

# FAS6337C - Lab 3

Marcel Gietzmann-Sanders

## Growth Models in R

Data for this laboratory are from two Florida Spotted Seatrout *Cynoscion nebulosus* populations (Indian River and Charlotte Harbor). The data set contains only females from each population. Fish were collected from 1986 to 1988 by Murphy and Taylor (1994, Transactions of the American Fisheries Society 123:482-497).

```
setwd("/workspaces/schooling/population_dynamics/lab_3/")
trout_data <- read.table("data/trout.txt", header=T, sep="")
head(trout_data)
```

```
##           bay  t1  sex annuli age yearsold
## 1 CharlotteHarbor 387 FALSE      1   1     1.652
## 2 CharlotteHarbor 355 FALSE      1   1     1.652
## 3 CharlotteHarbor 355 FALSE      1   1     1.652
## 4 CharlotteHarbor 320 FALSE      1   1     1.652
## 5 CharlotteHarbor 335 FALSE      1   1     1.652
## 6 CharlotteHarbor 410 FALSE      1   1     1.652
```

The objectives of this laboratory are:

1. To determine the best growth model for each population and estimate  $L_{\infty}$ ,  $t_0$ , and  $k$ ;
2. Estimate the mean length at age for each population;
3. Use an AIC model selection method to compare growth curves and parameters between populations.

Please conduct the following analyses in R and answer the questions. Each question or bullet point should be responded to either with text, a table, or a plot. Use trout.txt as your data file.

**1. Use nonlinear least squares (nls) to estimate the von Bertalanffy growth parameters separately for Charlotte Harbor and for Indian River populations.**

```
ch_data <- na.omit(
  trout_data[trout_data$bay == 'CharlotteHarbor',]
)
ch_data <- ch_data[order(ch_data$yearsold),]
ir_data <- na.omit(
  trout_data[trout_data$bay == 'IndianRiver',]
)
ir_data <- ir_data[order(ir_data$yearsold),]

do_least_squares_fit <- function(fish_data, Linf, vbk, tknot) {
  t1 <- as.numeric(fish_data$t1)
  yearsold <- as.numeric(fish_data$yearsold)
  result <- nls(
    t1 ~ Linf * (1 - exp(-vbk * (yearsold - tknot))),
    data=fish_data,
```

```

    start=list(Linf=Linf, vbk=vbk, tknot=tknot)
  )
  return(result)
}

ch_result = do_least_squares_fit(ch_data, 800, 0.2, -1.4)
ir_result = do_least_squares_fit(ir_data, 950, 0.19, -1.2)

summary(ch_result)

##
## Formula: tl ~ Linf * (1 - exp(-vbk * (yearsold - tknot)))
##
## Parameters:
##      Estimate Std. Error t value Pr(>|t|)
## Linf  788.88587   44.03635  17.914 < 2e-16 ***
## vbk    0.20546    0.02685   7.651 4.34e-14 ***
## tknot -1.45233    0.17197  -8.445 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 42.07 on 1099 degrees of freedom
##
## Number of iterations to convergence: 4
## Achieved convergence tolerance: 2.277e-06

```

```

summary(ir_result)

##
## Formula: tl ~ Linf * (1 - exp(-vbk * (yearsold - tknot)))
##
## Parameters:
##      Estimate Std. Error t value Pr(>|t|)
## Linf  977.2375    63.8731  15.300 < 2e-16 ***
## vbk    0.1894     0.0261   7.259 7.03e-13 ***
## tknot -1.2178     0.1550  -7.857 8.77e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 58.34 on 1192 degrees of freedom
##
## Number of iterations to convergence: 2
## Achieved convergence tolerance: 5.089e-06

```

```

ch_data$nlstl <- ch_result$m$predict(ch_data$yearsold)
ir_data$nlstl <- ir_result$m$predict(ir_data$yearsold)

```

```

ch_data$nlstres <- ch_data$tl - ch_data$nlstl
ir_data$nlstres <- ir_data$tl - ir_data$nlstl

```

```

head(ch_data)

```

```

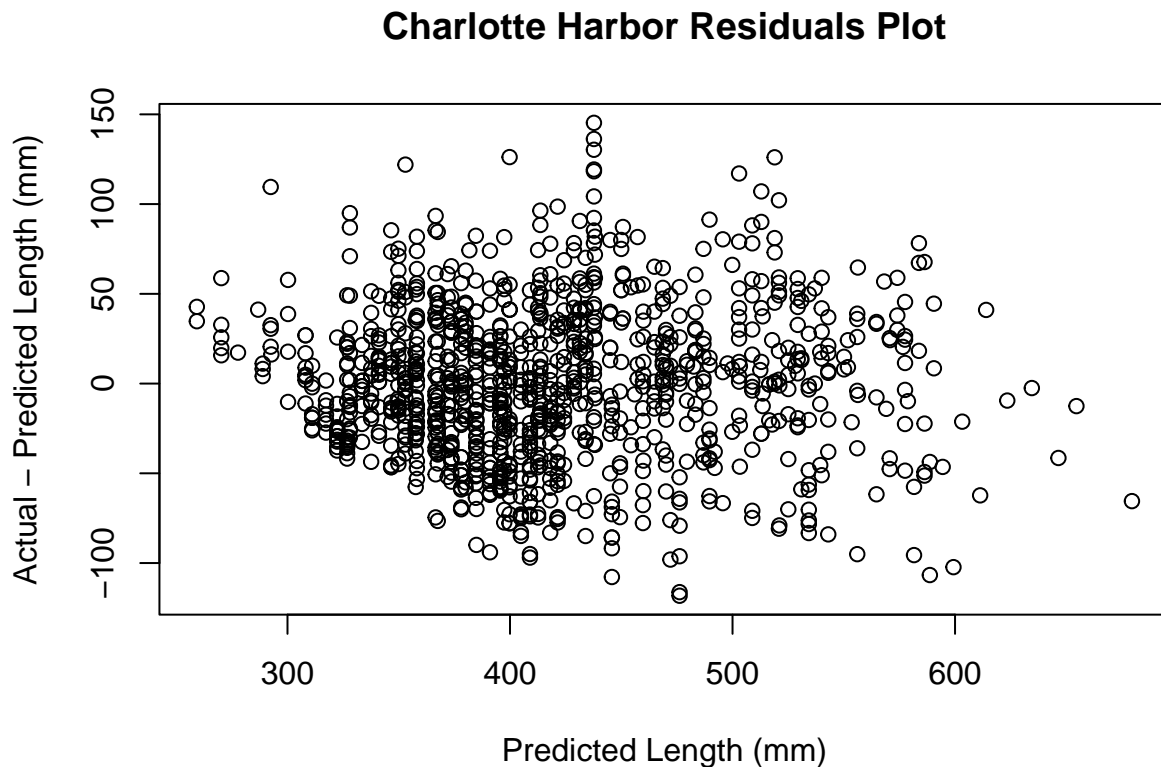
##              bay  tl  sex annuli age yearsold  nlstl  nlstres
## 2150 CharlotteHarbor 302 FALSE      0  0    0.487 259.2628 42.73723
## 2160 CharlotteHarbor 294 FALSE      0  0    0.487 259.2628 34.73723

```

```
## 1165 CharlotteHarbor 329 FALSE      1  0    0.589 270.2466 58.75343
## 1188 CharlotteHarbor 303 FALSE      0  0    0.589 270.2466 32.75343
## 1189 CharlotteHarbor 286 FALSE      0  0    0.589 270.2466 15.75343
## 1190 CharlotteHarbor 296 FALSE      0  0    0.589 270.2466 25.75343
```

Plot the residuals against the predicted values from the nls for each population.

