Data analysis project

Jess Qualley and Rebecca Hansen

March 27, 2020

##Clear code (**delete before submitting**\*)

rm(list = ls(all = TRUE))

#Preparation ##Loading and citing packages

#Load faraway package  
library(faraway)  
  
#Load plyr package  
library(plyr)

##   
## Attaching package: 'plyr'

## The following object is masked from 'package:faraway':  
##   
## ozone

#Load tidyverse package  
library(tidyverse)

## ── Attaching packages ────────────────────────────────────────────────────────────────────────────────────────────────────── tidyverse 1.3.0 ──

## ✔ ggplot2 3.2.1 ✔ purrr 0.3.3  
## ✔ tibble 2.1.3 ✔ dplyr 0.8.3  
## ✔ tidyr 1.0.2 ✔ stringr 1.4.0  
## ✔ readr 1.3.1 ✔ forcats 0.4.0

## ── Conflicts ───────────────────────────────────────────────────────────────────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::arrange() masks plyr::arrange()  
## ✖ purrr::compact() masks plyr::compact()  
## ✖ dplyr::count() masks plyr::count()  
## ✖ dplyr::failwith() masks plyr::failwith()  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::id() masks plyr::id()  
## ✖ dplyr::lag() masks stats::lag()  
## ✖ dplyr::mutate() masks plyr::mutate()  
## ✖ dplyr::rename() masks plyr::rename()  
## ✖ dplyr::summarise() masks plyr::summarise()  
## ✖ dplyr::summarize() masks plyr::summarize()

#Load gridextra package  
library(gridExtra)

##   
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':  
##   
## combine

#Load lmer package  
library(lme4)

## Loading required package: Matrix

##   
## Attaching package: 'Matrix'

## The following objects are masked from 'package:tidyr':  
##   
## expand, pack, unpack

#Load bbmle package  
library(bbmle)

## Loading required package: stats4

##   
## Attaching package: 'bbmle'

## The following object is masked from 'package:dplyr':  
##   
## slice

#Cite packages  
citation("faraway")

##   
## To cite package 'faraway' in publications use:  
##   
## Julian Faraway (2016). faraway: Functions and Datasets for Books  
## by Julian Faraway. R package version 1.0.7.  
## https://CRAN.R-project.org/package=faraway  
##   
## A BibTeX entry for LaTeX users is  
##   
## @Manual{,  
## title = {faraway: Functions and Datasets for Books by Julian Faraway},  
## author = {Julian Faraway},  
## year = {2016},  
## note = {R package version 1.0.7},  
## url = {https://CRAN.R-project.org/package=faraway},  
## }  
##   
## ATTENTION: This citation information has been auto-generated from  
## the package DESCRIPTION file and may need manual editing, see  
## 'help("citation")'.

citation("plyr")

##   
## To cite plyr in publications use:  
##   
## Hadley Wickham (2011). The Split-Apply-Combine Strategy for Data  
## Analysis. Journal of Statistical Software, 40(1), 1-29. URL  
## http://www.jstatsoft.org/v40/i01/.  
##   
## A BibTeX entry for LaTeX users is  
##   
## @Article{,  
## title = {The Split-Apply-Combine Strategy for Data Analysis},  
## author = {Hadley Wickham},  
## journal = {Journal of Statistical Software},  
## year = {2011},  
## volume = {40},  
## number = {1},  
## pages = {1--29},  
## url = {http://www.jstatsoft.org/v40/i01/},  
## }

citation("tidyverse")

##   
## Wickham et al., (2019). Welcome to the tidyverse. Journal of  
## Open Source Software, 4(43), 1686,  
## https://doi.org/10.21105/joss.01686  
##   
## A BibTeX entry for LaTeX users is  
##   
## @Article{,  
## title = {Welcome to the {tidyverse}},  
## author = {Hadley Wickham and Mara Averick and Jennifer Bryan and Winston Chang and Lucy D'Agostino McGowan and Romain François and Garrett Grolemund and Alex Hayes and Lionel Henry and Jim Hester and Max Kuhn and Thomas Lin Pedersen and Evan Miller and Stephan Milton Bache and Kirill Müller and Jeroen Ooms and David Robinson and Dana Paige Seidel and Vitalie Spinu and Kohske Takahashi and Davis Vaughan and Claus Wilke and Kara Woo and Hiroaki Yutani},  
## year = {2019},  
## journal = {Journal of Open Source Software},  
## volume = {4},  
## number = {43},  
## pages = {1686},  
## doi = {10.21105/joss.01686},  
## }

citation("bbmle")

##   
## To cite package 'bbmle' in publications use:  
##   
## Ben Bolker and R Development Core Team (2020). bbmle: Tools for  
## General Maximum Likelihood Estimation. R package version  
## 1.0.23.1. https://CRAN.R-project.org/package=bbmle  
##   
## A BibTeX entry for LaTeX users is  
##   
## @Manual{,  
## title = {bbmle: Tools for General Maximum Likelihood Estimation},  
## author = {Ben Bolker and {R Development Core Team}},  
## year = {2020},  
## note = {R package version 1.0.23.1},  
## url = {https://CRAN.R-project.org/package=bbmle},  
## }

citation("gridExtra")

##   
## To cite package 'gridExtra' in publications use:  
##   
## Baptiste Auguie (2017). gridExtra: Miscellaneous Functions for  
## "Grid" Graphics. R package version 2.3.  
## https://CRAN.R-project.org/package=gridExtra  
##   
## A BibTeX entry for LaTeX users is  
##   
## @Manual{,  
## title = {gridExtra: Miscellaneous Functions for "Grid" Graphics},  
## author = {Baptiste Auguie},  
## year = {2017},  
## note = {R package version 2.3},  
## url = {https://CRAN.R-project.org/package=gridExtra},  
## }

