

Usage

```
data(LWdata)
```

Format

A data frame with 2 columns:

Length Length of sampled fish (cm)

Weight Weight of sampled fish (g)

Source

Data collected for fish stock assessment studies

Examples

```
data(LWdata, package = "aLBI")  
head(LWdata)
```

LWR

Plot and Model Length-Weight Relationships with Optional Log Transformation

Description

This function visualizes and models the relationship between length and weight (or any two continuous variables) using linear regression. It supports both standard and log-transformed models, producing a ggplot2-based plot with a fitted line, optional confidence interval shading, and annotations for the regression equation, R^2 , and p-value. When `save_output` is TRUE, the plot and model summary are saved to the working directory as a PDF and text file, respectively.

Usage

```
LWR(
  data,
  log_transform = TRUE,
  point_col = "black",
  line_col = "red",
  shade_col = "red",
  point_size = 2,
  line_size = 1,
  alpha = 0.2,
  main = "Length-Weight Relationship",
  xlab = NULL,
  ylab = NULL,
  save_output = TRUE
)
```

Arguments

<code>data</code>	A data frame with at least two columns: the first for length, the second for weight.
<code>log_transform</code>	Logical. Whether to apply a log-log transformation to the variables. Default is TRUE.
<code>point_col</code>	Color of the data points. Default is "black".
<code>line_col</code>	Color of the regression line. Default is "red".
<code>shade_col</code>	Color for the confidence interval ribbon. Default is "red".
<code>point_size</code>	Size of the data points. Default is 2.
<code>line_size</code>	Size of the regression line. Default is 1.
<code>alpha</code>	Transparency for the confidence interval ribbon. Default is 0.2.
<code>main</code>	Title of the plot. Default is "Length-Weight Relationship".
<code>xlab</code>	Optional. Custom x-axis label. If NULL, a label is generated based on <code>log_transform</code> .
<code>ylab</code>	Optional. Custom y-axis label. If NULL, a label is generated based on <code>log_transform</code> .
<code>save_output</code>	Logical. Whether to save the plot as a PDF and the model summary as a text file. Default is TRUE.

Value

A list containing:

<code>model</code>	The fitted <code>lm</code> object
<code>intercept</code>	The estimated intercept (back-transformed if <code>log_transform = TRUE</code>)
<code>slope</code>	The estimated slope
<code>r_squared</code>	R-squared value
<code>correlation_r</code>	Correlation coefficient (r)
<code>p_value</code>	P-value for slope
<code>plot</code>	The <code>ggplot</code> object for further customization

Examples

```
data(LWdata, package = "aLBI")
result <- LWR(LWdata, log_transform = TRUE, save_output = FALSE)
print(result$plot)
```

Index

CPdata, [2](#)

ExData, [3](#)

FishPar, [4](#)

FishSS, [5](#)

FreqTM, [5](#)

FrequencyTable, [7](#)

lenfreq01, [8](#)

lenfreq02, [8](#)

lenfreqM, [9](#)

LWdata, [10](#)

LWR, [10](#)