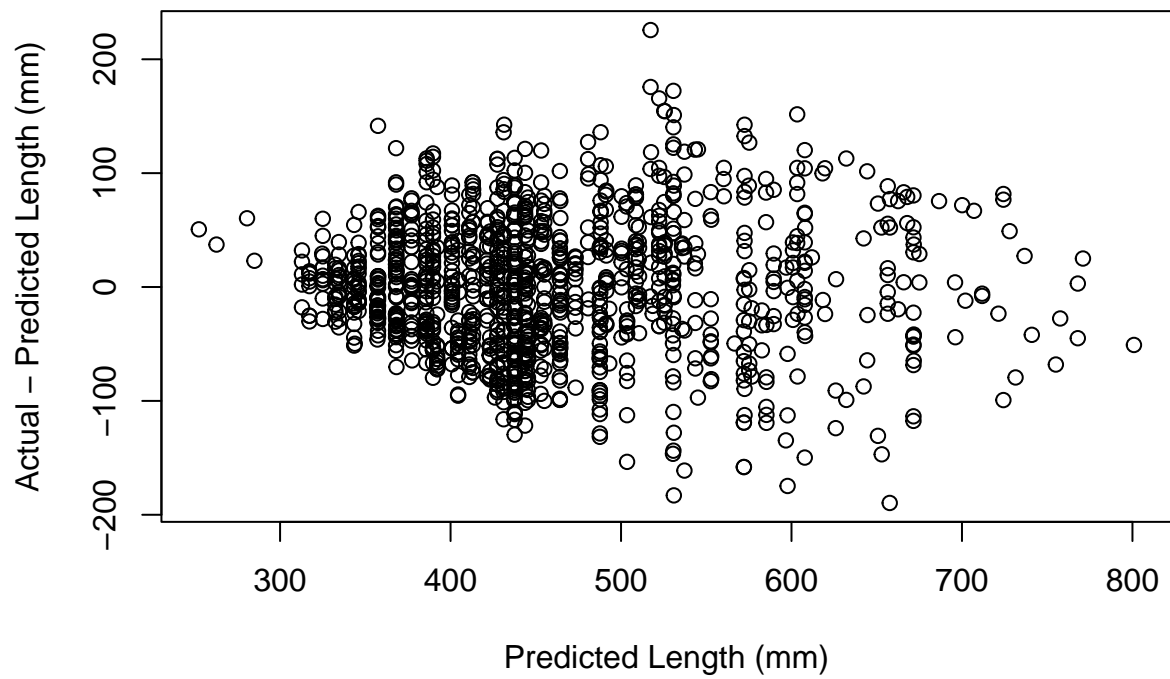
```
with(
  ch_data, {
    plot(
      nls_res ~ nls_tl,
      main='Charlotte Harbor Residuals Plot',
      xlab='Predicted Length (mm)',
      ylab='Actual - Predicted Length (mm)'
    )
  }
)
```



```
with(
  ir_data, {
    plot(
      nls_res ~ nls_tl,
      main='Indian River Residuals Plot',
      xlab='Predicted Length (mm)',
      ylab='Actual - Predicted Length (mm)'
    )
  }
)
```