##Read in and organize data

#Read from csv file  
herring <- read.csv("HerringOtolithDatabase 15 Jan 2020\_JQ.csv", header=TRUE, fileEncoding="UTF-8-BOM", stringsAsFactors = FALSE)  
  
#Create a data frame  
herring\_data <- data.frame(oto.width = herring$AverageWidth,  
 sal.id = herring$FishCode,  
 sal.length = herring$SalmonLength,  
 coll.doy = herring$CollectionDayofYear,  
 lat = herring$Latitude,  
 long = herring$Longitude,  
 coll.month = herring$CollectionMonth,  
   
 sal.sp = herring$SalmonSpecies,  
 coll.year = herring$CollectionYear,  
 stat.area = as.factor(herring$StatArea))  
  
#Filter for 2018 chinook salmon only, omit northern BC areas (Haid Gwaii) and west coast Vancouver Island  
herring\_data <- herring\_data %>% filter(herring\_data$sal.sp == "ch" & herring\_data$coll.year == "2018" & herring\_data$stat.area != "1" & herring\_data$stat.area != "101" & herring\_data$stat.area != "125" & herring\_data$stat.area != "23" & herring\_data$stat.area != "25")

#Section 1: Introduction, Question, Goals and Hypotheses (400 words)

##A) Pacific Salmon in British Columbia are valued for their social, cultural and economic benefits to many communities on our coast. Despite their important role in the coastal marine ecosystem (Schindler et al. 2003), diet studies from this region are outdated and limited to summer sampling. The adult Chinook and Coho Salmon Diet Program (Juanes Lab) seeks to address these knowledge gaps by sourcing stomach samples from the recreational fishing community to gather regional, seasonal and interannual diet data to monitor ecosystem response to environmental change.

We have found Pacific Herring are the dominant fish consumed throughout regions of the Salish Sea, year-round. For this project we will investigate factors that may be important to the size of herring consumed by Chinook Salmon. Previous work found a positive, linear relationship between otolith width (mm) and length (mm) of herring from this dataset (Murchy et al. 2017). Otolith width (mm) measurements averaged from left and right otoliths will be used as a proxy for herring size (Stevenson and Campana, 1992) in this analysis. We will also use salmon catch data recorded by each angler and submitted with the associated stomach sample.

##B) How does Chinook size, catch location and time of year in the Salish Sea affect the size of herring prey consumed?

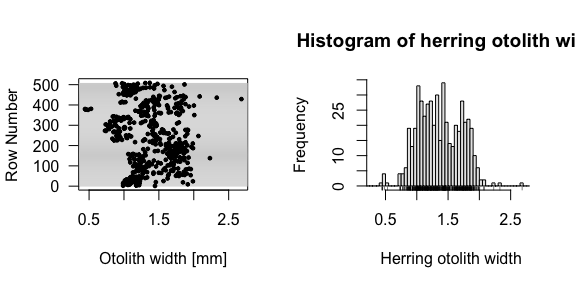
##C) This analysis is descriptive, with the goal of showing how explanatory variables describe patterns in herring size consumed by Chinook in the Salish Sea.

##D) We hypothesize that otolith width will be positively linear then asymptotic at maximum herring size as salmon length increases. Otolith width and day of year will show a quadratic, s-shaped relationship with the vertex midway through the calendar year. We expect a moderately positive, linear relationship with increasing latitude and a quadratic, concave up relationship with longitude with maximum herring size occurring in mid-channel waters. We expect that collinearity between latitude and longitude will cause in interaction effect on otolith width. An interaction may also exist between salmon length and latitude and between salmon length and day of year.

##E) We will use AIC fit with maximum likelihood estimation because it is flexible when comparing complex, non-nested models with parameters estimated from ecological field studies. AIC is commonly used in ecosystem studies where parameter estimates are relatively imprecise and we accept that the optimal model may be selected over the true, given the current sample size and our ecological understanding of the system.

#Section 2: The Response Variable (150 words) ##A)

par(mfrow = c(1, 2))  
  
#Add column for number of rows  
n <- nrow(herring\_data)  
herring\_data$row <- 1:n  
  
#Make dotchart of otolith width (response variable)  
with(herring\_data, plot(oto.width, row, las = 1, type = "n", xlab = "Otolith width [mm]", ylab = "Row Number"))  
abline (h = seq(1, n, by = 5), col = "lightgrey")  
with(herring\_data, points(oto.width, row, las = 1, type = "p", pch = 19, cex = 0.5))  
  
#Make histogram of response variable with rugmarks  
hist(herring\_data$oto.width, xlim = c(0.3, 2.8), breaks = seq(0.2, 2.8, by = 0.05), col = grey(0.9), main = "Histogram of herring otolith width", xlab = "Herring otolith width")  
rug(herring\_data$oto.width)



#Check for zeroes and NAs in otilith width  
table(herring\_data$oto.width > 0, useNA = "always")

##   
## TRUE <NA>   
## 508 1

#Exclude rows with NAs for continuous variables  
herring\_data <- herring\_data %>% na.omit()

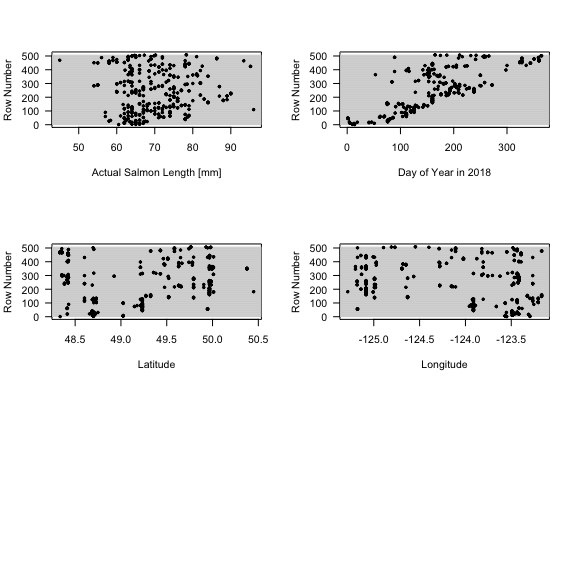
##B) The response variable (otolith width) is continuous and cannot take on zero or negative values. The range of possible response values in this dataset is 0.007 mm to 2.68 mm.

