Exercise4\_answers.R

2022-12-11

#Exercise 4 Answer Key  
library(FSA)

## ## FSA v0.9.3. See citation('FSA') if used in publication.  
## ## Run fishR() for related website and fishR('IFAR') for related book.

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(ggplot2)  
library(nlstools)

##   
## 'nlstools' has been loaded.

## IMPORTANT NOTICE: Most nonlinear regression models and data set examples

## related to predictive microbiolgy have been moved to the package 'nlsMicrobio'

#1. Calculate standard weight and relative weight of Cisco using the “CiscoTL”   
#data set in the FSAdata package. Add these two columns to the CiscoTL data frame.  
#Load the “CiscoTL” data set using:  
CiscoTL <- FSAdata::CiscoTL  
  
#Return a simplified object for calculation  
wsCIS <- wsVal("Cisco", units=c("metric"),   
 simplify = TRUE)  
  
#Add Ws and Wr column  
CiscoTL <- CiscoTL %>%  
 mutate(logW = log10(weight),  
 logL = log10(length),  
 Ws = 10 ^(wsCIS[["int"]] + wsCIS[["slope"]] \* logL ),  
 Wr = weight/Ws \* 100)  
  
#Note there are missing weight which will result in NA being  
#returned for relative weights.  
  
#2. Use the following data to create a scatterplot of count (y-axis) vs   
#age (x-axis) and determine instantaneous total mortality (Z)  
  
#Create a data frame for say, Brook Trout  
bkt <- data.frame(age=1:8,  
 ct=c(74,210,165,92,82,50,25,10))  
bkt

## age ct  
## 1 1 74  
## 2 2 210  
## 3 3 165  
## 4 4 92  
## 5 5 82  
## 6 6 50  
## 7 7 25  
## 8 8 10

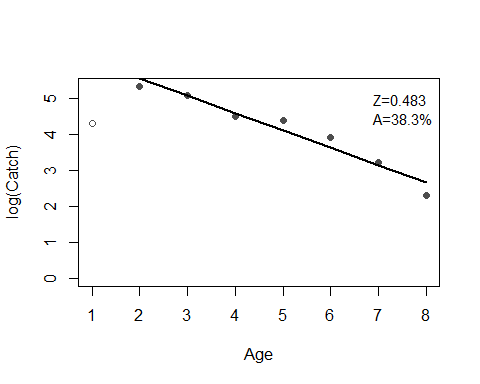
#Create a scatterplot with log(ct) to   
#identify the descending limb of the catch curve  
ggplot(bkt, aes(x = age, y = log(ct))) +  
 geom\_point() +  
 xlab("Age (years)") +   
 ylab("ln(count)")



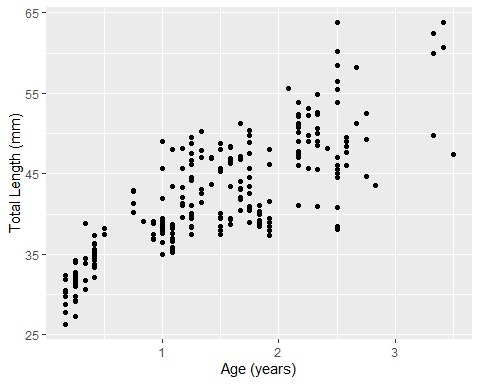
bktcc <- catchCurve(ct ~ age, data = bkt, ages2use=2:8)  
  
#The summary function will return  
#the instantaneous mortality (Z)  
#and annual mortality (A)  
summary(bktcc)

## Estimate Std. Error t value Pr(>|t|)  
## Z 0.4827669 0.05027943 9.601678 0.0002076709  
## A 38.2926331 NA NA NA

#the plot() function will create a plot of the   
#catch curve  
plot(bktcc)



#3. Create a scatterplot and determine parameters of the von Bertalanffy growth model   
#with 95% confidence intervals using “Bonito” data set in the FSAdata package.   
#Bonito length is recorded as “fl”.  
  
