Guide to aplicitation of LBPA in data-limited fisheries situations

Length Based Pseudo-Cohort Analysis

Canales, C., Punt, AE., Mardones, M.

2020-10-27

Contents

Introduction

This guide contains functions to compile and run the Length-based PseudoCohort Analysis (LBPA) fisheries stock assessment method in data limited conditions

This model is an estimation model by fitting to length composition data to estimate annual fishing mortality, annual recruitment and spawning potential ratio (SPR)

LBPA was 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., length composition data from the catch) and an understanding of the life history of the species.

Basic functionality

The LBPA model is built based on a C ++ language programmed in ADMB, and below we provide the steps to compile and build from a simple code implemented in R that had next step.

To do this, you must have installed the ADMB version (12 or newer). Can download in this site: http://www.admb-project.org/

The LBPA model and data example can be obtained from https://github.com/CristianCanales/LBPA

Bug Reports

Alert to any bugs or issues by using GitHub. Suggestions and comments for additional features are welcome and we can discuss via email at

Step 1

Set working directory

In this folder you need put two files, .tpl and .dat to run LBPA estimation model

```
rm(list=ls(all=TRUE))  # erasure all objects
setwd('~/')
```

Once you have defined your working directory, you can take a look and check the contains files.

```
##  [1] "admodel.cov"             "admodel.dep"
##  [3] "admodel.hes"             "fmin.log"
##  [5] "LBPA"                    "LBPA User Guide.Rmd"
##  [7] "LBPA User Guide.Rmd.txt" "LBPA-User-Guide.pdf"
##  [9] "LBPA-User-Guide.Rmd"     "LBPA.b01"
## [11] "LBPA.b02"                "LBPA.b03"
## [13] "LBPA.b04"                "LBPA.bar"
## [15] "LBPA.cor"                "LBPA.cpp"
## [17] "lbpa.dat"                "LBPA.eva"
## [19] "LBPA.exe"                "LBPA.htp"
## [21] "LBPA.log"                "LBPA.obj"
## [23] "LBPA.p01"                "LBPA.p02"
## [25] "LBPA.p03"                "LBPA.p04"
## [27] "LBPA.par"                "LBPA.r01"
## [29] "LBPA.r02"                "LBPA.r03"
## [31] "LBPA.r04"                "LBPA.rep"
## [33] "LBPA.Rproj"              "LBPA.std"
## [35] "LBPA.tpl"                "read.admb.R"
## [37] "variance"
```

Is necessary to get a function to can read report. This function is hold in the same folder than .tpl and .dat.

```
source('~/read.admb.R')
```

Step 2

Run model

Now we compile it in ADMB set. If you have read it, dont need run again:

```
system('~/admb-12.2/admb LBPA')
```

Now you need run the model with console. The *system* function could run *.tpl* code from R.

```
system('./LBPA')
```

Step 3

Read Report

Once run our LBPA model, we can read report with *read.rep()* function.

```
data <-read.rep('LBPA.rep')
```

In data example we have a set data from a clam population from Bahía de Ancud, in southern Chile. You can change the data relative to your stock in the *.dat* hosted in the github repository

Now, we can read data in our .rep file.

```
##  [1] "Length"
##  [2] "Observed_frequencies"
##  [3] "Predicted_frequency"
##  [4] "Catch_length_frequency"
##  [5] "Probability_of_length"
##  [6] "Age_Length_SE_N_Catch_Select_F"
##  [7] "Length_frequency_of_exploitable_population"
##  [8] "F_L50_slope_a0_cv_Lr_Ftar"
##  [9] "F/Ftae_SPR"
## [10] "Proportions"
## [11] "Lr"
## [12] "a0"
## [13] "cv"
## [14] "L50"
## [15] "slope"
## [16] "F"
## [17] "SPRtar"
## [18] "Total"
```