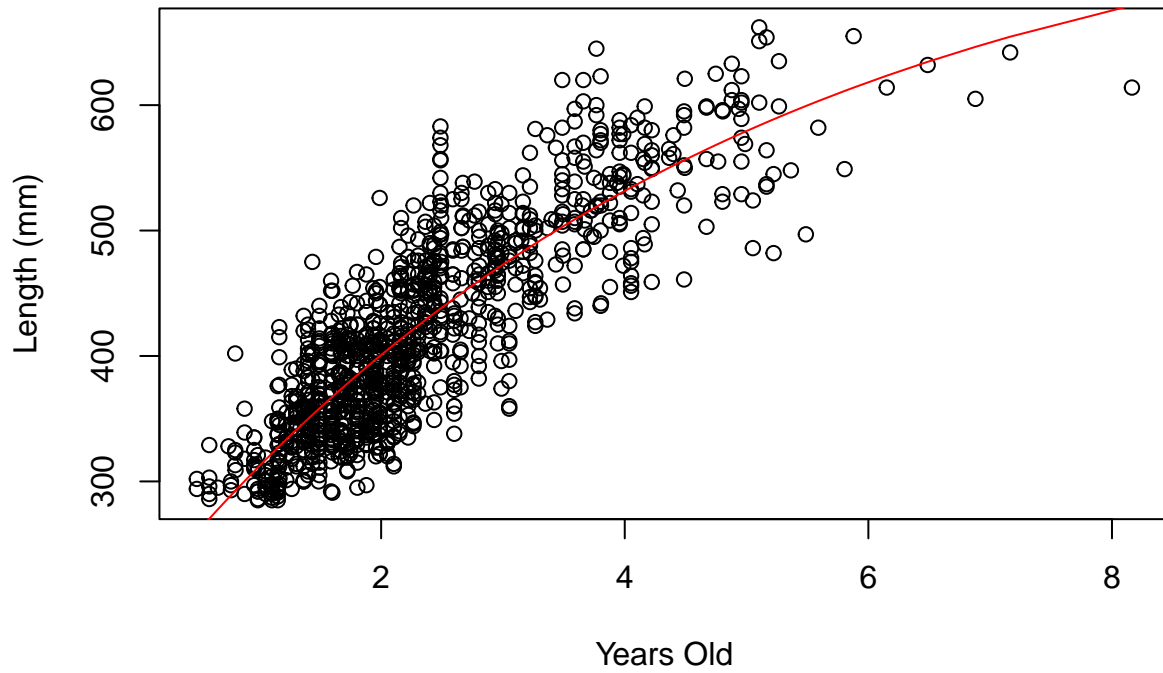
```
}  
)
```

**Indian River Residuals Plot**



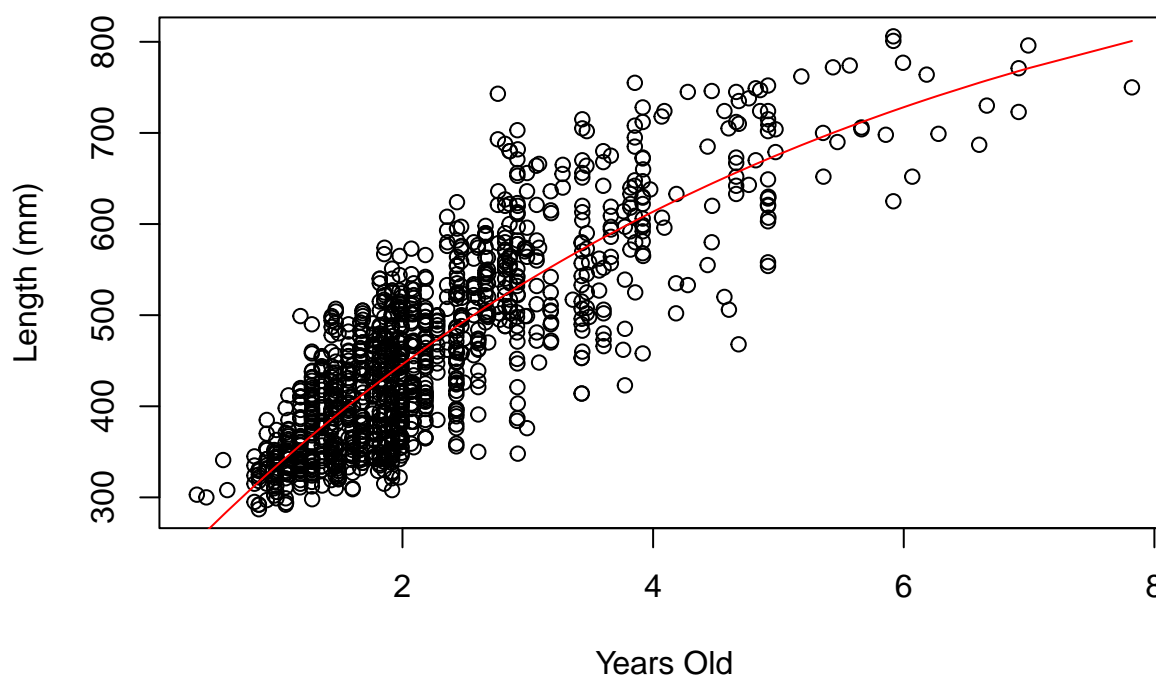
```
with(  
  ch_data, {  
    plot(  
      tl ~ yearsold,  
      main='Charlotte Harbor Fit',  
      xlab='Years Old',  
      ylab='Length (mm)'  
    )  
    lines(nls_tl ~ yearsold, col='red')  
  }  
)
```

## Charlotte Harbor Fit



```
with(  
  ir_data, {  
    plot(  
      tl ~ yearsold,  
      main='Indian River Fit',  
      xlab='Years Old',  
      ylab='Length (mm)'  
    )  
    lines(nls_tl ~ yearsold, col='red')  
  }  
)
```

## Indian River Fit



Does it appear that a von Bertalanffy curve is reasonable for each population?

The conclusion here is the same as in the last lab. The fit is reasonable but far from ideal. There is a clear pattern from under predicting to overpredicting as length increases so there's definitely a pattern to the residuals that (in theory) could be captured in a better model.

There's also quite a lot of variance left in these estimates, and that variance itself is patterned with our gear selectivity creating variance toward smaller predicted lengths.

**2. For the Charlotte Harbor population, create a function that estimates the negative log likelihood of the normal distribution. Minimize the negative log likelihood of the normal distribution to estimate the parameters of the von Bertalanffy growth equation.**

```
predict_length <- function(yearsold, Linf, vbk, tknot) {
  pred_tl <- Linf * (1 - exp(-vbk * (yearsold - tknot)))
  return(pred_tl)
}

get_likelihood <- function(yearsold, tl, Linf, vbk, tknot, sig) {
  pred_tl <- predict_length(yearsold, Linf, vbk, tknot)
  NLL <- -1 * sum(dnorm(tl, pred_tl, sig, log=T), na.rm=T)
  return(NLL)
}

do_likelihood_fit <- function(fish_data, Linf, vbk, tknot, sig) {
```

```

t1 <- as.numeric(fish_data$t1)
yearsold <- as.numeric(fish_data$yearsold)
lLinf <- log(Linf)
lsig <- log(sig)

objective <- function(v) {
  Linf <- exp(v[1])
  vbk <- v[2]
  tknot <- v[3]
  sig <- exp(v[4])
  NLL <- get_likelihoood(yearsold, t1, Linf, vbk, tknot, sig)
  return(NLL)
}

v <- c(lLinf, vbk, tknot, lsig)
fit <- optim(v, objective, hessian=T)
print("Fit Summary")
print(fit)

covm <- solve(fit$hessian)

pred <- c(
  exp(fit$par[1]),
  fit$par[2],
  fit$par[3],
  exp(fit$par[4])
)
print(pred)
return(list(pred, covm, fit$par))
}

ch_ml_result <- do_likelihoood_fit(ch_data, 800, 0.2, -1.4, 40)

## [1] "Fit Summary"
## $par
## [1] 6.6703456 0.2055542 -1.4520493 3.7381685
##
## $value
## [1] 5683.029
##
## $counts
## function gradient
##      157      NA
##
## $convergence
## [1] 0
##
## $message
## NULL
##
## $hessian
##           [,1]      [,2]      [,3]      [,4]
## [1,] 112145.777336 348609.06203 -19240.314501 7.352657

```

```
## [2,] 348609.062030 1102530.32351 -62593.953455 25.512458
## [3,] -19240.314501 -62593.95346 3753.243085 -1.014529
## [4,] 7.352657 25.51246 -1.014529 2203.590885
##
## [1] 788.6681316 0.2055542 -1.4520493 42.0209562

ch_pred_params <- ch_ml_result[[1]]
ch_covm <- ch_ml_result[[2]]
ch_ml_par <- ch_ml_result[[3]]

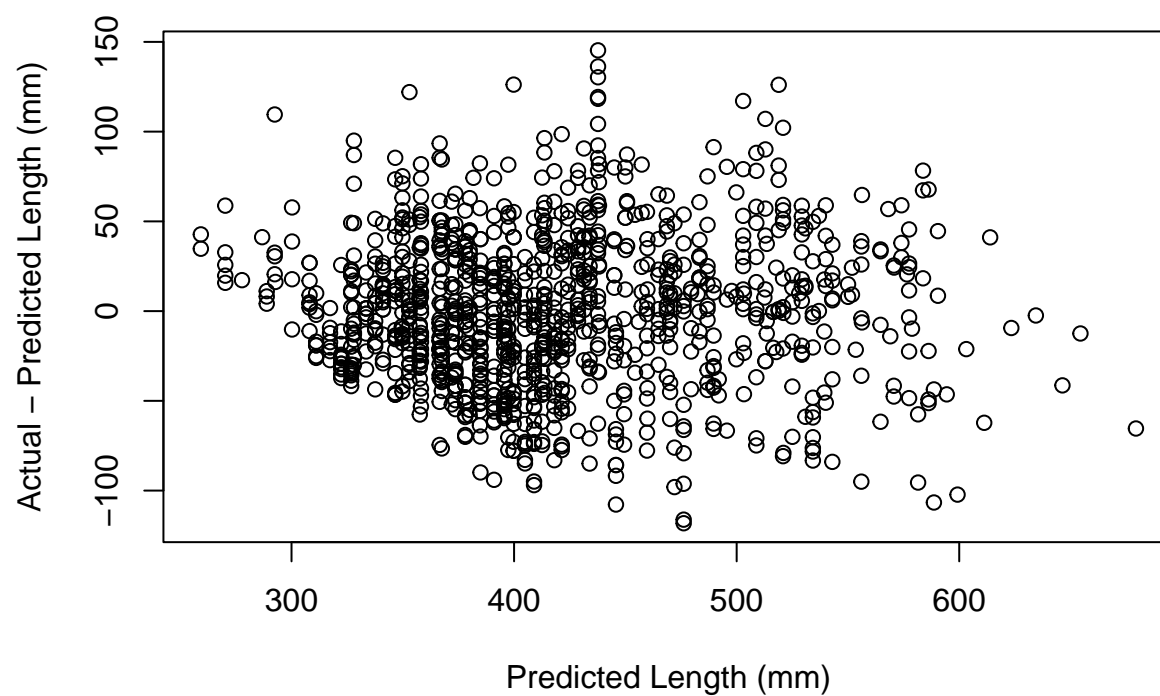
ch_data$ml_tl <- predict_length(
  ch_data$yearsold, ch_pred_params[1], ch_pred_params[2], ch_pred_params[3]
)
ch_data$ml_res <- ch_data$t1 - ch_data$ml_tl
```