##C) Outliers at < 0.6 mm and > 2.2 mm are reasonable measurements representing a few individuals in the extreme age classes. There are no zero values and one missing otolith measurement making up 0.11 % of the data. We anticipate no difficulties in removing the single NA value from subsequent analyses. Non-independence is expected from repeated measurements of multiple herring that occurred within the same salmon stomach. Herring consumed at similar times of year may have similar growth trajectories that contribute to temporal autocorrelation. Spatial autocorrelation may occur if herring consumed in similar areas have similar prey fields and grow rates.

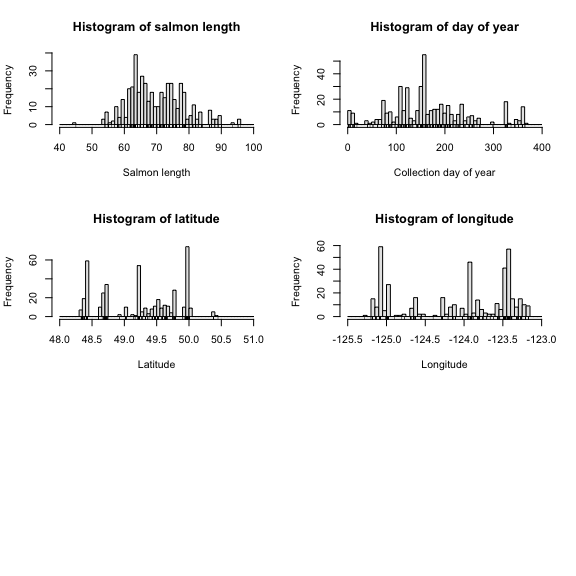
##D) We will consider the normal distribution for modelling.

#Section 3: The Explanatory Variables (150 words) ##A) ###Continuous explanatory variables

#Make multipanel of dotcharts for continuous explanatory variables  
par(mfrow = c(3, 2))  
  
#Salmon length dotchart  
with(herring\_data, plot(sal.length, row, las = 1, type = "n", xlab = "Actual Salmon Length [mm]", ylab = "Row Number"))  
abline (h = seq(1, n, by = 5), col = "lightgrey")  
with(herring\_data, points(sal.length, row, las = 1, type = "p", pch = 19, cex = 0.5))  
  
#Collection day of year dotchart  
with(herring\_data, plot(coll.doy, row, las = 1, type = "n", xlab = "Day of Year in 2018", ylab = "Row Number"))  
abline (h = seq(1, n, by = 5), col = "lightgrey")  
with(herring\_data, points(coll.doy, row, las = 1, type = "p", pch = 19, cex = 0.5))  
  
#Latitude dotchart  
with(herring\_data, plot(lat, row, las = 1, type = "n", xlab = "Latitude", ylab = "Row Number"))  
abline (h = seq(1, n, by = 5), col = "lightgrey")  
with(herring\_data, points(lat, row, las = 1, type = "p", pch = 19, cex = 0.5))  
  
#Longitude dotchart  
with(herring\_data, plot(long, row, las = 1, type = "n", xlab = "Longitude", ylab = "Row Number"))  
abline (h = seq(1, n, by = 5), col = "lightgrey")  
with(herring\_data, points(long, row, las = 1, type = "p", pch = 19, cex = 0.5))  
  
#Make multipanel of histograms for continuous explanatory variables  
par(mfrow = c(3, 2))

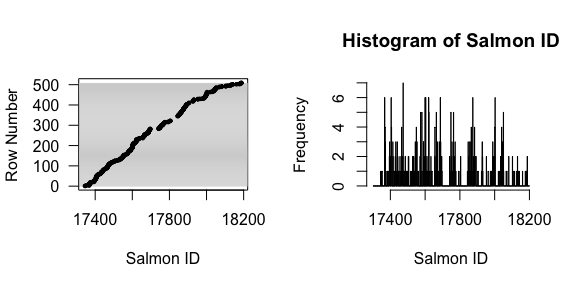


#Salmon length histogram  
hist(herring\_data$sal.length, col = grey(0.9), breaks = seq(40, 100, by = 1), main = "Histogram of salmon length", xlab = "Salmon length")  
rug(herring\_data$sal.length)  
  
#Collection day of year histogram  
hist(herring\_data$coll.doy, col = grey(0.9), breaks = seq(0, 400, by = 7), main = "Histogram of day of year", xlab = "Collection day of year")  
rug(herring\_data$coll.doy)  
  
#Latitude histogram  
hist(herring\_data$lat, col = grey(0.9), breaks = seq(48.0, 51.0, by = 0.05), main = "Histogram of latitude", xlab = "Latitude")  
rug(herring\_data$lat)  
  
#Longitude histogram  
hist(herring\_data$long, col = grey(0.9), breaks = seq(-125.5, -123, by = 0.05), main = "Histogram of longitude", xlab = "Longitude")  
rug(herring\_data$long)



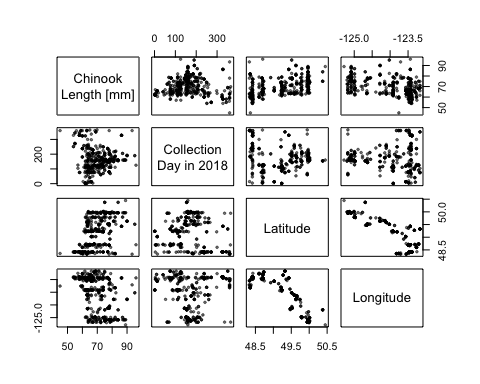
###Categorical explanatory variables

#Make multipanel for plots of categorical explanatory variables  
par(mfrow = c(1, 2))  
  
