**DataSet S1.** The data below were used to create Figure 5 and 6 and parameters in Table 1. Yearly data (t = 1 and dt = 1), Season 1 data (t = 1.5 and dt = 0.5), Season 2 data (t = 1 and dt = 0.5, initial jaw size in mm (JS\_start), final jaw size in mm (JS\_end), final test diameter in mm (TD\_end)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **t** | **dt** | **JS\_start** | **JS\_end** | **TD\_end** |
| 1 | 1 | 7.683 | 8.481 | 38.24 |
| 1 | 1 | 6.712 | 7.730 | 33.82 |
| 1 | 1 | 6.512 | 7.729 | 34.38 |
| 1 | 1 | 7.427 | 7.925 | 33.35 |
| 1 | 1 | 6.255 | 7.432 | 35.85 |
| 1 | 1 | 2.954 | 5.574 | 25.59 |
| 1 | 1 | 5.993 | 7.333 | 35.89 |
| 1 | 1 | 8.826 | 9.061 | 37.44 |
| 1 | 1 | 7.408 | 7.879 | 34.08 |
| 1 | 1 | 8.372 | 8.451 | 42.74 |
| 1 | 1 | 5.560 | 7.065 | 34.26 |
| 1 | 1 | 7.788 | 8.204 | 32.98 |
| 1 | 1 | 8.206 | 8.952 | 40.16 |
| 1 | 1 | 7.165 | 7.749 | 32.67 |
| 1 | 1 | 6.356 | 7.618 | 34.24 |
| 1 | 1 | 8.544 | 8.701 | 34.30 |
| 1 | 1 | 9.448 | 9.663 | 35.83 |
| 1 | 1 | 7.378 | 8.334 | 34.93 |
| 1 | 1 | 7.035 | 7.501 | 35.66 |
| 1 | 1 | 5.932 | 6.693 | 32.80 |
| 1 | 0.5 | 7.051 | 7.505 | 32.35 |
| 1 | 0.5 | 8.580 | 8.761 | 39.01 |
| 1 | 0.5 | 8.602 | 8.856 | 34.97 |
| 1 | 0.5 | 8.354 | 8.526 | 31.17 |
| 1 | 0.5 | 8.301 | 8.506 | 33.47 |
| 1 | 0.5 | 7.813 | 8.018 | 32.03 |
| 1 | 0.5 | 2.990 | 4.977 | 23.85 |
| 1 | 0.5 | 6.661 | 6.996 | 31.87 |
| 1 | 0.5 | 7.176 | 7.312 | 33.30 |
| 1 | 0.5 | 8.355 | 8.563 | 38.52 |
| 1 | 0.5 | 7.110 | 7.412 | 33.88 |
| 1 | 0.5 | 7.432 | 7.763 | 33.03 |
| 1 | 0.5 | 8.092 | 8.190 | 36.95 |
| 1 | 0.5 | 7.164 | 7.336 | 30.98 |
| 1 | 0.5 | 2.885 | 4.338 | 18.86 |
| 1 | 0.5 | 7.976 | 8.192 | 34.11 |
| 1 | 0.5 | 7.972 | 8.225 | 35.90 |
| 1 | 0.5 | 7.290 | 7.380 | 36.72 |
| 1 | 0.5 | 7.291 | 7.577 | 30.39 |
| 1 | 0.5 | 7.250 | 7.844 | 31.80 |
| 1 | 0.5 | 6.632 | 7.313 | 31.71 |
| 1 | 0.5 | 5.151 | 5.786 | 29.44 |
| 1 | 0.5 | 7.727 | 7.898 | 31.31 |
| 1 | 0.5 | 8.847 | 8.897 | 40.60 |
| 1 | 0.5 | 6.937 | 7.219 | 36.34 |
| 1 | 0.5 | 7.566 | 7.859 | 33.07 |
| 1 | 0.5 | 7.822 | 8.015 | 32.46 |
| 1 | 0.5 | 8.377 | 8.583 | 40.00 |
| 1 | 0.5 | 7.437 | 7.565 | 34.44 |
| 1 | 0.5 | 8.181 | 8.505 | 34.81 |
| 1 | 0.5 | 8.191 | 8.415 | 34.53 |
| 1 | 0.5 | 3.042 | 4.587 | 21.96 |
| 1 | 0.5 | 8.217 | 8.320 | 32.61 |
| 1 | 0.5 | 7.521 | 7.648 | 35.49 |
| 1 | 0.5 | 7.683 | 7.968 | 38.24 |
| 1 | 0.5 | 6.712 | 7.018 | 33.82 |
| 1 | 0.5 | 6.512 | 6.849 | 34.38 |
| 1 | 0.5 | 7.427 | 7.643 | 33.35 |
| 1 | 0.5 | 6.255 | 6.535 | 35.85 |
| 1.5 | 0.5 | 6.292 | 7.170 | 31.87 |
| 1.5 | 0.5 | 7.388 | 7.480 | 35.90 |
| 1.5 | 0.5 | 6.841 | 7.335 | 37.76 |
| 1.5 | 0.5 | 7.683 | 8.196 | 38.24 |
| 1.5 | 0.5 | 6.712 | 7.424 | 33.82 |
| 1.5 | 0.5 | 6.512 | 7.392 | 34.38 |
| 1.5 | 0.5 | 7.427 | 7.709 | 33.35 |
| 1.5 | 0.5 | 6.255 | 7.152 | 35.85 |

