LBSPR: An R package for simulation and estimation using life-history ratios and length composition data

Modificado de Adrian Hordyk, Autor:Mauricio Mardones

18 de marzo de 2019

Resumen

El presente documento tiene como objetivo aplicar la metodologia del metodo de evualación de stock basado en datos pobres llamado Length-Based Spawning Potential Ratio (LBSPR) (Hordyk, 2014). En este ejercicio se estudia el caso de la almeja (Venus antiqua) de la región XI en el sur de Chile. Este adelanto metodológico se ha realizado en las 2 ultimas evaluaciones reportasdas en el proyecto ASIPA "Programa de Seguimiento de pesquerías Bentónicas bajo Planes de Manejo".

1 Introduction

This package contains functions to run the Length-Based Spawning Potential Ratio (LBSPR) method. The LBSPR package can be used in two ways: 1) simulating the expected length composition, growth curve, and SPR and yield curves using the LBSPR model and 2) fitting to empirical length data to provide an estimate of the spawning potential ratio (SPR).

The LBSPR method has been developed for data-limited fisheries, where few data are available other than a representative sample of the size structure of the vulnerable portion of the population (i.e., the catch) and an understanding of the life history of the species. The LBSPR method does not require knowledge of the natural mortality rate (M), but instead uses the ratio of natural mortality and the von Bertalanffy growth coefficient (K) (M/K), which is believed to vary less across stocks and species than M (Prince et al. 2015).

Like any assessment method, the LBSPR model relies on a number of simplifying assumptions. In particular, the LBSPR models are equilibrium based, and assume that the length composition data is representative of the exploited population at steady state. See the publicaitons listed in the reference list for full details of the assumptions of the model, including simulation testing to evauate the effect of violations of these assumptions.

There are two versions of the LBSPR model included in this package.

1.1 Age-Structured Length-Based Model

The LBSPR model described by Hordyk et al. (2015a, b), and tested in a MSE framework (Hordyk et al. 2015c), use a conventional age-structured equilibrium population model. An important assumption of this model structure is that selectivity is age-based not length-based.

1.2 Length-Structured Growth-Type-Group Model

Hordyk et al. (2016) describe a length-structured version of the LBSPR model that uses growth-type-groups (GTG) to account for size-based selectivity. The GTG-LBSPR model also has the ability to include variable M at size (by default M is assumed to be constant). The GTG-LBSPR model typically estimates a lower fishing mortality rate for a given size structure compared to the earlier age-structured model. This is because the age-structured model has a 'regeneration assumption', where, because of the age-based selectivity assumption, large individuals are expected even at high fishing mortality (large, young fish).

The default setting for the LBSPR package is to use the GTG-LBSPR model for all simulation and estimation. Control options in the simulation and estimation functions can be used to switch to the age-structured LBSPR model.

2 First Steps

2.1 Installing the Package

The LBSPR package is now available on CRAN:

```
rm(list=ls())
```

```
#install.packages("LBSPR")
#install.packages("devtools")
#devtools::install_github("AdrianHordyk/LBSPR")
```

Load the package

```
###load the package
library(LBSPR)
library(devtools)#funcion para install_github
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyr)
library(ggplot2)
library(stringr)
```

3 Simulation

The LBSPR package can be used to generate the expected size composition, the SPR, and relative yield for a given set of biological and exploitation pattern parameters.

3.1 LB_pars Object

The first thing to do is to create a LB_pars object that contains all of the required parameters for the simulation model. LB_pars is an S4 class object.

3.1.1 Create a new LB_pars Object

To create a new LB_pars object you use the new function:

```
MyPars <- new("LB_pars")
```

```
## A blank LB_pars object created
```

```
## Default values have been set for some parameters
```

You can see the elements or slots of the LB_pars object using the slotNames function:

```
slotNames(MyPars)
```

```
##  [1] "Species"      "MK"           "M"            "Linf"
##  [5] "L_units"      "CVLinf"       "L50"          "L95"
##  [9] "Walpha"       "Walpha_units" "Wbeta"        "FecB"
## [13] "Steepness"    "Mpow"         "R0"           "SL50"
## [17] "SL95"         "MLL"          "sdLegal"      "fDisc"
## [21] "FM"           "SPR"          "BinMin"       "BinMax"
## [25] "BinWidth"
```

MyPars is an object of class LB_pars. You can access the help file for classes by using the ? symbol (similar to how you find the help file for functions):

```
#class?LB_pars
```

3.1.2 Populate the LB_pars Object

The LB_pars object has 25 slots. However, not all parameters need to be specified for the simulation model.

Some parameters are essential, and a warning message should appear if you attempt to progress without values (please let me know if there are issues).

Default values will be used for some of the other parameters if no value is specified. For example, the first slot (Species) is a character object that can be used for the species name. If this slot is left empty, the simulation model will populate it with a default value.

A message should alert you any time a default value is being used. The minimum parameters that are needed for the simulation model are:

Biology

von Bertalanffy asymptotic length (Linf) M/K ratio (natural mortality divided by von Bertalanffy K coefficient) (MK) Length at 50% maturity (L50) Length at 95% maturity (L95) Exploitation - Length at 50% selectivity (SL50) - Length at 95% selectivity (SL95) - F/M ratio (FM) or SPR (SPR). If you specify both, the F/M value will be ignored.

Size Classes - Width of the length classes (BinWidth)

Remember, you can find the help documentation for the LB_pars object by typing: class?LB_pars in the console.