We need assign variables to get different plots;

```
BinLen <- data$Length
ObsFre <- data$Observed_frequencies
PredFre <- data$Predicted_frequency
CatchLFre <- data$Catch_length_frequency
ProbLen <- data$Probability_of_length
```

Lets plot size structure from data

```
par(mfrow=c(3,3),mar=c(3,4,1,1)+1)
for(i in 1:9){
  plot(BinLen, ObsFre[i,],type="h", xlab="mm.", ylab="Lenght Density")
  lines(BinLen, PredFre, type="l", col=2)
 }
```

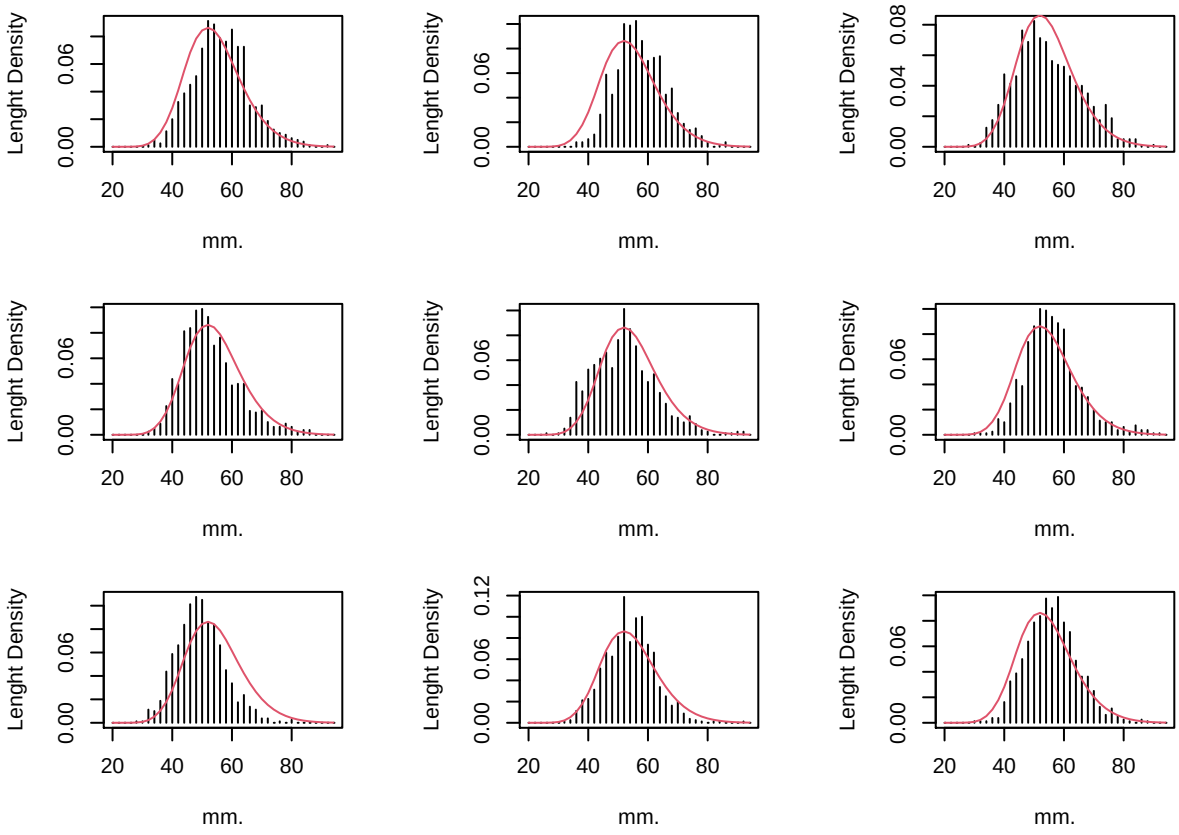

Figure 1: Set of Length data with model adjust

we can do a plot with the sum of all length set data available;

```
plot(BinLen, ObsFre[7,],type="s", xlab="mm.", ylab="Lenght Density")
lines(BinLen, PredFre, type="l", col=2, lwd=4)
legend("topleft", c("Length Predicted", "Length Observed"), col=c(2, 1), lwd =c(1.5, 1.5), lty=c(1,1), l
```