#Make dotchart of salmon id  
with(herring\_data, plot(sal.id, row, las = 1, type = "n", xlab = "Salmon ID", ylab = "Row Number"))  
abline (h = seq(1, n, by = 5), col = "lightgrey")  
with(herring\_data, points(sal.id, row, las = 1, type = "p", pch = 19, cex = 0.5))  
  
#Make a histogram of herring frequency per salmon  
hist(herring\_data$sal.id, col = grey(0.9), breaks = seq(17300, 18200, by = 1), main = "Histogram of Salmon ID", xlab = "Salmon ID")

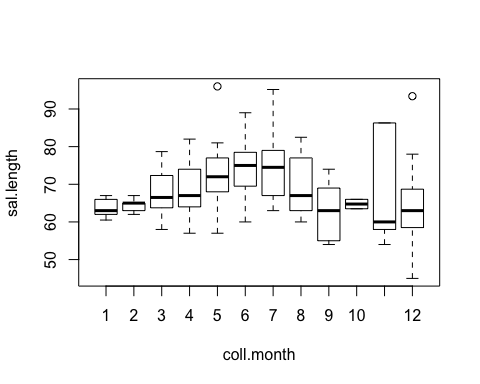
 ## B) One outlier is observed at 45cm and three outliers at 90 - 100cm. However, these are reasonable sizes for recreationally caught chinook salmon. No transformations are necessary as no substantial skewedness is observed.

#Section 4: Collinearity, Balance, and Variance Inﬂation Factors (200 words) ##A)

#Create dataframe for pairwise scatter plots that only includes continuous explanatory variables - salmon lenght, weight, collection day of year, latitude and longitude  
herring\_collinearity <- data.frame("sal.length" = herring\_data$sal.length,  
 "coll.doy" = herring\_data$coll.doy, "lat" = herring\_data$lat, "long" = herring\_data$long)  
  
#Make pairwise scatterplot of explanatory variables  
plot(herring\_collinearity[1:4], cex = 0.5, pch = 19, col = rgb(0, 0, 0, 0.5),   
 labels = c("Chinook\nLength [mm]", "Collection\nDay in 2018", "Latitude", "Longitude"))



#Plot mean salmon size vs. collection month  
boxplot(sal.length ~ coll.month, data = herring\_data)



#Calculate and view pairwise correlation coefficients for explanatory variables  
print(cor(na.omit(herring\_data[ ,c("sal.length", "coll.doy", "lat", "long")])), digits = 2)

## sal.length coll.doy lat long  
## sal.length 1.000 -0.0619 0.45 -0.3991  
## coll.doy -0.062 1.0000 -0.13 -0.0033  
## lat 0.449 -0.1284 1.00 -0.8721  
## long -0.399 -0.0033 -0.87 1.0000

#Calulate and view Variance Inflation Factors  
print(vif(na.omit(herring\_data[ ,c("sal.length", "coll.doy", "lat", "long")])))

## sal.length coll.doy lat long   
## 1.253474 1.077615 4.721313 4.428887

Our only categorical variable is salmon id so the range in the number of otoliths per salmon was x to x will not confounded any other categorical variables. We plan to model salmon id as a random effect which will account for the effects of any unbalancing associated with salmon id on the continuous variables.

A boxplot of mean salmon length per month shows that mean salmon length in summer months overlapped with mean salmon length observed in earlier months in the year, likely due to unbalanced sampling effort over the year as opposed to the concave up shape observed in the pairwise scatterplot. The remaining relationships observed in the pairwise scatterplots were consistent with the pairwise correlation coefficients. Salmon length was moderately, positively correlated (0.449) with latitude but moderately, negatively correlated longitude (-0.3991). There was a strong, negative correlation (-0.8721) between latitude and longitude. VIF values of latitude (4.721313) and longitude (4.428887) were greater than 3 and strongly colinear, which is expected due to the geographical constraints of the Strait of Georgia and Juan de Fuca. We will run models with either latitude or longitude to avoid the confounding effects of including both variables in a single model.

#Section 5: Statistical Methods and Model Fitting (200 words)

We will be using linear mixed-effects models using the R Package lme4 (Bates et al., 2015). We are considering the normal distribution of otolith width (mm) modelled as function of salmon length (mm), day of year in 2018, latitude and longitude. We will include a random intercept corresponding to individual salmon because there are > 10 levels (ref.). Ecologically, it is likely that multiple herring in one stomach have similar sizes if they travel in the same school targeted by an individual. In our first round of modelling fitting all salmon lengths will be included followed by a second round of model fitting to salmon 62 cm and above as this is the regulation size restrictions for recreationally caught salmon in the Strait of Georgia. This will check the influence of removing under size regulation fish from the dataset.

\*\*come back to salmon size subset justification.