**Syntax and script**

The code below was used to create Figure 6 and parameters in Table 1 (R environment version 3.6).

#####################################################################

#### **Figure 6**

#####################################################################

#####################################################################

#### **Jaw length**

#####################################################################

####

#

# Column names:

# t (time)

# dt (season)

# JS\_start (initial jaw length)

# JS\_end (final jaw length)

# TD\_end (Final test diameter)

#

# File name:

# urriago.csv

#

#fileloc <- Use location of the file on your computer

#

#####################################################################

library(ggplot2)

fileloc <- 'C:/Users/LENOVO/Desktop/urriago.csv'

dat <- read.csv(fileloc, header=TRUE, stringsAsFactors=FALSE)

#####################################################################

# Compute Jaw length at the beginning

#####################################################################

JS\_change <- with(dat, JS\_end- JS\_start)

Dev\_JS\_end <- dat$JS\_end - mean(dat$JS\_end)

Dev\_JS\_start <- dat$JS\_start - mean(dat$JS\_start)

m0 <- sum(Dev\_JS\_end\*Dev\_JS\_start)/sum(Dev\_JS\_start^2)

c0 <- mean(dat$JS\_end)-mean(dat$JS\_start)\*m0

K0 <- -log(m0)

Jinf0 <- c0/(1-m0)

C0 <- 1

ts0 <- 0

J0 <- 0.025

D0 <- 0.2

seavec <- c(K0, Jinf0, C0, ts0)

nonseavec <- c(K0, Jinf0)

#####################################################################

# Objective function

# for parameters in seasonal model

#####################################################################

fb <- function(x) {

yhat <- with(dat,

(x[2]-JS\_start)\*(1-exp(-(x[1]\*dt-(x[3]\*x[1]/(2\*pi)\*sin(2\*pi\*(t-x[4])))+

(x[3]\*x[1]/(2\*pi)\*sin(2\*pi\*(t+dt-x[4])))))))

sum((JS\_change-yhat)^2)

}

#####################################################################

# Objective function

# for parameters in non-seasonal model

#####################################################################

fc <- function(x) {

yhat <- with(dat,

(x[2]-JS\_start)\*(1-exp(-(x[1]\*dt))))

sum((JS\_change-yhat)^2)

}

mod2 <- optim(seavec, fb, method="BFGS")

seapar <- data.frame(K=mod2$par[1], Jinf=mod2$par[2], C=mod2$par[3], ts=mod2$par[4])

#####################################################################

# Objective function for seasonal t0

#####################################################################

fd <- function(x) {

(seapar$K\*x-(seapar$C\*seapar$K)/(2\*pi)\*sin(2\*pi\*seapar$ts)-

(seapar$C\*seapar$K)/(2\*pi)\*sin(2\*pi\*(x-seapar$ts))-log(1-D0/seapar$Jinf))^2

}

mod3 <- optim(0, fd, method="BFGS")

seapar$t0 <- mod3$par[1]

print("Estimated seasonal model parameters")

print(round(seapar,4), row.names=FALSE)

mod4 <- optim(nonseavec, fc, method="BFGS")

nonseapar <- data.frame(K=mod4$par[1], Jinf=mod4$par[2], C=0, ts=0)

#####################################################################

# Objective function for non-seasonal t0

#####################################################################

fe <- function(x) {

(nonseapar$K\*x-(nonseapar$C\*nonseapar$K)/(2\*pi)\*sin(2\*pi\*nonseapar$ts)-

(nonseapar$C\*nonseapar$K)/(2\*pi)\*sin(2\*pi\*(x-nonseapar$ts))-log(1-J0/nonseapar$Jinf))^2

}

mod5 <- optim(0, fe, method="BFGS")

nonseapar$t0 <- mod5$par[1]

print("Estimated non-seasonal mOdel parameters")

print(round(nonseapar,4), row.names=FALSE)