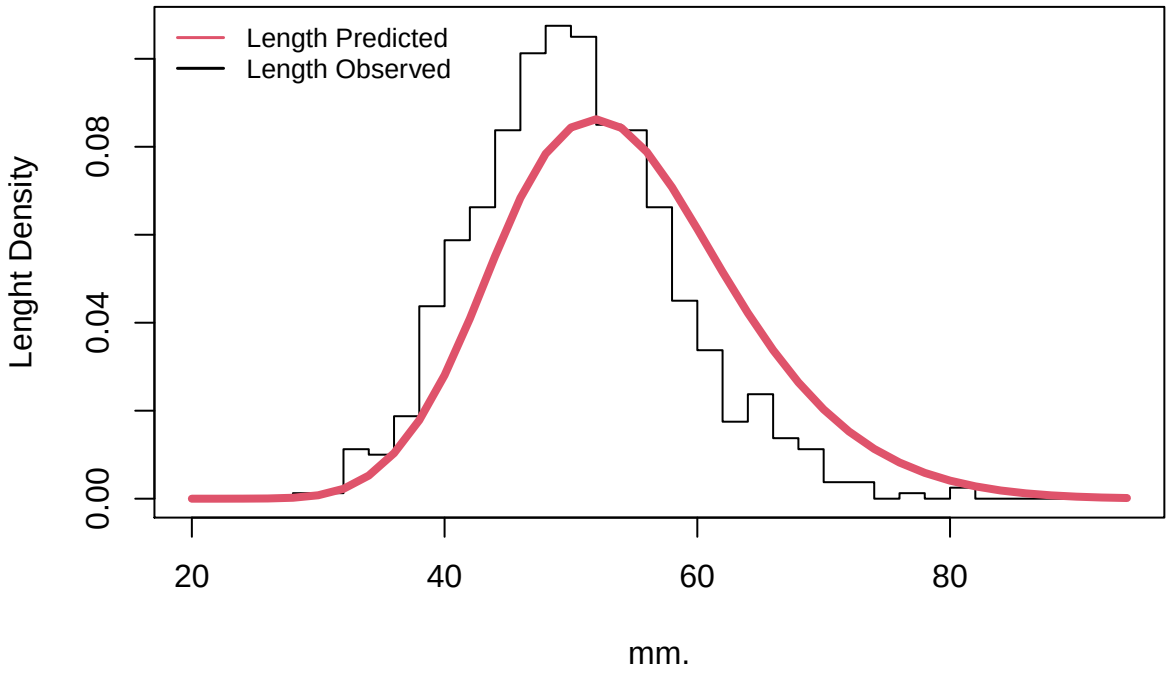

Figure 2: Sum of Length data with model adjust

Step 4

Ploting outputs LBPA

Now we plot diferent outputs, as lengths, adjust models, fishing mortality, selectivity, maturity ogive and SPR.

Plot of estimates parameters

```
p <- ggplot(data=NULL, aes(x=age))+
  geom_line(aes(y = Length,  fill= "#238b45"))+
  theme_bw()
```

```
## Warning: Ignoring unknown aesthetics: fill
```

```
q <- ggplot(data=NULL, aes(x=age))+
  geom_line(aes(y = SE,  fill= "#41b6c4"))+
  theme_bw()
```

```
## Warning: Ignoring unknown aesthetics: fill
```

```
s <- ggplot(data=NULL, aes(x=age))+
  geom_line(aes(y = Fm,  colour = 4))+
  theme_bw()
t <- ggplot(data=NULL, aes(x=age))+
  geom_line(aes(y = Abun,  colour = 5))+
  theme_bw()
o <- ggplot(data=NULL, aes(x=age))+
  geom_line(aes(y = Catch,  colour = 6))+
  theme_bw()
v <- ggplot(data=NULL, aes(x=age))+
  geom_line(aes(y = Selec,  colour = 7))+
  theme_bw()
p/q|s/t|o/v
```