Plot the residuals against the predicted values from the likelihood estimation

```
with(
  ch_data, {
    plot(
      ml_res ~ ml_tl,
      main='Charlotte Harbor Residuals Plot',
      xlab='Predicted Length (mm)',
      ylab='Actual - Predicted Length (mm)'
    )
  }
)
```

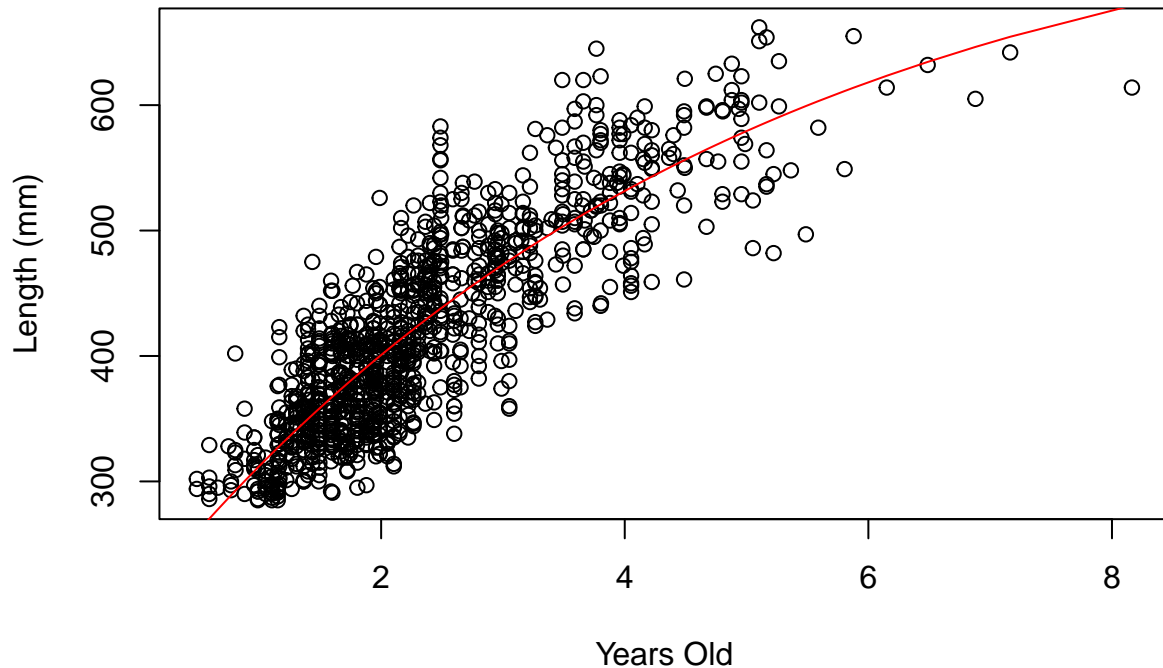


## Charlotte Harbor Residuals Plot



```
with(  
  ch_data, {  
    plot(  
      tl ~ yearsold,  
      main='Charlotte Harbor Fit',  
      xlab='Years Old',  
      ylab='Length (mm)'  
    )  
    lines(ml_t1 ~ yearsold, col='red')  
  }  
)
```

## Charlotte Harbor Fit



Does the likelihood estimation appear to return VBGM parameter estimates that are a reasonable model for the data?

$t_0$  is clearly off and the model seems to be overpredicting  $L_\infty$ . Furthermore there is clearly changing variance with both age and length so the  $\sigma$  parameter is also contentious. Overall the fit to the data given is reasonable.

Compare your estimates to the between the nls (question 1) and the likelihood estimation.

Besides having added  $\sigma$  this is the same fit. So the estimates are the same.

**3. Generate the confidence intervals for each parameter from the model using the output from nls and the Hessian matrix from Optim.**

Report the confidence intervals for each parameter from each method in a table

```
ml_stderr <- sqrt(diag(ch_covm))

ALPHA <- 0.05
ML_L95 <- (ch_ml_par - qnorm(1-(ALPHA/2)) * ml_stderr)
ML_U95 <- (ch_ml_par + qnorm(1-(ALPHA/2)) * ml_stderr)
ML_U95[1] <- exp(ML_U95[1])
ML_L95[1] <- exp(ML_L95[1])
ML_U95[4] <- exp(ML_U95[4])
ML_L95[4] <- exp(ML_L95[4])
```

```
ch_nls_par <- summary(ch_result)$parameters[,1]
ch_nls_stderr <- summary(ch_result)$parameters[,2]
NLS_L95 <- (ch_nls_par - qnorm(1-(ALPHA/2)) * ch_nls_stderr)
NLS_U95 <- (ch_nls_par + qnorm(1-(ALPHA/2)) * ch_nls_stderr)
```

```
NLS_U95['sig'] <- NaN
NLS_L95['sig'] <- NaN
rbind(ML_U95, ML_L95, NLS_U95, NLS_L95)
```

```
##           Linf           vbk           tknot           sig
## ML_U95  870.2427 0.2528076 -1.146851 43.81258
## ML_L95  714.7402 0.1583008 -1.757248 40.30259
## NLS_U95 875.1955 0.2580933 -1.115282      NaN
## NLS_L95 702.5762 0.1528279 -1.789379      NaN
```

Compare the estimated confidence intervals between methods.

$\sigma$  obviously offers no comparison as it is not in both methods.  $t_0$  and  $k$  are both very similar between the two methods.  $L_\infty$  is the only one that stands out but it was fit as the log of itself during maximum likelihood estimation and so it's expected that the confidence interval for  $L_\infty$  is asymmetric in the likelihood estimation.