To create an example parameter object:

```
MyPars@Species <- "Venus antiqua"
MyPars@Linf <- 97.0#Jerez 1991
MyPars@L50 <- 32.7 #Jaramillo
MyPars@L95 <- 70 # verrificar bibliografia
MyPars@MK <- 0.28/0.218


#Explotacion
MyPars@SL50 <- 55#numeric() #1
MyPars@SL95 <- 90#numeric() #27
MyPars@SPR <- 0.4 #numeric()# ###cambia el numero 0.4 a en blanco
MyPars@BinWidth <- 1
#MyPars@FM <- 1

MyPars@Walpha <- 0.005
MyPars@Wbeta <- 3.0637 #r2 = 0.9651
```

BinMax not set. Using default of 1.3 Linf BinMin not set. Using default value of 0 You will notice some messages in the console alerting you that default values have been used. You can change these by specifying values in MyPars and re-running the LBSPRsim function.

We'll manually set those values here so we don't keep seeing the messages throughout the vignette.

```
MyPars@BinWidth <-1
MyPars@BinMax <- 120
MyPars@BinMin <- 0
MyPars@L_units <- "mm"
```

We can also choose to set the units for the length parameters:

MyPars@L_units <- "mm"

3.2 Running the Simulation Model

Now we are ready to run the LBSPR simulation model. To do this we use the LBSPRsim function: ngtg function es el # de grupos para el GTG model, por default es 13)

```
MySim <- LBSPRsim(MyPars, Control=list(modtype="GTG", maxFM=5))
```

```
## ngtg increased to 40 because of small bin size
```

3.2.1 The LB_obj Object

The output of the LBSPRsim function is an object of class LB_obj. This is another S4 object, and contains all of the information from the LB_pars object and the output of the LBSPRsim function.

Many of the functions in the LBSPR package return an object of class LB_obj. You should not modify the LB_obj object directly. Rather, make changes to the LB_pars object (MyPars in this case), and re-run the simulation model (or other functions, covered later in the vignette).

3.2.2 Simulation Output

Let's take a look at some of the simulated output.

```
MySim@SPR
```

```
## [1] 0.4
```

The simulated SPR is the same as our input value (MyPars@SPR).

What is the ratio of fishing mortality to natural mortality in this scenario?

```
MySim@FM
```

```
## [1] 1.09
```

It is important to note that the F/M ratio reported in the LBSPR model refers to the apical F over the adult natural mortality rate. That is, the value for fishing mortality refers to the highest level of F experienced by any single size class.

If the selectivity pattern excludes all but the largest individuals from being exploited, it is possible to have a very high F/M ratio in a sustainable fishery (high SPR). And visceverse!!

3.2.3 Control Options

There are a number of additional parameters that can be modified to control other aspects of the simulation model.

For example, by default the LBSPR model using the Growth-Type-Group model (Hordyk et at. 2016). The Control argument can be used to switch to the Age-Structured model (Hordyk et al. 2015a, b):

See the help file for the LBSPRsim function for additional parameters for the Control argument.

3.2.4 Plotting the Simulation

The plotSim function can be used to plot MySim:

By default the function plots: a) the expected (equilibrium) size structure of the catch and the expected unfished size structure of the vulnerable population, b) the maturity and selectivity-at-length curves, c) the von Bertalanffy growth curve with relative age, and d) the SPR and relative yield curves as a function of relative fishing mortality (see note above on the F/M ratio).

The plotSim function can be controlled in a number of ways. For example, you can plot the expected unfished and fished size structure of the population by changing the lf.type argument:

Individual plots can be created using the type argument:

See ?plotSim for more options for plotting the output of the LBSPR simulation model.