##Standardizing data for models

#Add columns for standardized continuous predictor variable for modelling (x - minus mean / 1 sd)  
herring\_data$sal.length\_standardized <- (herring\_data$sal.length - mean(herring\_data$sal.length)) / sd(herring\_data$sal.length)  
herring\_data$coll.doy\_standardized <- (herring\_data$coll.doy - mean(herring\_data$coll.doy)) / sd(herring\_data$coll.doy)  
herring\_data$lat\_standardized <- (herring\_data$lat - mean(herring\_data$lat)) / sd(herring\_data$lat)  
herring\_data$long\_standardized <- (herring\_data$long - mean(herring\_data$long)) / sd(herring\_data$long)

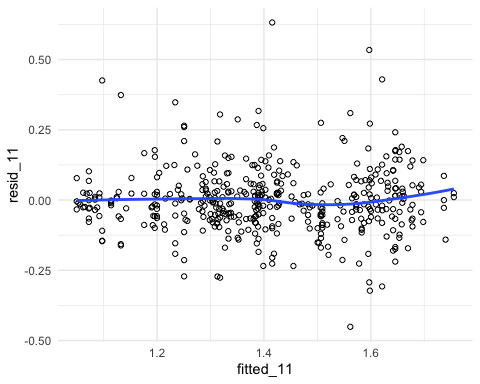
##Creating linear mixed-effect models

#Intercept-only model  
model\_0 <- lmer(oto.width ~ (1 | sal.id), data = herring\_data)  
  
#Models with one explanatory variable  
model\_1 <- lmer(oto.width ~ sal.length\_standardized + (1 | sal.id), data = herring\_data)  
  
model\_2 <- lmer(oto.width ~ coll.doy\_standardized + (1 | sal.id), data = herring\_data)  
  
model\_3 <- lmer(oto.width ~ lat\_standardized + (1 | sal.id), data = herring\_data)  
  
model\_4 <- lmer(oto.width ~ long\_standardized + (1 | sal.id), data = herring\_data)  
  
#Models with two explanatory variables  
model\_5 <- lmer(oto.width ~ coll.doy\_standardized + sal.length\_standardized + (1 | sal.id), data = herring\_data)  
  
model\_6 <- lmer(oto.width ~ coll.doy\_standardized + long\_standardized + (1 | sal.id), data = herring\_data)  
  
model\_7 <- lmer(oto.width ~ coll.doy\_standardized + lat\_standardized + (1 | sal.id), data = herring\_data)  
  
model\_8 <- lmer(oto.width ~ sal.length\_standardized + long\_standardized + (1 | sal.id), data = herring\_data)  
  
model\_9 <- lmer(oto.width ~ sal.length\_standardized + lat\_standardized + (1 | sal.id), data = herring\_data)  
  
#Models with three explanatory variables  
model\_10 <- lmer(oto.width ~ coll.doy\_standardized + lat\_standardized + sal.length\_standardized + (1 | sal.id), data = herring\_data)  
  
model\_11 <- lmer(oto.width ~ coll.doy\_standardized + long\_standardized + sal.length\_standardized + (1 | sal.id), data = herring\_data)  
  
#Models with two explanatory variables and one interaction term  
model\_5i <- lmer(oto.width ~ coll.doy\_standardized\*sal.length\_standardized + (1 | sal.id), data = herring\_data)  
  
model\_6i <- lmer(oto.width ~ coll.doy\_standardized\*long\_standardized + (1 | sal.id), data = herring\_data)  
  
model\_7i <- lmer(oto.width ~ coll.doy\_standardized\*lat\_standardized + (1 | sal.id), data = herring\_data)  
  
model\_8i <- lmer(oto.width ~ sal.length\_standardized\*long\_standardized + (1 | sal.id), data = herring\_data)  
  
model\_9i <- lmer(oto.width ~ sal.length\_standardized\*lat\_standardized + (1 | sal.id), data = herring\_data)  
  
#Models with three explanatory variables and one interaction term  
model\_10l <- lmer(oto.width ~ coll.doy\_standardized + lat\_standardized\*sal.length\_standardized + (1 | sal.id), data = herring\_data)  
  
model\_11l <- lmer(oto.width ~ coll.doy\_standardized + long\_standardized\*sal.length\_standardized + (1 | sal.id), data = herring\_data)  
  
model\_10c <- lmer(oto.width ~ coll.doy\_standardized\*sal.length\_standardized + lat\_standardized + (1 | sal.id), data = herring\_data)  
  
model\_11c <- lmer(oto.width ~ coll.doy\_standardized\*sal.length\_standardized + long\_standardized + (1 | sal.id), data = herring\_data)  
  
#Models with three explanatory variables and two interaction terms  
model\_10cl <- lmer(oto.width ~ coll.doy\_standardized\*sal.length\_standardized\*lat\_standardized + (1 | sal.id), data = herring\_data)  
  
model\_11cl <- lmer(oto.width ~ coll.doy\_standardized\*sal.length\_standardized\*long\_standardized + (1 | sal.id), data = herring\_data)

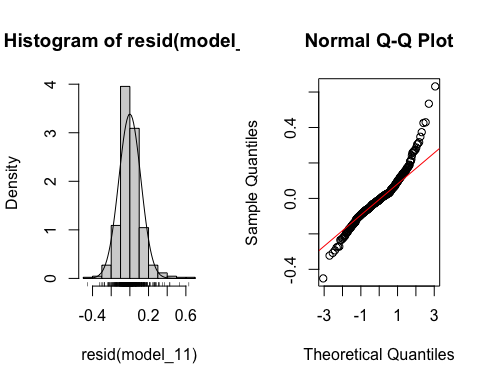
#Section 6: Model Checking ##Checking assumptions of model 11

#Plot residuals versus fitted values (shown with a smoothing curve)  
herring\_data$resid\_11 <- resid(model\_11)   
herring\_data$fitted\_11 <- predict(model\_11, re.form = ~0)  
  
ggplot(herring\_data) +  
 theme\_minimal() +   
 geom\_point(aes(fitted\_11, resid\_11), shape = 1) +  
 geom\_smooth(aes(fitted\_11, resid\_11), se = FALSE)

## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'



#Make multipanel of residual plots  
par(mfrow = c(1, 2))  
  
#Plot histogram of residuals as probability density with rug marks and reference curve  
hist(resid(model\_11), prob = TRUE, col = "lightgrey")  
rug(resid(model\_11))  
sd\_hat <- sd(resid(model\_11))  
curve(dnorm(x, sd = sd\_hat), add = TRUE)  
  