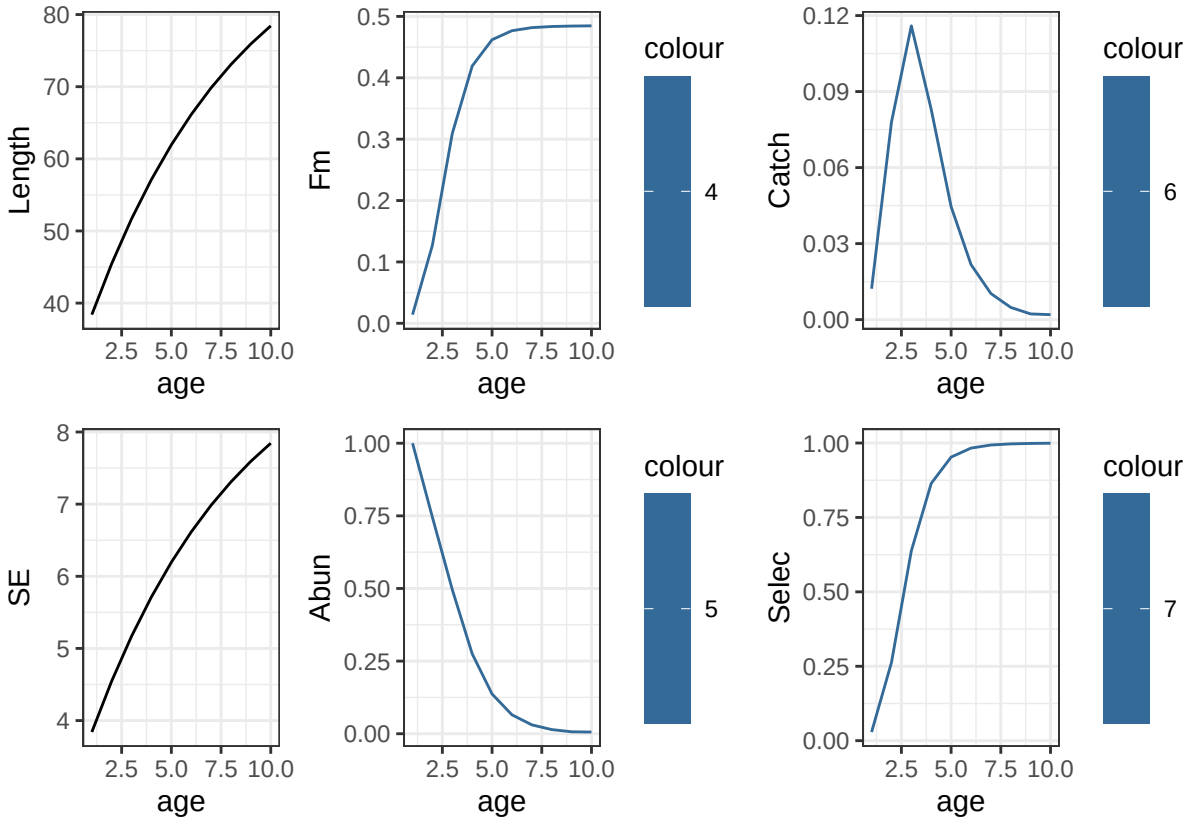

Figure 3: Parameters used in LBPA estimation model

Length current, objetctive and expected

```
L1 <- (data$Length_frequency_of_exploitable_population[1,])
L2 <- (data$Length_frequency_of_exploitable_population[2,])
L3 <- (data$Length_frequency_of_exploitable_population[3,])
```