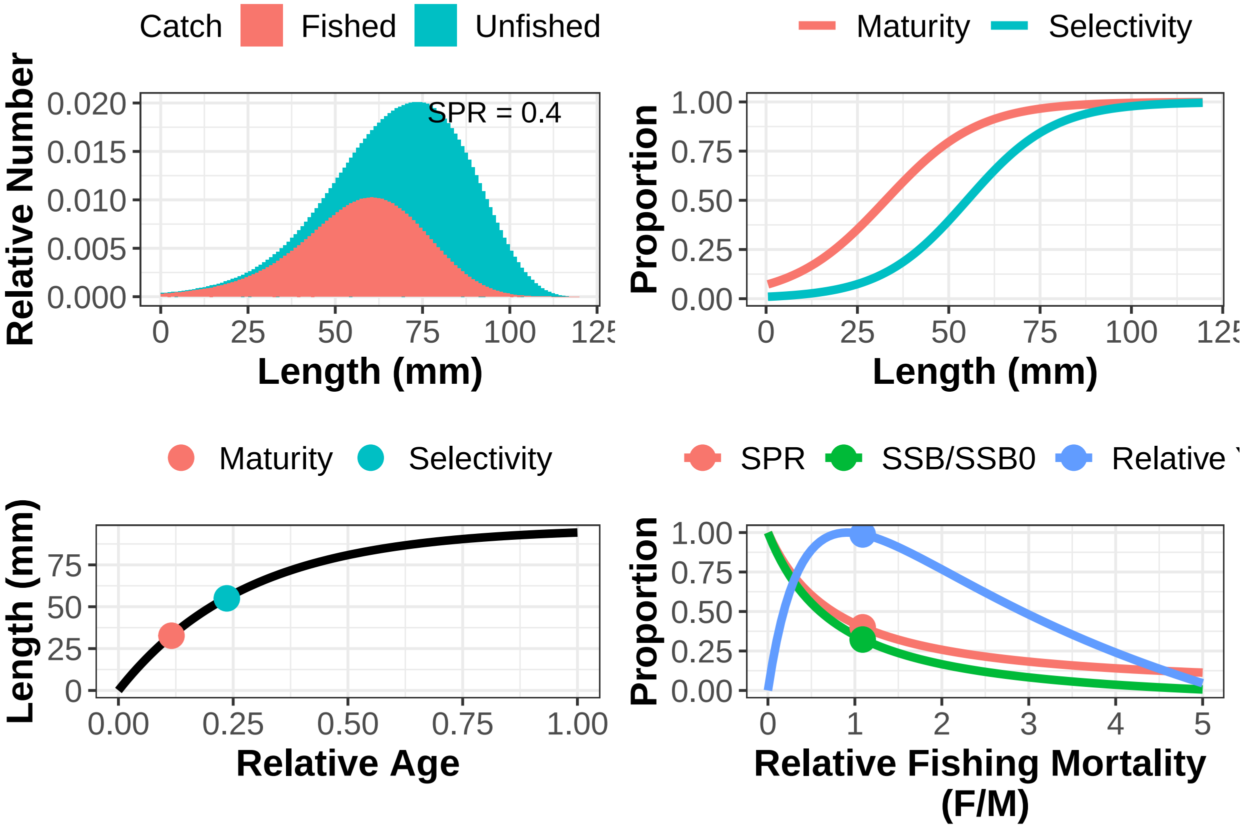


Figure 1: Ploteo de Simulación estructuras.

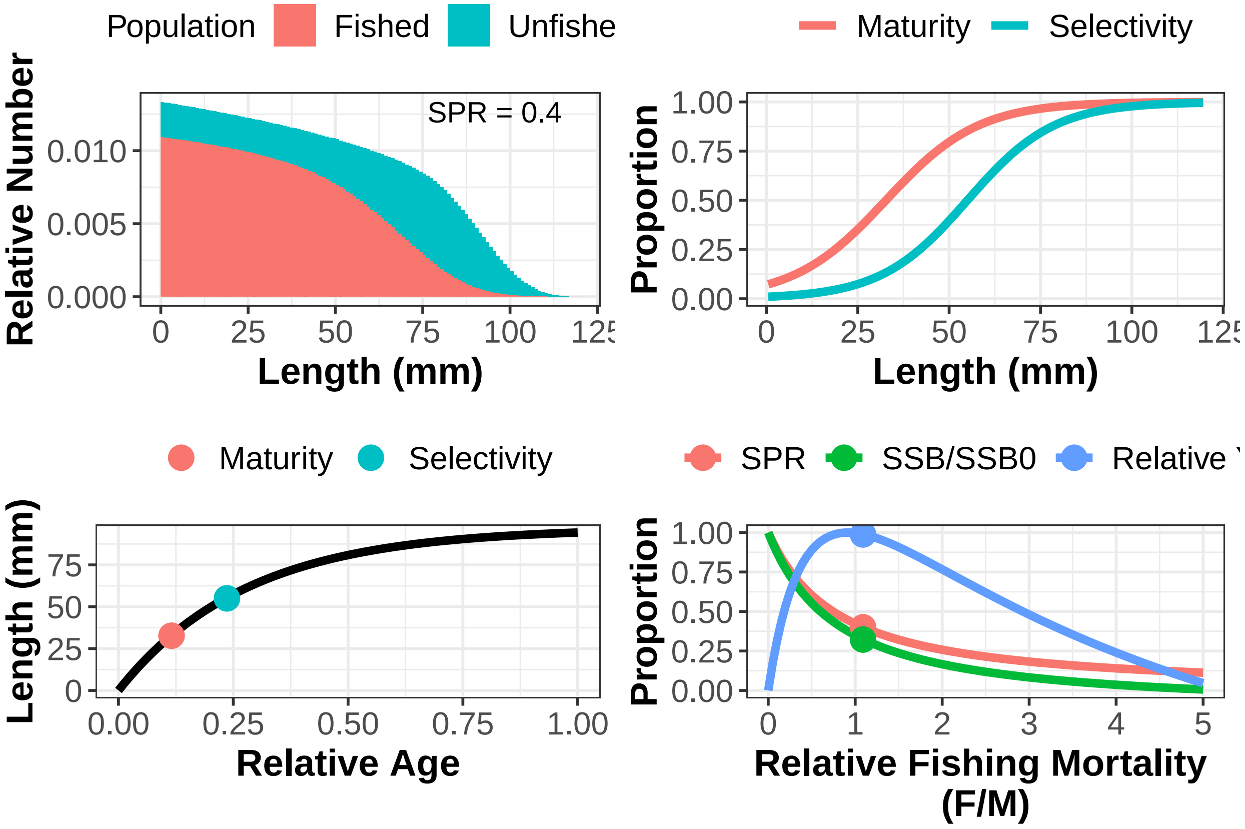


Figure 2: Ploteo de Simulación Population.