#Make quantile-quantile plot with reference line  
qqnorm(resid(model\_11))  
qqline(resid(model\_11), col = "red")

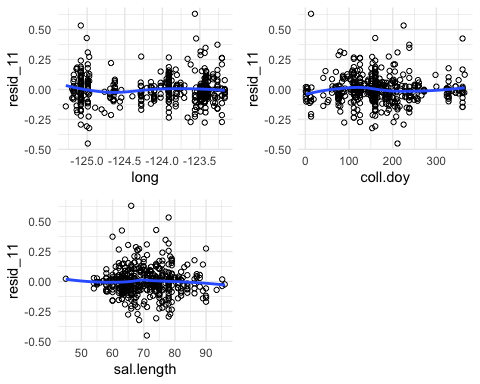


#Run a Shapiro-Wilk test for normality of the residuals  
shapiro.test(resid(model\_11))

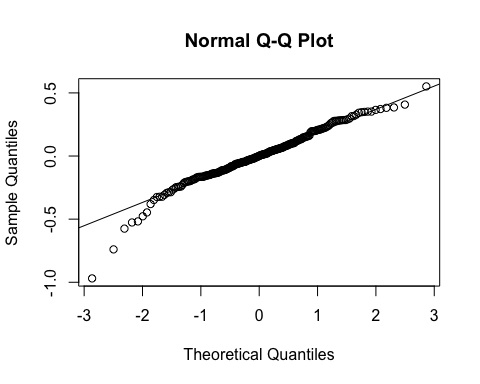
##   
## Shapiro-Wilk normality test  
##   
## data: resid(model\_11)  
## W = 0.94396, p-value = 7.724e-12

#Plot residuals vs. longitude (shown with smoothing curve)  
resid\_vs\_long\_11 <- ggplot(herring\_data) +  
 theme\_minimal() +   
 geom\_point(aes(long, resid\_11), shape = 1) +  
 geom\_smooth(aes(long, resid\_11), se = FALSE)  
  
#Plot residuals vs. coll.doy (shown with smoothing curve)  
resid\_vs\_coll.doy\_11 <- ggplot(herring\_data) +  
 theme\_minimal() +   
 geom\_point(aes(coll.doy, resid\_11), shape = 1) +  
 geom\_smooth(aes(coll.doy, resid\_11), se = FALSE)  
  
#Plot residuals vs. sal.length (shown with smoothing curve)  
resid\_vs\_sal.length\_11 <- ggplot(herring\_data) +  
 theme\_minimal() +   
 geom\_point(aes(sal.length, resid\_11), shape = 1) +  
 geom\_smooth(aes(sal.length, resid\_11), se = FALSE)  
  
grid.arrange(resid\_vs\_long\_11, resid\_vs\_coll.doy\_11, resid\_vs\_sal.length\_11, nrow = 2)

## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'  
## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'  
## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'

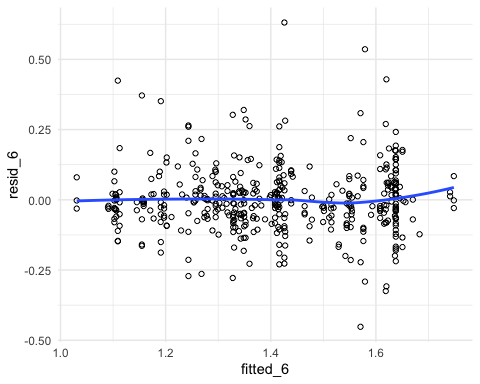


#Make multipanel of residual plots  
par(mfrow = c(1, 1))  
  
#Make quantile-quantile plot for checking whether the random intercept is normally distributed  
qqnorm(ranef(model\_11)$sal.id[, "(Intercept)"])  
qqline(ranef(model\_11)$sal.id[, "(Intercept)"])

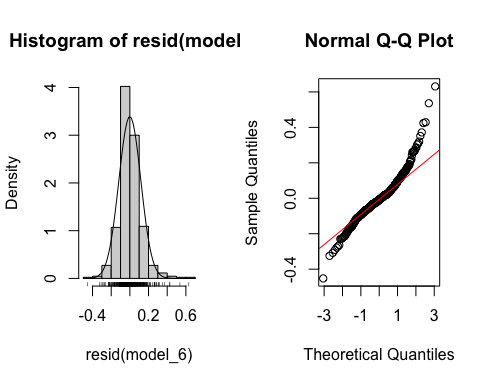
 ##Checking assumptions of model 6

#Plot residuals versus fitted values (shown with a smoothing curve)  
herring\_data$resid\_6 <- resid(model\_6)   
herring\_data$fitted\_6 <- predict(model\_6, re.form = ~0)  
  
ggplot(herring\_data) +  
 theme\_minimal() +   
 geom\_point(aes(fitted\_6, resid\_6), shape = 1) +  
 geom\_smooth(aes(fitted\_6, resid\_6), se = FALSE)

## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'



#Make multipanel of residual plots  
par(mfrow = c(1, 2))  
  
#Plot histogram of residuals as probability density with rug marks and reference curve  
hist(resid(model\_6), prob = TRUE, col = "lightgrey")  
rug(resid(model\_6))  
sd\_hat <- sd(resid(model\_6))  
curve(dnorm(x, sd = sd\_hat), add = TRUE)  
  
#Make quantile-quantile plot with reference line  
qqnorm(resid(model\_6))  
qqline(resid(model\_6), col = "red")