#### 4. Conduct the same analysis for the Indian River population (questions 1-3). Answer the same questions/bullet points.

```
ir_ml_result <- do_likelihood_fit(ir_data, 950, 0.19, -1.2, 40)
```

```
## [1] "Fit Summary"
## $par
## [1] 6.8842492 0.1896156 -1.2168596 4.0649088
##
## $value
## [1] 6553.262
##
## $counts
## function gradient
##      177      NA
##
## $convergence
## [1] 0
##
## $message
## NULL
##
## $hessian
##           [,1]           [,2]           [,3]           [,4]
## [1,] 76708.491347 276090.71644 -15324.743368 2.288774
## [2,] 276090.716445 1009606.09770 -57676.210383 11.120881
## [3,] -15324.743368 -57676.21038 3507.467652 -0.400318
## [4,] 2.288774 11.12088 -0.400318 2390.262904
##
## [1] 976.7680293 0.1896156 -1.2168596 58.2595954
ir_pred_params <- ir_ml_result[[1]]
ir_covm <- ir_ml_result[[2]]
```

```

ir_ml_par <- ir_ml_result[[3]]

ir_data$ml_t1 <- predict_length(
  ir_data$yearsold, ir_pred_params[1], ir_pred_params[2], ir_pred_params[3]
)
ir_data$ml_res <- ir_data$t1 - ir_data$ml_t1

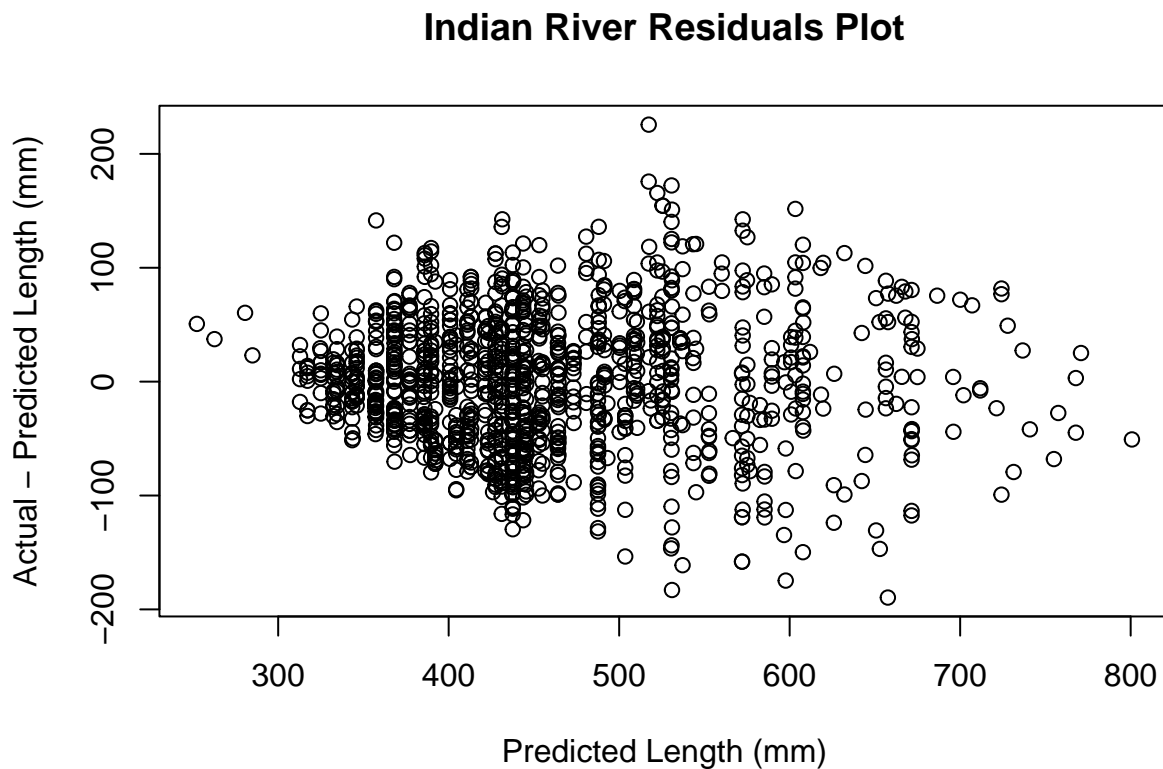
```

Plot the residuals against the predicted values from the likelihood estimation

```

with(
  ir_data, {
    plot(
      ml_res ~ ml_t1,
      main='Indian River Residuals Plot',
      xlab='Predicted Length (mm)',
      ylab='Actual - Predicted Length (mm)'
    )
  }
)

```



```

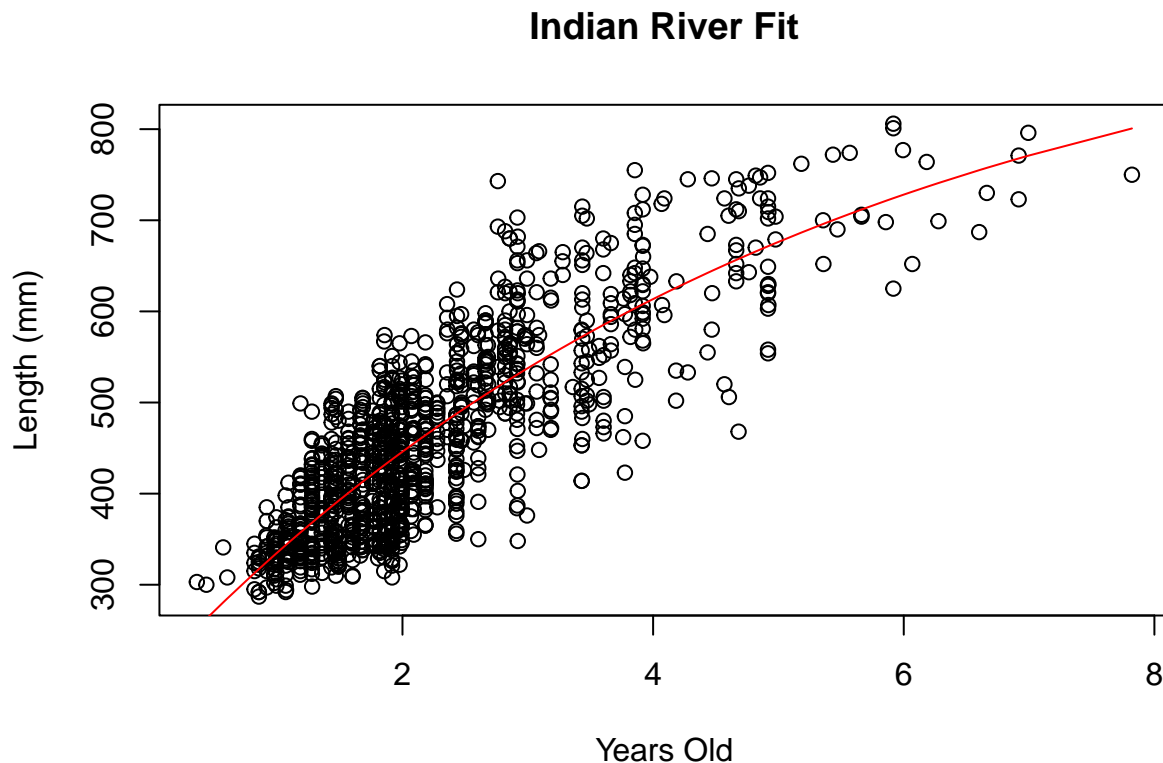
with(
  ir_data, {
    plot(
      t1 ~ yearsold,
      main='Indian River Fit',
      xlab='Years Old',

```

```

    ylab='Length (mm)'
  )
  lines(ml_t1 ~ yearsold, col='red')
}
)

```



Does the likelihood estimation appear to return VBGM parameter estimates that are a reasonable model for the data?

$t_0$  is clearly off again. There is still clearly changing variance with both age and length so the  $\sigma$  parameter is also contentious. Overall the fit to the data given is reasonable.