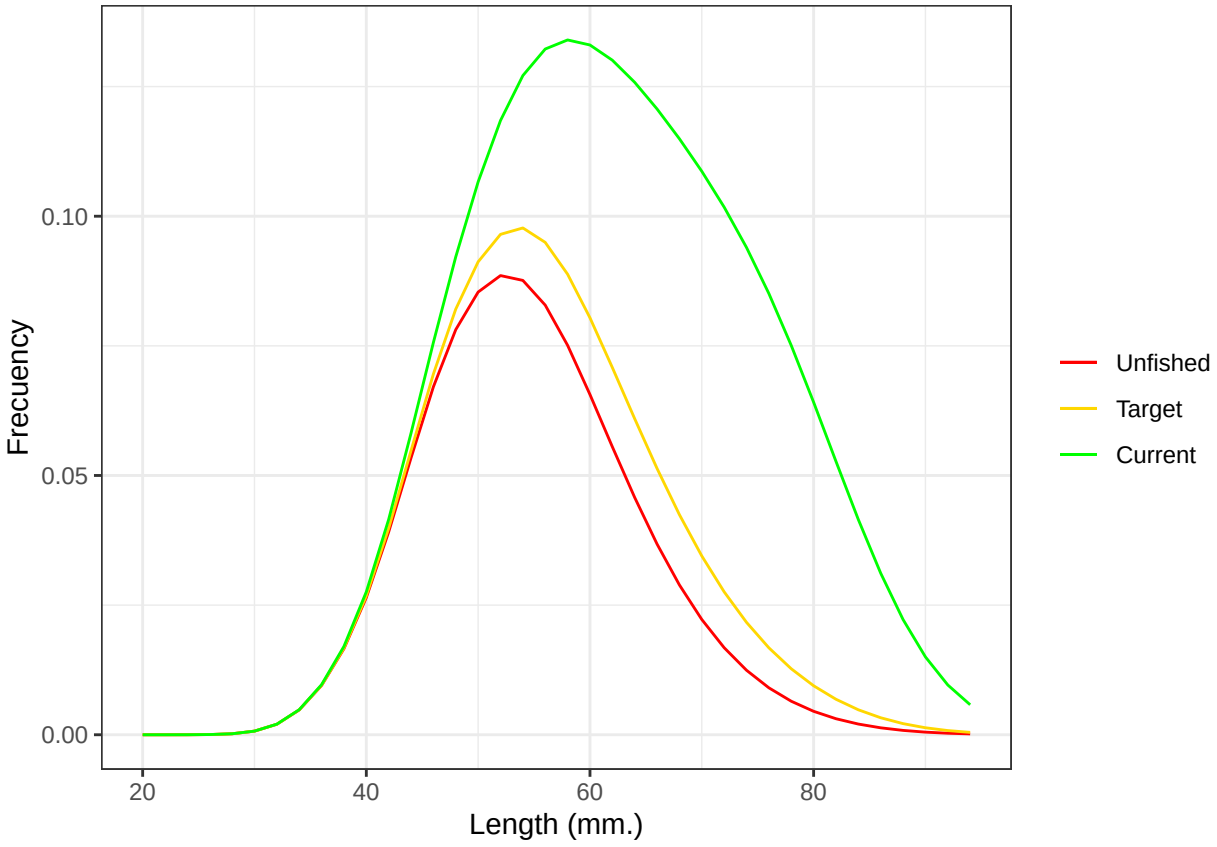

Figure 4: Set of Length estimated in LBPA

Table Parameters Models used in LBPA estimation model;

```
tabla <- matrix(ncol=1, round(data$F_L50_slope_a0_cv_Lr_Ftar, 3))
rownames(tabla) <- c("F", "L50", "Slope", "a0", "cv", "Lr", "F_Target")
kable(tabla)
```

F	0.485
L50	49.363
Slope	6.900
a0	0.001
cv	0.100
Lr	38.399
F_Target	0.327