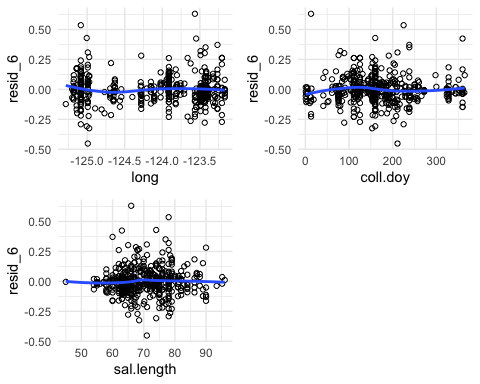


#Run a Shapiro-Wilk test for normality of the residuals  
shapiro.test(resid(model\_6))

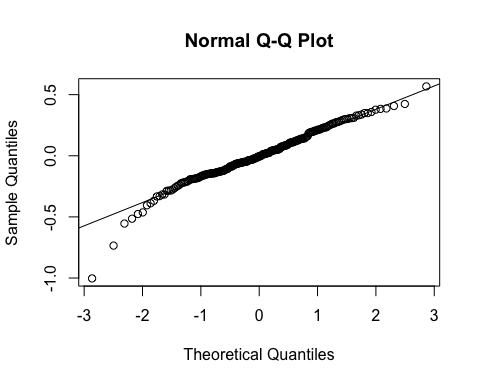
##   
## Shapiro-Wilk normality test  
##   
## data: resid(model\_6)  
## W = 0.94325, p-value = 6.278e-12

#Plot residuals vs. longitude (shown with smoothing curve)  
resid\_vs\_long\_6 <- ggplot(herring\_data) +  
 theme\_minimal() +   
 geom\_point(aes(long, resid\_6), shape = 1) +  
 geom\_smooth(aes(long, resid\_6), se = FALSE)  
  
#Plot residuals vs. coll.doy (shown with smoothing curve)  
resid\_vs\_coll.doy\_6 <- ggplot(herring\_data) +  
 theme\_minimal() +   
 geom\_point(aes(coll.doy, resid\_6), shape = 1) +  
 geom\_smooth(aes(coll.doy, resid\_6), se = FALSE)  
  
#Plot residuals vs. sal.length (shown with smoothing curve)  
resid\_vs\_sal.length\_6 <- ggplot(herring\_data) +  
 theme\_minimal() +   
 geom\_point(aes(sal.length, resid\_6), shape = 1) +  
 geom\_smooth(aes(sal.length, resid\_6), se = FALSE)  
  
grid.arrange(resid\_vs\_long\_6, resid\_vs\_coll.doy\_6, resid\_vs\_sal.length\_6, nrow = 2)

## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'  
## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'  
## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'



#Make multipanel of residual plots  
par(mfrow = c(1, 1))  
  
#Make quantile-quantile plot for checking whether the random intercept is normally distributed  
qqnorm(ranef(model\_6)$sal.id[, "(Intercept)"])  
qqline(ranef(model\_6)$sal.id[, "(Intercept)"])

 1. Written summary (200 words) -assessment of plausibility of model assumptions (LINE) -check for correlation in the residuals due to grouping variables (e.g., repeated measurements on individuals or within study sites), spatial autocorrelation and/or temporal autocorrelation -comment on potential outliers, as relevant

#Section 7: Model Summary, Conﬁdence Intervals and Model Comparison ##Rewriting all models with REML= FALSE

#Intercept-only model  
model\_0 <- lmer(oto.width ~ (1 | sal.id), data = herring\_data, REML= FALSE)  
  
#Models with one explanatory variable  
model\_1 <- lmer(oto.width ~ sal.length\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
model\_2 <- lmer(oto.width ~ coll.doy\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
model\_3 <- lmer(oto.width ~ lat\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
model\_4 <- lmer(oto.width ~ long\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
#Models with two explanatory variables  
model\_5 <- lmer(oto.width ~ coll.doy\_standardized + sal.length\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
model\_6 <- lmer(oto.width ~ coll.doy\_standardized + long\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
model\_7 <- lmer(oto.width ~ coll.doy\_standardized + lat\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
model\_8 <- lmer(oto.width ~ sal.length\_standardized + long\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
model\_9 <- lmer(oto.width ~ sal.length\_standardized + lat\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
#Models with three explanatory variables  
model\_10 <- lmer(oto.width ~ coll.doy\_standardized + lat\_standardized + sal.length\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
model\_11 <- lmer(oto.width ~ coll.doy\_standardized + long\_standardized + sal.length\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
#Models with two explanatory variables and one interaction term  
model\_5i <- lmer(oto.width ~ coll.doy\_standardized\*sal.length\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
model\_6i <- lmer(oto.width ~ coll.doy\_standardized\*long\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
model\_7i <- lmer(oto.width ~ coll.doy\_standardized\*lat\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
model\_8i <- lmer(oto.width ~ sal.length\_standardized\*long\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
model\_9i <- lmer(oto.width ~ sal.length\_standardized\*lat\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
#Models with three explanatory variables and one interaction term  
model\_10l <- lmer(oto.width ~ coll.doy\_standardized + lat\_standardized\*sal.length\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
model\_11l <- lmer(oto.width ~ coll.doy\_standardized + long\_standardized\*sal.length\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
model\_10c <- lmer(oto.width ~ coll.doy\_standardized\*sal.length\_standardized + lat\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
model\_11c <- lmer(oto.width ~ coll.doy\_standardized\*sal.length\_standardized + long\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
#Models with three explanatory variables and two interaction terms  
model\_10cl <- lmer(oto.width ~ coll.doy\_standardized\*sal.length\_standardized\*lat\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)  
  
model\_11cl <- lmer(oto.width ~ coll.doy\_standardized\*sal.length\_standardized\*long\_standardized + (1 | sal.id), data = herring\_data, REML= FALSE)