Compare your estimates to the between the nls (question 1) and the likelihood estimation.

Besides having added  $\sigma$  this is the same fit. So the estimates are the same.

Report the confidence intervals for each parameter from each method in a table

```

ml_stderr <- sqrt(diag(ir_covm))

ALPHA <- 0.05
ML_L95 <- (ir_ml_par - qnorm(1-(ALPHA/2)) * ml_stderr)
ML_U95 <- (ir_ml_par + qnorm(1-(ALPHA/2)) * ml_stderr)
ML_U95[1] <- exp(ML_U95[1])
ML_L95[1] <- exp(ML_L95[1])

```

```

ML_U95[4] <- exp(ML_U95[4])
ML_L95[4] <- exp(ML_L95[4])

ir_nls_par <- summary(ir_result)$parameters[,1]
ir_nls_stderr <- summary(ir_result)$parameters[,2]
NLS_L95 <- (ir_nls_par - qnorm(1-(ALPHA/2)) * ir_nls_stderr)
NLS_U95 <- (ir_nls_par + qnorm(1-(ALPHA/2)) * ir_nls_stderr)

NLS_U95['sig'] <- NaN
NLS_L95['sig'] <- NaN
rbind(ML_U95, ML_L95, NLS_U95, NLS_L95)

```

```

##           Linf          vbk          tknot          sig
## ML_U95  1101.7064 0.2376684 -0.9299628 60.64261
## ML_L95   865.9982 0.1415629 -1.5037564 55.97022
## NLS_U95 1102.4265 0.2405846 -0.9139983      NaN
## NLS_L95  852.0484 0.1382837 -1.5216089      NaN

```

**Compare the estimated confidence intervals between methods.**

It's interesting that here all parameters are extremely close in terms of bounds - even  $L_\infty$ . However the confidence interval for  $L_\infty$  is still asymmetric for the maximum likelihood estimation.

**5. Using the likelihood-based model parameter estimates for each population, calculate the mean length at each age for age 0 to age 8 for each population.**

**Report the mean length at age for each population.**

```

age <- seq(0, 8, 1)
ch_t1 <- predict_length(
  age, ch_pred_params[1], ch_pred_params[2], ch_pred_params[3]
)
ir_t1 <- predict_length(
  age, ir_pred_params[1], ir_pred_params[2], ir_pred_params[3]
)
cbind(age, ch_t1, ir_t1)

```

```

##      age   ch_t1   ir_t1
## [1,]  0 203.5166 201.2637
## [2,]  1 312.2402 335.2111
## [3,]  2 400.7624 446.0227
## [4,]  3 472.8369 537.6946
## [5,]  4 531.5196 613.5326
## [6,]  5 579.2988 676.2717
## [7,]  6 618.2005 728.1743
## [8,]  7 649.8741 771.1122
## [9,]  8 675.6626 806.6337

```

**Compare the length at age between the two populations.**

Clearly Indian River is growing faster and larger than the Charlotte Harbor. It's also pretty unbiologically sound that at age 0 they are 200mm already given they should be more like 6mm.

## 6. Set up an AIC table:

```
map_columns <- function(v, cols) {
  c <- rep(0, 8)
  i <- 0
  for (col in cols) {
    i <- i + 1
    if (endsWith(col, 'Linf')) {
      j <- 1
    } else if (endsWith(col, 'vbk')) {
      j <- 3
    } else if (endsWith(col, 'tknot')) {
      j <- 5
    } else {
      j <- 7
    }

    if (startsWith(col, 'ch_')) {
      c[j] <- v[i]
    } else if (startsWith(col, 'ir_')) {
      c[j+1] <- v[i]
    } else {
      c[j] <- v[i]
      c[j+1] <- v[i]
    }
  }
  return(c)
}

map_columns(c(1, 3, 2, 3, 4), c('ir_Linf', 'ch_Linf', 'vbk', 'tknot', 'sig'))

## [1] 3 1 2 2 3 3 4 4

do_likelihood_fit <- function(v, cols, runs) {
  objective <- function(v) {
    c <- map_columns(v, cols)

    ch_Linf <- exp(c[1])
    ch_vbk <- c[3]
    ch_tknot <- c[5]
    ch_sig <- exp(c[7])

    ir_Linf <- exp(c[2])
    ir_vbk <- c[4]
    ir_tknot <- c[6]
    ir_sig <- exp(c[8])

    ch_NLL <- get_likelihood(ch_data$yearsold, ch_data$t1, ch_Linf, ch_vbk, ch_tknot, ch_sig)
    ir_NLL <- get_likelihood(ir_data$yearsold, ir_data$t1, ir_Linf, ir_vbk, ir_tknot, ir_sig)

    NLL <- ch_NLL + ir_NLL
    return(NLL)
  }
}
```

```

for (i in 1:runs) {
  fit <- optim(v, objective, hessian=T)
  v <- fit$par
}
return(fit)
}

summarize_aic <- function(fit, cols) {
  dof <- length(cols)
  nll <- fit$value
  aic <- 2*nll + 2*dof
  row <- c(paste(cols, collapse=','), dof, nll, aic)
  return(row)
}

```

Determine the log likelihood, the number of parameters, and the AIC for:

```

v <- c(
  6.7, 0.2, -1.5, 3.7,
  6.9, 0.2, -1.2, 4.1
)
cols <- c(
  'ch_Linf', 'ch_vbk', 'ch_tknot', 'ch_sig',
  'ir_Linf', 'ir_vbk', 'ir_tknot', 'ir_sig'
)
(fit <- do_likelihood_fit(v, cols, 25))

```

A model with different parameters ( $L_\infty$ ,  $t_0$ ,  $k$ , and  $\sigma$ ) for each population.

```

## $par
## [1] 6.6698015 0.2058859 -1.4495751 3.7381019 6.8934140 0.1860532 -1.2361554
## [8] 4.0651624
##
## $value
## [1] 12236.3
##
## $counts
## function gradient
##      215      NA
##
## $convergence
## [1] 0
##
## $message
## NULL
##
## $hessian
##           [,1]      [,2]      [,3]      [,4]      [,5]
## [1,] 112181.383961 347993.25653 -1.925144e+04 -6.426805e+00 0.000000
## [2,] 347993.256531 1098196.84390 -6.250397e+04 -1.593776e+01 0.000000
## [3,] -19251.438278 -62503.97185 3.756849e+03 9.573509e-01 0.000000
## [4,] -6.426805 -15.93776 9.573509e-01 2.203885e+03 0.000000
## [5,] 0.000000 0.00000 0.000000e+00 0.000000e+00 76678.357183