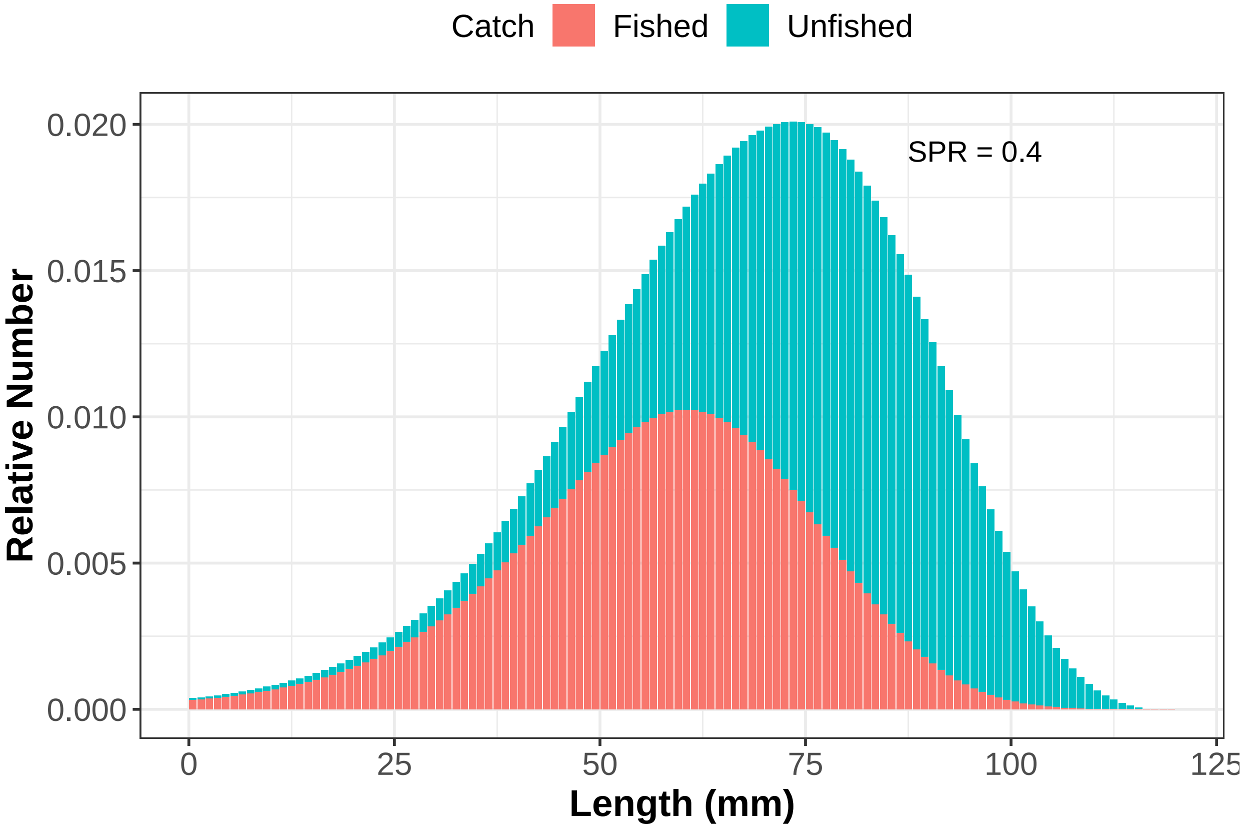


Figure 3: Plot Leng Freq

4 Fitting Empirical Length Data

Two objects are required to fit the LBSPR model to length data: LB_pars which contains the life-history parameters (described above) and LB_lengths, which contains the length frequency data.

4.1 Creating a LB_lengths object

A LB_lengths object can be created in two ways. The new function can be used to create an empty object which can be manually populated:

```
MyLengths <- new("LB_lengths")
```

```
## File not found. A blank LB_lengths object created
```

```
slotNames(MyLengths)
```

```
## [1] "LMids"   "LData"   "L_units" "Years"   "NYears"  "Elog"
```

However, it is probably easier to create the LB_lengths object by directly reading in a CSV file.

Now, we need set our directory

```
datdir <- setwd("C:/Users/mauricio.mardones/Documents/IFOP/Prog_Seg_PM_Bentónicos_2018/X_XI_Assessment/
```

4.2 Reading in Example CSV

A valid LB_pars object must be first created (see sections above):

```
MyPars <- new("LB_pars")
```

```
## A blank LB_pars object created
```

```
## Default values have been set for some parameters
```

```
## A blank LB_pars object created
## Default values have been set for some parameters

MyPars@Species <- "Venus antiqua"
MyPars@Linf <- 97.0#Jerez 1991
MyPars@L50 <- 32.7 #Jaramillo
MyPars@L95 <- 70 # verrificar bibliografia
MyPars@MK <- 0.28/0.218


#Explotacion
MyPars@SL50 <- 55#numeric() #1
MyPars@SL95 <- 90#numeric() #27
MyPars@SPR <- 0.4 #numeric()# ###cambia el numero 0.4 a en blanco
MyPars@BinWidth <- 1
#MyPars@FM <- 1

MyPars@Walpha <- 0.005
MyPars@Wbeta <- 3.0637 #r2 = 0.9651
MyPars@BinWidth <-1
MyPars@BinMax <- 120
MyPars@BinMin <- 0
MyPars@L_units <- "mm"
```

Note that only the life history parameters need to be specified for the estimation model. The exploitation parameters will be estimated.

A length frequency data set with multiple years:

```
Len1 <- new("LB_lengths", LB_pars=MyPars, file=paste0(datdir, "/Frecuencia_tallas_Venus.csv"), dataType
```

Another form to read data is: A length frequency data set with multiple years and a header row (identical to Len1 data, but with a header row):