Bonito <- FSAdata::Bonito  
  
ggplot(Bonito, aes(x = age, y = fl)) +  
 geom\_point() +  
 xlab("Age (years)") +   
 ylab("Total Length (mm)")



vbT <- vbFuns("typical", simple=FALSE)  
fitBonito <- nls(fl ~ vbT(age, Linf ,K, t0),  
 data=Bonito,  
 start=vbStarts(fl ~ age,   
 data = Bonito,   
 type="typical"))  
  
sumBonito <- summary(fitBonito, correlation = TRUE)  
sumBonito

##   
## Formula: fl ~ vbT(age, Linf, K, t0)  
##   
## Parameters:  
## Estimate Std. Error t value Pr(>|t|)   
## Linf 77.3176 17.2242 4.489 1.10e-05 \*\*\*  
## K 0.2161 0.1069 2.022 0.0443 \*   
## t0 -2.2784 0.5525 -4.124 5.09e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 4.256 on 248 degrees of freedom  
##   
## Correlation of Parameter Estimates:  
## Linf K   
## K -1.00   
## t0 -0.95 0.98  
##   
## Number of iterations to convergence: 6   
## Achieved convergence tolerance: 3.218e-06

coefBonito <- coef(fitBonito)  
coefBonito

## Linf K t0   
## 77.3175630 0.2160503 -2.2784486

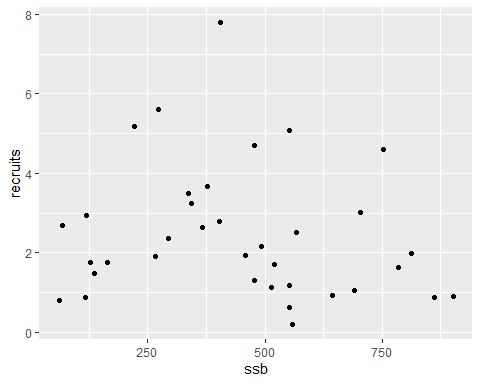
confinT <- confint2(fitBonito)  
confinT

## 2.5 % 97.5 %  
## Linf 43.393099635 111.2420265  
## K 0.005571408 0.4265292  
## t0 -3.366715768 -1.1901814

#4. Estimate parameters of the Ricker stock-recruitment model using the HerringBWE   
#data set from the FSAdata package  
HerringSR <- FSAdata::HerringBWE  
  
headtail(HerringSR)

## spawning.year ssb recruits  
## 1 1962 163.5 1.770  
## 2 1963 221.8 5.182  
## 3 1964 272.5 5.607  
## 34 1995 690.7 1.051  
## 35 1996 458.8 1.936  
## 36 1997 376.4 3.670

#Plot spawners (stock) vs recruits  
ggplot(HerringSR, aes(x = ssb, y = recruits)) +  
 geom\_point()



#Ricker function  
#E[R|S] = alpha \* S \* exp(-beta \* S)  
  
#We will use the nls function to fit this   
#non-linear model  
#Requires starting values  
svR <- srStarts(recruits ~ ssb,   
 data = HerringSR,   
 type = "Ricker")  
svR

## $a  
## [1] 0.02485303  
##   
## $b  
## [1] 0.003467873

#Obtain Ricker function from FSA  
rckr <- srFuns("Ricker")  
  
#Fit Ricker function to stock and recruitment data  
srR <- nls(recruits ~ rckr(ssb,a,b),   
 data = HerringSR,   
 start=svR)  
summary(srR)

##   
## Formula: recruits ~ rckr(ssb, a, b)  
##   
## Parameters:  
## Estimate Std. Error t value Pr(>|t|)   
## a 0.0299247 0.0073916 4.048 0.000282 \*\*\*  
## b 0.0034234 0.0005659 6.049 7.43e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.557 on 34 degrees of freedom  
##   
## Number of iterations to convergence: 3   
## Achieved convergence tolerance: 1.935e-06

#Coefficients with 95% Confidence Intervals  
cbind(estimates=coef(srR), confint(srR))

## Waiting for profiling to be done...

## estimates 2.5% 97.5%  
## a 0.029924748 0.017225827 0.047195351  
## b 0.003423401 0.002297006 0.004603326

#5. Create a scatterplot with predictions of the Ricker model fit in 4 above.   
#Note, select an appropriate break in the range of spawning stock biomass for predictions.  
#Visualize the model fit  
#Range of spawning stock  
x <- seq(from=min(HerringSR$ssb),   
 to = max(HerringSR$ssb),   
 by= 10)  
  
#Predict recruitment from model fit above  
pR<- rckr(x, a=coef(srR))  
#combine in a data frame  
CombSR <- data.frame(x = x, pR = pR)  
  
#Plot predictions with raw data  
ggplot() +  
 geom\_point(aes(x = CombSR$x, y = CombSR$pR),   
 color = "orange", size = 3) +  
 geom\_point(aes(x = HerringSR$ssb,   
 y = HerringSR$recruits)) +  
 xlab("Spawning stock biomass") +   
 ylab("Predicted recruitment")