```



```
## [6,]      0.000000      0.00000 2.273737e-07 -2.273737e-07 282775.217390
## [7,]      0.000000      0.00000 0.000000e+00 0.000000e+00 -15298.653825
## [8,]      0.000000      0.00000 0.000000e+00 0.000000e+00      -3.567621
##           [,6]           [,7]           [,8]
## [1,] 0.000000e+00 0.000000e+00 0.00000000
## [2,] 0.000000e+00 0.000000e+00 0.00000000
## [3,] 2.273737e-07 0.000000e+00 0.00000000
## [4,] -2.273737e-07 0.000000e+00 0.00000000
## [5,] 2.827752e+05 -1.529865e+04 -3.5676208
## [6,] 1.058861e+06 -5.893007e+04 6.2779386
## [7,] -5.893007e+04 3.489742e+03 -0.1100952
## [8,] 6.277939e+00 -1.100952e-01 2389.0876109
```

```
(all_free <- summarize_aic(fit, cols))
```

```
## [1] "ch_Linf,ch_vbk,ch_tknot,ch_sig,ir_Linf,ir_vbk,ir_tknot,ir_sig"
## [2] "8"
## [3] "12236.300720374"
## [4] "24488.6014407481"
```

```
v <- c(
  6.7, 0.2, -1.5, 3.7
)
cols <- c(
  'Linf', 'vbk', 'tknot', 'sig'
)
(fit <- do_likelihood_fit(v, cols, 25))
```

**A model with shared parameters for both populations.**

```
## $par
## [1] 6.8171686 0.1797327 -1.4666141 4.0528655
##
## $value
## [1] 12569.06
##
## $counts
## function gradient
##      99      NA
##
## $convergence
## [1] 0
##
## $message
## NULL
##
## $hessian
##           [,1]           [,2]           [,3]           [,4]
## [1,] 137876.9646 520338.81924 -25168.801018 -7.391700
## [2,] 520338.8192 1990716.25051 -98792.191938 -18.505014
## [3,] -25168.8010 -98792.19194 5182.480004 1.192387
## [4,] -7.3917 -18.50501 1.192387 4595.299049
```

```
(all_shared <- summarize_aic(fit, cols))
```

```
## [1] "Linf,vbk,tknot,sig" "4" "12569.0570197932"
## [4] "25146.1140395863"
```

Which hypothesis had the most support based on AIC?

As in the last lab, all parameters shared has the most support.

## 7. Continue adding to your AIC table.

Determine the log likelihood function, the log likelihood, the number of parameters, and the AIC for

```
v <- c(
  6.7, 0.2, -1.5, 3.7,
  0.2, -1.2, 4.1
)
cols <- c(
  'Linf', 'ch_vbk', 'ch_tknot', 'ch_sig',
  'ir_vbk', 'ir_tknot', 'ir_sig'
)
(fit <- do_likelihood_fit(v, cols, 25))
```

Models that share only one of the VBGM parameters ( $L_\infty$ ,  $t_0$ ,  $k$ , and  $\sigma$ ) for each population (hint: this should result in four additional models for each population).

```
## $par
## [1] 6.8068761 0.1532091 -1.8059854 3.7401227 0.2239987 -1.0478161 4.0655605
##
## $value
## [1] 12239.52
##
## $counts
## function gradient
##      192      NA
##
## $convergence
## [1] 0
##
## $message
## NULL
##
## $hessian
##           [,1]      [,2]      [,3]      [,4]      [,5]
## [1,] 188324.30850 5.088257e+05 -1.887554e+04 -5.441198e+01 2.218592e+05
## [2,] 508825.67475 2.337803e+06 -8.853964e+04 8.041794e+00 -9.094947e-07
## [3,] -18875.53594 -8.853964e+04 3.518782e+03 3.565174e-01 0.000000e+00
## [4,] -54.41198 8.041794e+00 3.565174e-01 2.204096e+03 -9.094947e-07
## [5,] 221859.22437 -9.094947e-07 0.000000e+00 -9.094947e-07 6.574851e+05
## [6,] -15462.87936 0.000000e+00 0.000000e+00 0.000000e+00 -4.738922e+04
## [7,] 54.78169 0.000000e+00 0.000000e+00 0.000000e+00 2.943767e+00
##           [,6]      [,7]
## [1,] -1.546288e+04 5.478169e+01
## [2,] 0.000000e+00 0.000000e+00
## [3,] 0.000000e+00 0.000000e+00
## [4,] 0.000000e+00 0.000000e+00
```

```

## [5,] -4.738922e+04  2.943767e+00
## [6,]  3.650500e+03 -2.950856e-03
## [7,] -2.950856e-03  2.390953e+03

(Linf_shared <- summarize_aic(fit, cols))

## [1] "Linf,ch_vbk,ch_tknot,ch_sig,ir_vbk,ir_tknot,ir_sig"
## [2] "7"
## [3] "12239.5224904294"
## [4] "24493.0449808588"

v <- c(
  6.7, 0.2, -1.5, 3.7,
  6.9, -1.2, 4.1
)
cols <- c(
  'ch_Linf', 'vbk', 'ch_tknot', 'ch_sig',
  'ir_Linf', 'ir_tknot', 'ir_sig'
)
(fit <- do_likelihood_fit(v, cols, 25))

## $par
## [1] 6.6866966 0.1977578 -1.5017712 3.7381292 6.8648702 -1.1711513 4.0649778
##
## $value
## [1] 12236.39
##
## $counts
## function gradient
##      180      NA
##
## $convergence
## [1] 0
##
## $message
## NULL
##
## $hessian
##           [,1]      [,2]      [,3]      [,4]      [,5]
## [1,] 1.121581e+05 366804.57392 -1.918373e+04  4.9503260 -4.547474e-07
## [2,] 3.668046e+05 2126473.25099 -6.550165e+04  39.7806680 2.615699e+05
## [3,] -1.918373e+04 -65501.65115  3.713248e+03  0.1034027 4.547474e-07
## [4,] 4.950326e+00  39.78067  1.034027e-01 2203.9542305 0.000000e+00
## [5,] -4.547474e-07 261569.89134 4.547474e-07 0.0000000 7.669719e+04
## [6,] -2.273737e-07 -54995.95528 2.273737e-07 0.0000000 -1.538221e+04
## [7,] 0.000000e+00 -10.83637 0.000000e+00 0.0000000 2.695255e+00
##           [,6]      [,7]
## [1,] -2.273737e-07 0.000000
## [2,] -5.499596e+04 -10.836374
## [3,] 2.273737e-07 0.000000
## [4,] 0.000000e+00 0.000000
## [5,] -1.538221e+04 2.695255
## [6,] 3.551922e+03 -1.031214
## [7,] -1.031214e+00 2390.151049

```

```

(vbk_shared <- summarize_aic(fit, cols))

## [1] "ch_Linf,vbk,ch_tknot,ch_sig,ir_Linf,ir_tknot,ir_sig"
## [2] "7"
## [3] "12236.3934261983"
## [4] "24486.7868523965"

v <- c(
  6.7, 0.2, -1.5, 3.7,
  6.9, 0.2, 4.1
)
cols <- c(
  'ch_Linf', 'ch_vbk', 'tknot', 'ch_sig',
  'ir_Linf', 'ir_vbk', 'ir_sig'
)
(fit <- do_likelihood_fit(v, cols, 25))

## $par
## [1] 6.6411004 0.2216115 -1.3462903 3.7384119 6.9343917 0.1699523 4.0654174
##
## $value
## [1] 12236.86
##
## $counts
## function gradient
##      194      NA
##
## $convergence
## [1] 0
##
## $message
## NULL
##
## $hessian
##           [,1]      [,2]      [,3]      [,4]      [,5]
## [1,] 1.121026e+05 315760.971000 -19409.8411 -0.5502118 0.000000e+00
## [2,] 3.157610e+05 907209.514896 -57515.5350 9.6947774 0.000000e+00
## [3,] -1.940984e+04 -57515.534999 7230.1371 -10.1015044 -1.510730e+04
## [4,] -5.502118e-01 9.694777 -10.1015 2203.4393414 0.000000e+00
## [5,] 0.000000e+00 0.000000 -15107.3000 0.0000000 7.663296e+04
## [6,] 0.000000e+00 0.000000 -64885.4245 0.0000000 3.167108e+05
## [7,] 0.000000e+00 0.000000 10.1164 0.0000000 1.951494e-01
##           [,6]      [,7]
## [1,] 0.000000e+00 0.0000000
## [2,] 0.000000e+00 0.0000000
## [3,] -6.488542e+04 10.1163982
## [4,] 0.000000e+00 0.0000000
## [5,] 3.167108e+05 0.1951494
## [6,] 1.325922e+06 5.3842418
## [7,] 5.384242e+00 2389.1819192

(tknot_shared <- summarize_aic(fit, cols))

## [1] "ch_Linf,ch_vbk,tknot,ch_sig,ir_Linf,ir_vbk,ir_sig"
## [2] "7"