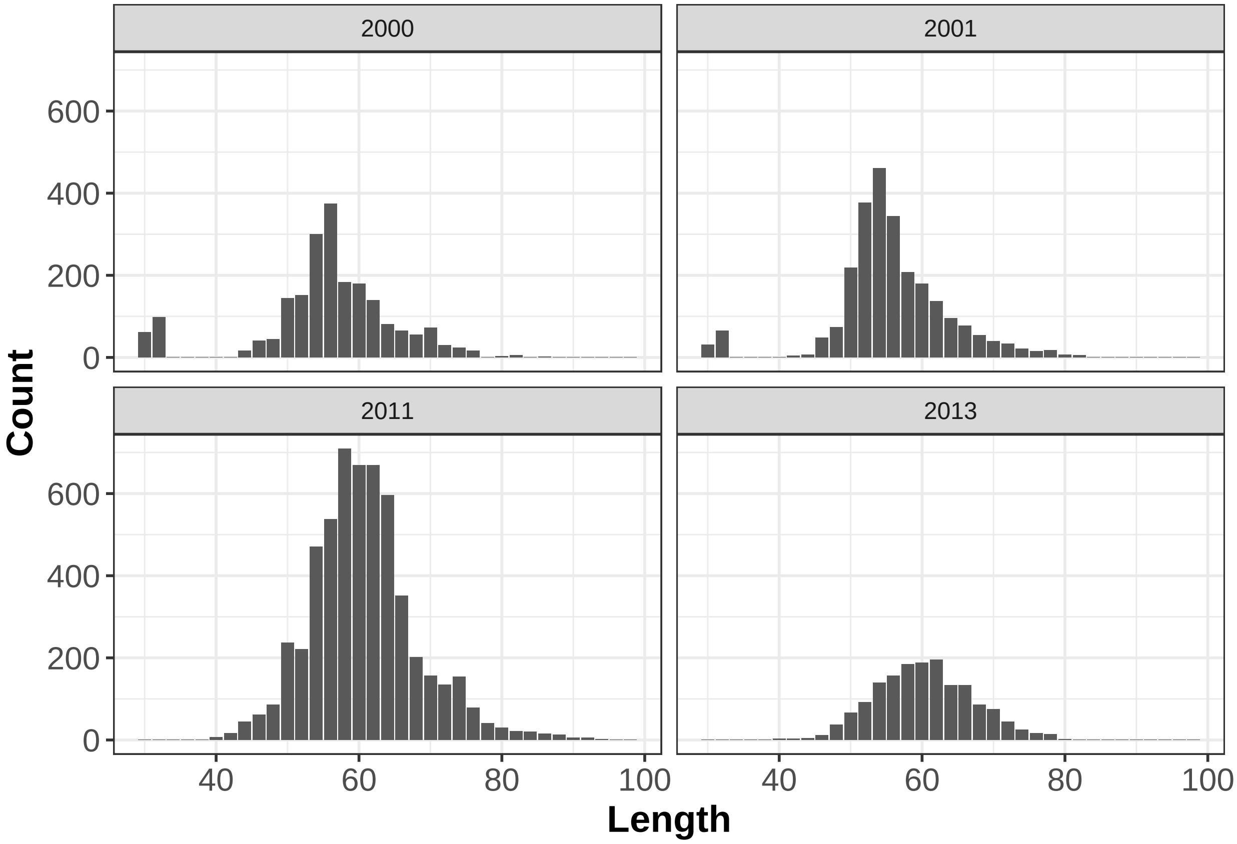
Len2 <- new("LB_lengths", LB_pars=MyPars, file=paste0(datdir, "/LFreq_MultiYrHead.csv"), dataType="freq", header=TRUE) A raw length data set with multiple years:

Len3 <- new("LB_lengths", LB_pars=MyPars, file=paste0(datdir, "/LRaw_MultiYr.csv"), dataType="raw") Length bin parameters (BinMax) must be set for raw data. Using defaults Length bin parameters (BinMin) must be set for raw data. Using defaults Length bin parameters (BinWidth) must be set for raw data. Using defaults Notice that for raw length measurements you must specify the parameters for the length bins (maximum, minimum, and width of length classes) in the LB_pars object. If these are left blank, default values are used.

4.3 Plotting Length Data

The plotSize function can be used to plot the imported length data. This is usually a good idea to do before proceeding with fitting the model, to confirm that everything has been read in correctly:

```
plotSize(Len1)
```



4.4 Fit the Model

The LBSPR model is fitted using the LBSPRfit function:

```
myFit1 <- LBSPRfit(MyPars, Len1)
```

```
## Fitting model

## Year:

## 1

## 2

## 3

## 4
```

to fit another data, myFit2 <- LBSPRfit(MyPars, Len2) Note that the Control argument can be used to modify the additional parameters or LBSPR model type (see description in earlier section).

4.5 Examine and Plot Results

The LBSPR package uses a Kalman filter and the Rauch-Tung-Striebel smoother function (see FilterSmooth) to smooth out the multi-year estimates of SPR, F/M, and selectivity parameters.

The smoother parameter estimates can be accessed from the myFit object (which is an object of class LB_obj [see earlier section for details]):

```
myFit1@Ests
```

```
##       SL50  SL95   FM  SPR
## [1,] 55.72 67.51 4.80 0.21
## [2,] 55.57 66.93 4.68 0.21
## [3,] 55.71 66.73 4.55 0.22
## [4,] 55.84 66.75 4.53 0.22
```