#####################################################################

# Growth curve of seasonal model

#####################################################################

ageMax <- 22

age <- seq(0,ageMax,.1)

sea\_JS <- rep(0,length(age))

sea\_JS[1] <- J0

for (i in 2:length(age)) {

JS\_tmp <-

seapar$K\*0.1-(seapar$C\*seapar$K)/(2\*pi)\*sin((2\*pi)\*(age[i-1]-seapar$ts))+(seapar$C\*seapar$K)/(2\*pi)\*sin((2\*pi)\*(age[i]-seapar$ts))

sea\_JS[i] <- seapar$Jinf\*(1-exp(-JS\_tmp))+exp(-JS\_tmp)\*sea\_JS[i-1]

}

#####################################################################

# Growth curve of non-seasonal model

#####################################################################

nonsea\_JS <- rep(0,length(age))

nonsea\_JS[1] <- J0

for (i in 2:length(age)) {

JS\_tmp <-

nonseapar$K\*(age[i]-nonseapar$t0)-(nonseapar$C\*nonseapar$K)/(2\*pi)\*sin((2\*pi)\*(age[i]-nonseapar$ts))+(nonseapar$C\*nonseapar$K)/(2\*pi)\*sin((2\*pi)\*(nonseapar$t0-nonseapar$ts))

nonsea\_JS[i] <- nonseapar$Jinf\*(1-exp(-JS\_tmp))

}

#####################################################################

# Figure 6. Seasonal and non-seasonal models (Jaw Size/JS)

#####################################################################

ggplot() +

geom\_line(aes(x=age[age <= 10], y=sea\_JS[age <= 10], color='Seasonal'), lwd=0.8) +

geom\_line(aes(x=age[age <= 10], y=nonsea\_JS[age <= 10], color='Non-seasonal'), lwd=0.8, linetype='dashed') +

scale\_color\_manual(name='Model', values=c('Seasonal'='blue', 'Non-seasonal'='red')) +

scale\_x\_continuous(breaks=seq(0,10,1)) +

ggtitle('Jaw lenght vs Age') +

ylab("Jaw lenght (mm)") +

xlab("Age (years)") +

theme\_bw() +

theme(plot.title=element\_text(hjust=0.5))

#####################################################################

#JS and Age points (seasonal)

#####################################################################

View(cbind(age[age <= 10], sea\_JS[age <= 10]))

#####################################################################

#JS and Age points (non-seasonal)

#####################################################################

View(cbind(age[age <= 10], nonsea\_JS[age <= 10]))

####

####

#####################################################################

#### **Test diameter**

#####################################################################

####

#

# Column names:

# t (time)

# dt (season)

# JS\_start (initial jaw length)

# JS\_end (final jaw length)

# TD\_end (Final test diameter)

#

# File name:

# urriago.csv

#

#fileloc <- Use location of the file on your computer

#

#####################################################################

fileloc <- 'C:/Users/LENOVO/Desktop/urriago.csv'

dat <- read.csv(fileloc, header=TRUE, stringsAsFactors=FALSE)

#####################################################################

# Estimate allometric parameters based on J\_end and TD\_end

#####################################################################

allo\_dat <- dat[dat$t == 1 & dat$dt == 1,]

lTDend <- log(allo\_dat$TD\_end)

lJSend <- log(allo\_dat$JS\_end)

mod0 <- lm(lTDend ~ lJSend)

avec <- c(exp(coef(mod0)[1]), coef(mod0)[2])

mod1a <- nls(TD\_end ~ I(a\*JS\_end^b), data=allo\_dat, start=list(a=avec[1],b=avec[2]))

summary(mod1a)

rsq2 <- with(allo\_dat, cor(TD\_end, coef(mod1a)[1]\*JS\_end^coef(mod1a)[2])^2)

print(paste("RSQ =", round(rsq2,4), collapse=''))

#####################################################################

# Compute TD at the beginning

#####################################################################

dat$TD\_start <- coef(mod1a)[1]\*dat$JS\_start^coef(mod1a)[2]

TD\_change <- with(dat, TD\_end-TD\_start)

Dev\_TD\_end <- dat$TD\_end - mean(dat$TD\_end)

Dev\_TD\_start <- dat$TD\_start - mean(dat$TD\_start)

m0 <- sum(Dev\_TD\_end\*Dev\_TD\_start)/sum(Dev\_TD\_start^2)

c0 <- mean(dat$TD\_end)-mean(dat$TD\_start)\*m0

K0 <- -log(m0)

Dinf0 <- c0/(1-m0)

C0 <- 1

ts0 <- 0

D0 <- 0.2

seavec <- c(K0, Dinf0, C0, ts0)

nonseavec <- c(K0, Dinf0)