```

```

## [3] "12236.8592624778"
## [4] "24487.7185249557"

v <- c(
  6.7, 0.2, -1.5, 3.7,
  6.9, 0.2, -1.2
)
cols <- c(
  'ch_Linf', 'ch_vbk', 'ch_tknot', 'sig',
  'ir_Linf', 'ir_vbk', 'ir_tknot'
)
(fit <- do_likelihood_fit(v, cols, 25))

## $par
## [1] 6.6676369 0.2069044 -1.4436927 3.9341872 6.8839452 0.1897212 -1.2165016
##
## $value
## [1] 12295.98
##
## $counts
## function gradient
##      182      NA
##
## $convergence
## [1] 0
##
## $message
## NULL
##
## $hessian
##           [,1]      [,2]      [,3]      [,4]      [,5]
## [1,] 7.578281e+04 2.335499e+05 -1.300820e+04 -0.55997748 0.000000e+00
## [2,] 2.335499e+05 7.323867e+05 -4.197229e+04 -2.61466539 -4.547474e-07
## [3,] -1.300820e+04 -4.197229e+04 2.540665e+03 0.22086579 4.547474e-07
## [4,] -5.599775e-01 -2.614665e+00 2.208658e-01 4593.40523503 2.110082e+00
## [5,] 0.000000e+00 -4.547474e-07 4.547474e-07 2.11008182 9.963053e+04
## [6,] -2.273737e-07 4.547474e-07 -2.273737e-07 9.33348656 3.583302e+05
## [7,] -4.547474e-07 0.000000e+00 4.547474e-07 -0.07147719 -1.990348e+04
##           [,6]      [,7]
## [1,] -2.273737e-07 -4.547474e-07
## [2,] 4.547474e-07 0.000000e+00
## [3,] -2.273737e-07 4.547474e-07
## [4,] 9.333487e+00 -7.147719e-02
## [5,] 3.583302e+05 -1.990348e+04
## [6,] 1.309389e+06 -7.485600e+04
## [7,] -7.485600e+04 4.555389e+03

(sig_shared <- summarize_aic(fit, cols))

## [1] "ch_Linf,ch_vbk,ch_tknot,sig,ir_Linf,ir_vbk,ir_tknot"
## [2] "7"
## [3] "12295.979579067"
## [4] "24605.959158134"

```

8. Calculate the  $\Delta AIC$  for all the models you trialed (report your AIC table to answer questions 6, 7, and 8).

```
aic_table <- data.frame(rbind(
  all_free,
  all_shared,
  Linf_shared,
  vbk_shared,
  tknot_shared,
  sig_shared
))
colnames(aic_table) <- c('cols', 'dof', 'nll', 'aic')

aic_table$aic <- as.numeric(aic_table$aic)
aic_table$delta <- aic_table$aic - min(aic_table$aic)
(aic_table <- aic_table[order(aic_table$delta),])
```

```
##                                cols dof
## vbk_shared          ch_Linf,vbk,ch_tknot,ch_sig,ir_Linf,ir_tknot,ir_sig  7
## tknot_shared        ch_Linf,ch_vbk,tknot,ch_sig,ir_Linf,ir_vbk,ir_sig  7
## all_free      ch_Linf,ch_vbk,ch_tknot,ch_sig,ir_Linf,ir_vbk,ir_tknot,ir_sig  8
## Linf_shared        Linf,ch_vbk,ch_tknot,ch_sig,ir_vbk,ir_tknot,ir_sig  7
## sig_shared        ch_Linf,ch_vbk,ch_tknot,sig,ir_Linf,ir_vbk,ir_tknot  7
## all_shared                                Linf,vbk,tknot,sig  4
##                                nll      aic      delta
## vbk_shared    12236.3934261983 24486.79   0.0000000
## tknot_shared  12236.8592624778 24487.72   0.9316726
## all_free      12236.300720374 24488.60   1.8145884
## Linf_shared   12239.5224904294 24493.04   6.2581285
## sig_shared    12295.979579067 24605.96 119.1723057
## all_shared    12569.0570197932 25146.11 659.3271872
```

Which hypothesis had the most support based on AIC?

Sharing  $k$  had the best support.

What are the top models based on the AIC?

Sharing  $k$ , sharing  $t_0$ , sharing  $L_\infty$ , and all free were the best.

Do these top models make sense given the biology of Spotted Seatrout?

Sharing  $t_0$  makes sense but the  $t_0$  here is certainly unbiological. Growth rates would definitely change from place to place, but I think we're still able to fit reasonably well because  $t_0$  and  $L_\infty$  are still free in the shared  $k$  case and we know these are all extremely correlated with one another. Same argument for  $L_\infty$ , I'd expect it to be different but the model is likely taking advantage of the other parameters to still fit. In general across these models the wild  $t_0$  makes these rather unbiological.

9. Adjust the slot limits for the following scenarios and answer the following questions:

MinLL	MaxLL	Bias	Over	Under	Parameters
300 (0.05)	$\infty$	< 2 years old undersampled	< 4	N/A	$L_{\infty} = 914, k = 0.178, t_0 = -1.09, \sigma = 110.0$
0	700 (0.05)	> 3 years old undersampled	$0 < age < 3$	> 3	$L_{\infty} = 667, k = 0.335, t_0 = -0.39, \sigma = 66.2$
380 (0)	480 (0)	Almost no sampling for age > 6 or < 2, extremely poor sampling across the board.	< 3	> 3	$L_{\infty} = 446, k = 0.939, t_0 = -1.15, \sigma = 26.4$
380 (0)	530 (0.03)	No sampling for age < 2 biased toward middling or smaller fish throughout.	< 3	> 3	$L_{\infty} = 532, k = 0.222, t_0 = -5.56, \sigma = 40.2$

**Across scenarios, which had the biggest impact on  $L_{\infty}$ ? on  $k$ ? on  $t_0$ ?**

- $L_{\infty}$  - Definitely case 3, the total lack of sampling of large fish threw things off entirely.
- $k$  - Likewise case 3.
- $t_0$  - case 4, the complete lack of samples on the low end with samples on the high end gave us a weird almost straight relationship between age and growth.