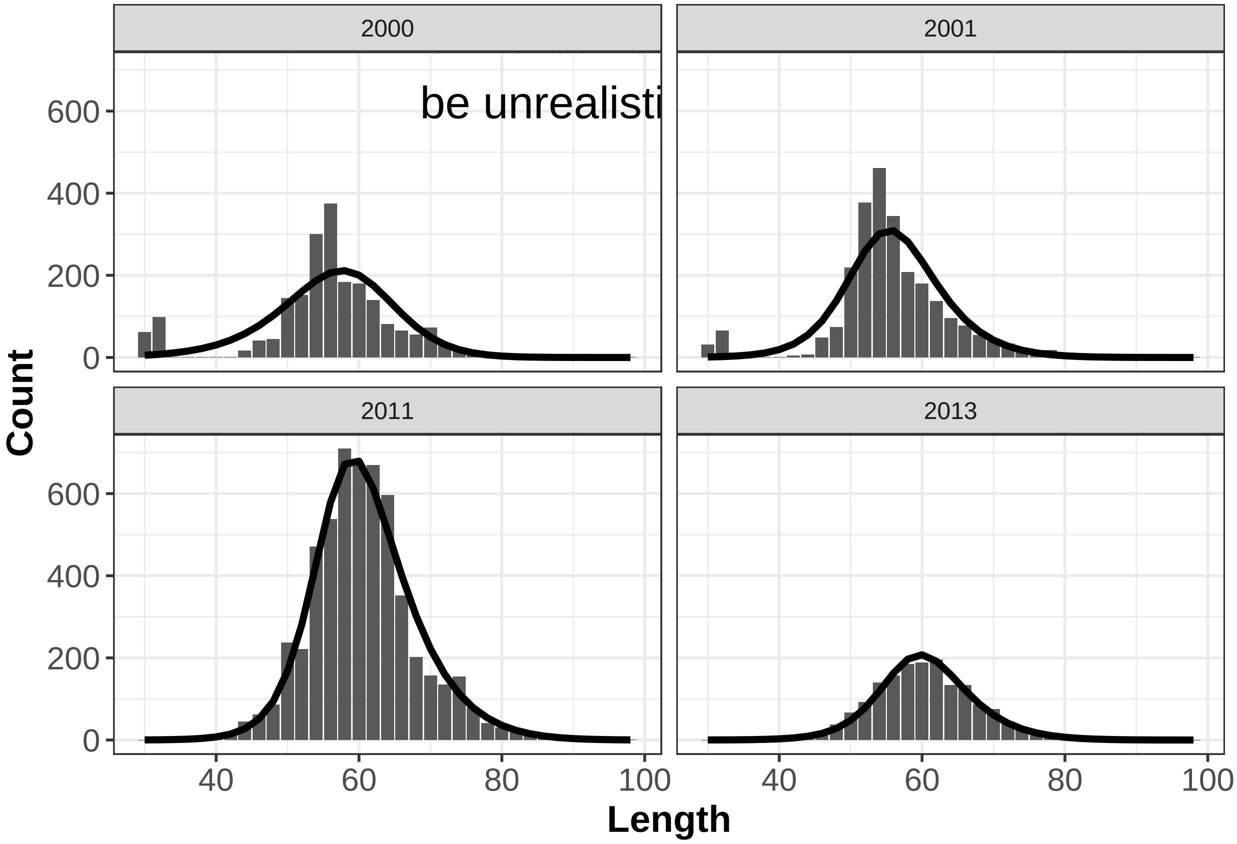
Note that by default the smoothed estimates are used in the plotting routines.

The individual point estimates for each year can be accessed from the LB_obj object:

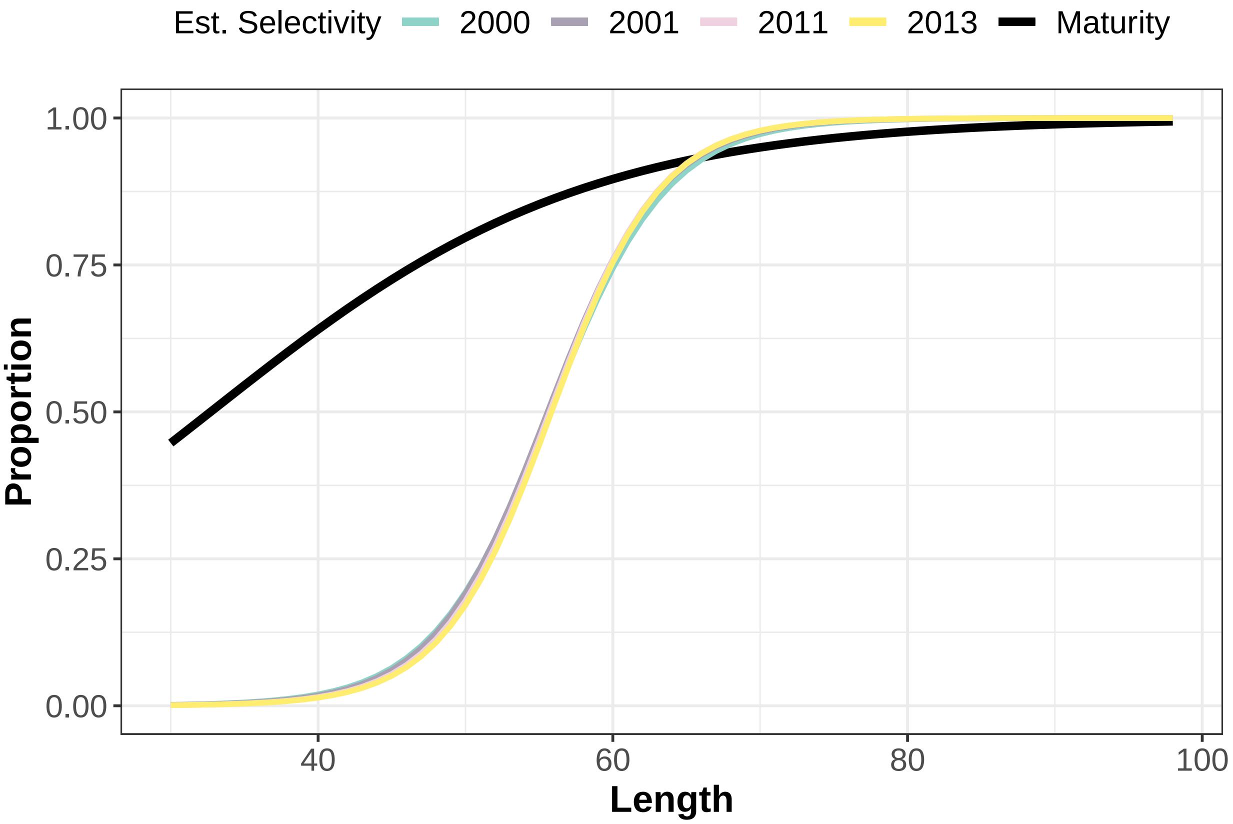
```
data.frame(rawSL50=myFit1@SL50, rawSL95=myFit1@SL95, rawFM=myFit1@FM, rawSPR=myFit1@SPR)
```

```
##   rawSL50 rawSL95 rawFM    rawSPR
## 1   57.75   73.99  6.09 0.1872487
## 2   52.73   63.07  4.77 0.1790576
## 3   55.78   64.65  3.39 0.2549786
## 4   57.14   66.88  4.36 0.2390725
```