#####################################################################

# Objective function for parameters in seasonal model

#####################################################################

fb <- function(x) {

yhat <- with(dat,

(x[2]-TD\_start)\*(1-exp(-(x[1]\*dt-(x[3]\*x[1]/(2\*pi)\*sin(2\*pi\*(t-x[4])))+

(x[3]\*x[1]/(2\*pi)\*sin(2\*pi\*(t+dt-x[4])))))))

sum((TD\_change-yhat)^2)

}

#####################################################################

# Objective function for parameters in non-seasonal model

#####################################################################

fc <- function(x) {

yhat <- with(dat,

(x[2]-TD\_start)\*(1-exp(-(x[1]\*dt))))

sum((TD\_change-yhat)^2)

}

mod2 <- optim(seavec, fb, method="BFGS")

seapar <- data.frame(K=mod2$par[1], Dinf=mod2$par[2], C=mod2$par[3], ts=mod2$par[4])

#####################################################################

# Objective function for seasonal t0

#####################################################################

fd <- function(x) {

(seapar$K\*x-(seapar$C\*seapar$K)/(2\*pi)\*sin(2\*pi\*seapar$ts)-

(seapar$C\*seapar$K)/(2\*pi)\*sin(2\*pi\*(x-seapar$ts))-log(1-D0/seapar$Dinf))^2

}

mod3 <- optim(0, fd, method="BFGS")

seapar$t0 <- mod3$par[1]

print("Estimated seasonal model parameters")

print(round(seapar,4), row.names=FALSE)

mod4 <- optim(nonseavec, fc, method="BFGS")

nonseapar <- data.frame(K=mod4$par[1], Dinf=mod4$par[2], C=0, ts=0)

#####################################################################

# Objective function for non-seasonal t0

#####################################################################

fe <- function(x) {

(nonseapar$K\*x-(nonseapar$C\*nonseapar$K)/(2\*pi)\*sin(2\*pi\*nonseapar$ts)-

(nonseapar$C\*nonseapar$K)/(2\*pi)\*sin(2\*pi\*(x-nonseapar$ts))-log(1-D0/nonseapar$Dinf))^2

}

mod5 <- optim(0, fe, method="BFGS")

nonseapar$t0 <- mod5$par[1]

print("Estimated non-seasonal mOdel parameters")

print(round(nonseapar,4), row.names=FALSE)

#####################################################################

# Growth curve of seasonal model

#####################################################################

ageMax <- 22

age <- seq(0,ageMax,.1)

sea\_TD <- rep(0,length(age))

sea\_TD[1] <- D0

for (i in 2:length(age)) {

TD\_tmp <-

seapar$K\*0.1-(seapar$C\*seapar$K)/(2\*pi)\*sin((2\*pi)\*(age[i-1]-seapar$ts))+(seapar$C\*seapar$K)/(2\*pi)\*sin((2\*pi)\*(age[i]-seapar$ts))

sea\_TD[i] <- seapar$Dinf\*(1-exp(-TD\_tmp))+exp(-TD\_tmp)\*sea\_TD[i-1]

}

#####################################################################

# Growth curve of non-seasonal model

#####################################################################

nonsea\_TD <- rep(0,length(age))

nonsea\_TD[1] <- D0

for (i in 2:length(age)) {

TD\_tmp <-

nonseapar$K\*(age[i]-nonseapar$t0)-(nonseapar$C\*nonseapar$K)/(2\*pi)\*sin((2\*pi)\*(age[i]-nonseapar$ts))+(nonseapar$C\*nonseapar$K)/(2\*pi)\*sin((2\*pi)\*(nonseapar$t0-nonseapar$ts))

nonsea\_TD[i] <- nonseapar$Dinf\*(1-exp(-TD\_tmp))

}

#####################################################################

# Figure 6. Seasonal and non-seasonal models (Test Diameter/TD)

#####################################################################

ggplot() +

geom\_line(aes(x=age[age <= 10], y=sea\_TD[age <= 10], color='Seasonal'), lwd=0.8) +

geom\_line(aes(x=age[age <= 10], y=nonsea\_TD[age <= 10], color='Non-seasonal'), lwd=0.8, linetype='dashed') +

scale\_color\_manual(name='Model', values=c('Seasonal'='blue', 'Non-seasonal'='red')) +

scale\_x\_continuous(breaks=seq(0,10,1)) +

ggtitle('Test diameter vs Age') +

ylab("Test diameter (mm)") +

xlab("Age (years)") +

theme\_bw() +

theme(plot.title=element\_text(hjust=0.5))

#####################################################################

#TD and Age points seasonal

#####################################################################

View(cbind(age[age <= 10], sea\_TD[age <= 10]))

#####################################################################

#TD and Age points Non-seasonal

#####################################################################

View(cbind(age[age <= 10], nonsea\_TD[age <= 10]))