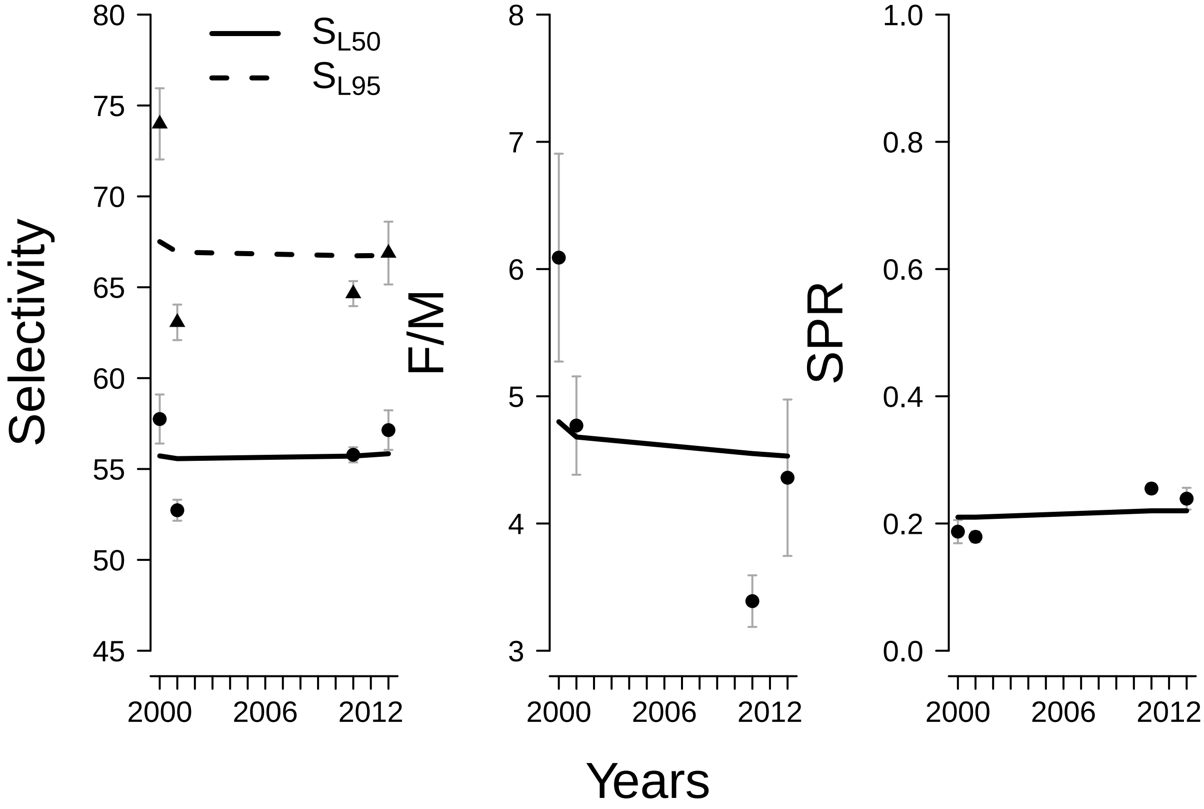
The plotSize function can also be used to show the model fit to the data:



Similarly, the plotMat function can be used to show the specified maturity-at-length curve, and the estimated selectivity-at-length curve:



Finally, the plotEsts function can be used to visually display the estimated parameters. Note that this works for all data sets, but only makes sense when there are several years of data:



By default the plotting function adds the smoother line to the estimated points.

5 Comparing Observed Length Data to Target Size Structure

You can compare the observed size data against an expected size composition at a target SPR using the plotTarg function. To do this, you need a LB_pars object with the life history parameters and the target SPR:

```
MyPars <- new("LB_pars")
```

```
## A blank LB_pars object created
```

```
## Default values have been set for some parameters
```

```
MyPars@Species <- "Venus antiqua"
MyPars@Linf <- 97.0#Jerez 1991
MyPars@L50 <- 32.7 #Jaramillo
MyPars@L95 <- 70 # verrificar bibliografia
MyPars@MK <- 0.28/0.218



#Explotacion
MyPars@SL50 <- 55#numeric() #1
MyPars@SL95 <- 90#numeric() #27
MyPars@SPR <- 0.4#numeric()#0.4 ###cambia el numero 0.4 a en blanco
MyPars@BinWidth <- 1
#MyPars@FM <- 1

MyPars@Walpha <- 0.005
MyPars@Wbeta <- 3.0637 #r2 = 0.9651
MyPars@BinWidth <-2
MyPars@BinMax <- 100
MyPars@BinMin <- 0
MyPars@L_units <- "mm"
MyPars@SPR <- 0.4 # Target SPR
```

Here we have set the target SPR at 40%.

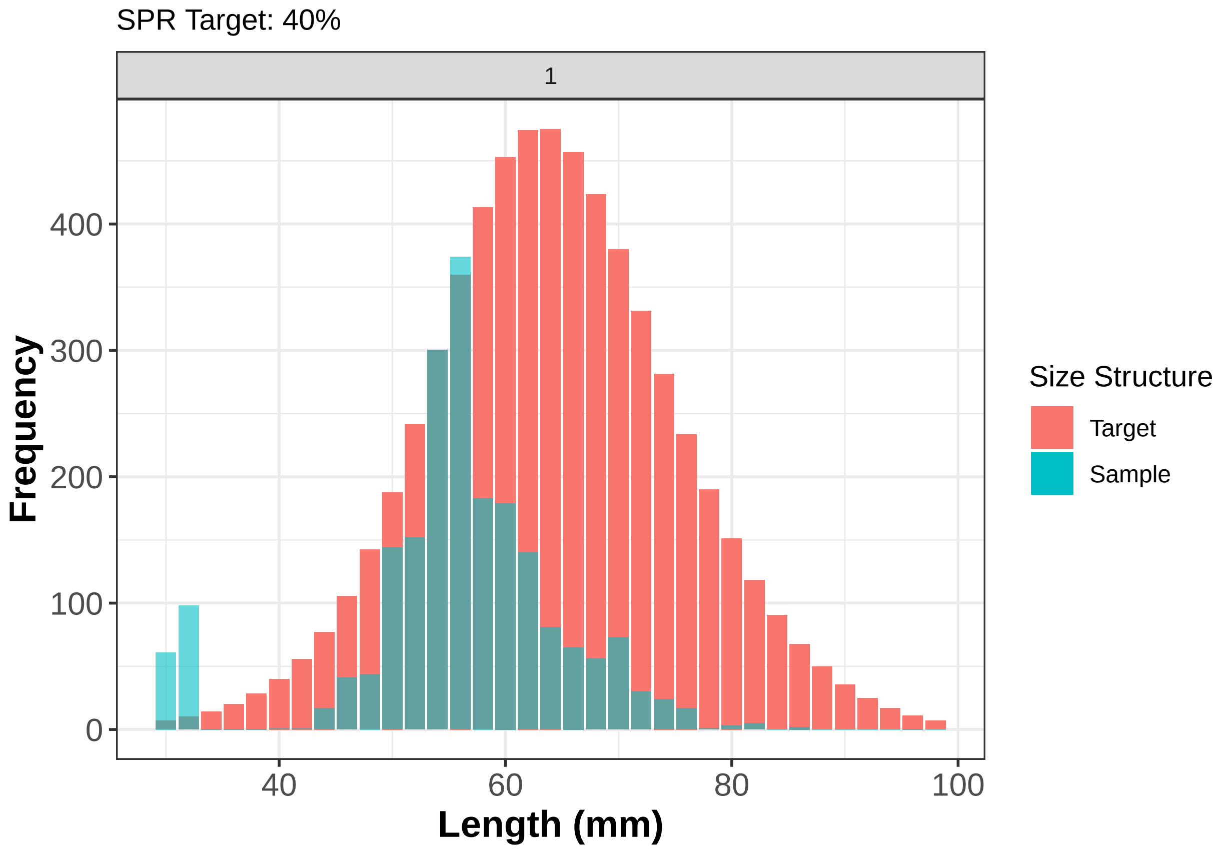
Then import your length data:

```
LenDat <- new("LB_lengths", LB_pars=MyPars, file=paste0(datdir, "/Frecuencia_tallas_Venus.csv"), dataTy
```

Finally, you must set the selectivity parameters for the simulated size data. You may be able to estimate these from the data:

```
Mod <- LBSPRfit(MyPars, LenDat, verbose=FALSE)

yr <- 1 # first year of data
MyPars@SL50 <- Mod@SL50[yr]
MyPars@SL95 <- Mod@SL95[yr]
```


SPR Target: 40%
1
Frequency
0
100
200
300
400
Length (mm)
40
60
80
100
Size Structure
🟥 Target
🟦 Sample

6 References

Hordyk, A.R., Ono, K., Sainsbury, K.J., Loneragan, N., and Prince, J.D. 2015a. Some explorations of the life history ratios to describe length composition, spawning-per-recruit, and the spawning potential ratio. ICES J. Mar. Sci. 72: 204 - 216.

Hordyk, A.R., Ono, K., Valencia, S.R., Loneragan, N.R., and Prince, J.D. 2015b. A novel length-based empirical estimation method of spawning potential ratio (SPR), and tests of its performance, for small-scale, data-poor fisheries. ICES J. Mar. Sci. 72: 217 - 231.

Hordyk, A.R., Loneragan, N.R., and Prince, J.D. 2015c. An evaluation of an iterative harvest strategy for data-poor fisheries using the length-based spawning potential ratio assessment methodology. Fish. Res. 171: 20- 32.

Hordyk, A., Ono, K., Prince, J.D., and Walters, C.J. 2016. A simple length-structured model based on life history ratios and incorporating size-dependent selectivity: application to spawning potential ratios for data-poor stocks. Can. J. Fish. Aquat. Sci. 13: 1- 13. doi: 10.1139/cjfas-2015-0422.

Prince, J.D., Hordyk, A.R., Valencia, S.R., Loneragan, N.R., and Sainsbury, K.J. 2015. Revisiting the concept of Beverton-Holt life-history invariants with the aim of informing data-poor fisheries assessment. ICES J. Mar. Sci. 72: 194 - 203.