##Running the AIC

#Calculate AIC values  
AIC\_out <- AIC(model\_0, model\_1, model\_2, model\_3, model\_4, model\_5, model\_6, model\_7, model\_8, model\_9, model\_5i, model\_6i, model\_7i, model\_8i, model\_9i, model\_10, model\_11, model\_10l, model\_10c, model\_11l, model\_11c, model\_10cl, model\_11cl)  
  
#Calculate delta AIC values  
AIC\_out$delta\_AIC <- AIC\_out$AIC - min(AIC\_out$AIC)  
  
#Show AIC table with weights  
AICtab(model\_0, model\_1, model\_2, model\_3, model\_4, model\_5, model\_6, model\_7, model\_8, model\_9, model\_5i, model\_6i, model\_7i, model\_8i, model\_9i, model\_10, model\_11, model\_10l, model\_10c, model\_11l, model\_11c, model\_10cl, model\_11cl, base = TRUE, weights = TRUE)

## AIC dAIC df weight  
## model\_11 -21.3 0.0 6 0.328   
## model\_6 -20.3 1.0 5 0.198   
## model\_11c -20.1 1.2 7 0.179   
## model\_11l -19.7 1.6 7 0.146   
## model\_6i -19.5 1.8 6 0.133   
## model\_11cl -15.3 6.1 10 0.016   
## model\_8 -7.3 14.0 5 <0.001  
## model\_8i -5.5 15.8 6 <0.001  
## model\_4 -5.3 16.0 4 <0.001  
## model\_10 -1.6 19.7 6 <0.001  
## model\_10c -0.8 20.6 7 <0.001  
## model\_10l -0.2 21.1 7 <0.001  
## model\_7 0.3 21.7 5 <0.001  
## model\_7i 2.1 23.4 6 <0.001  
## model\_9 3.9 25.3 5 <0.001  
## model\_10cl 5.0 26.3 10 <0.001  
## model\_9i 5.8 27.1 6 <0.001  
## model\_3 5.9 27.2 4 <0.001  
## model\_5 30.6 51.9 5 <0.001  
## model\_5i 31.4 52.7 6 <0.001  
## model\_1 37.2 58.5 4 <0.001  
## model\_2 47.4 68.7 4 <0.001  
## model\_0 54.5 75.8 3 <0.001

##Summarizing model 11

#Call coeﬃcient estimates for fixed effects  
fixef(model\_11)

## (Intercept) coll.doy\_standardized long\_standardized   
## 1.40823938 -0.07112385 -0.14511708   
## sal.length\_standardized   
## 0.03339340

#Call coeﬃcient estimates for random effects  
ranef(model\_11)

## $sal.id  
## (Intercept)  
## 17345 3.839366e-02  
## 17349 -2.557192e-01  
## 17353 -1.231612e-01  
## 17368 -1.329946e-01  
## 17369 -2.407656e-01  
## 17370 -3.060796e-01  
## 17372 -2.057584e-01  
## 17374 -1.848628e-01  
## 17382 3.497390e-01  
## 17383 4.190458e-02  
## 17392 -1.132955e-01  
## 17393 -1.046923e-01  
## 17394 -1.804837e-01  
## 17395 3.482364e-01  
## 17396 -1.158095e-01  
## 17397 -2.655593e-01  
## 17398 -1.351614e-01  
## 17400 -1.902822e-01  
## 17402 -1.845819e-01  
## 17405 -8.211657e-02  
## 17406 -2.909760e-01  
## 17407 -2.860386e-01  
## 17408 2.622459e-01  
## 17409 2.075253e-01  
## 17410 2.139131e-01  
## 17415 1.323746e-01  
## 17416 7.941823e-02  
## 17421 1.506659e-01  
## 17425 3.852776e-03  
## 17428 6.064999e-02  
## 17429 3.834160e-01  
## 17430 2.833972e-01  
## 17431 9.937773e-02  
## 17432 -9.133995e-02  
## 17433 -4.653457e-02  
## 17434 3.135194e-01  
## 17440 -1.358328e-01  
## 17441 4.799893e-02  
## 17442 2.083092e-01  
## 17443 -8.695398e-03  
## 17446 1.210068e-01  
## 17448 3.147665e-01  
## 17449 -1.059795e-01  
## 17455 2.044091e-01  
## 17456 3.640150e-01  
## 17461 1.301916e-01  
## 17462 1.938045e-01  
## 17464 1.201371e-02  
## 17465 -6.633453e-02  
## 17466 -1.409698e-01  
## 17467 -2.531202e-02  
## 17468 -2.067718e-02  
## 17469 -1.393902e-01  
## 17470 2.211832e-01  
## 17474 1.964844e-01  
## 17478 -5.438970e-02  
## 17479 -1.104686e-01  
## 17480 1.518155e-01  
## 17483 5.090278e-02  
## 17485 1.972429e-01  
## 17486 2.571816e-01  
## 17490 -1.552627e-01  
## 17493 8.303053e-02  
## 17495 -2.298487e-02  
## 17498 -6.060544e-02  
## 17501 3.699436e-01  
## 17504 3.457668e-01  
## 17506 -4.736766e-02  
## 17516 1.000071e-02  
## 17519 -8.151039e-02  
## 17523 3.253812e-01  
## 17529 2.252082e-01  
## 17534 -1.028577e-01  
## 17535 -1.544567e-01  
## 17538 1.323983e-02  
## 17540 2.777821e-01  
## 17541 1.599318e-01  
## 17543 4.047983e-01  
## 17544 2.735661e-01  
## 17548 1.360879e-01  
## 17550 1.005230e-01  
## 17551 9.405641e-03  
## 17552 2.432509e-01  
## 17553 -4.610406e-02  
## 17554 -6.035737e-02  
## 17555 -1.359683e-01  
## 17559 -7.412553e-02  
## 17563 2.894842e-01  
## 17571 -1.652265e-01  
## 17575 3.589575e-02  
## 17581 -3.569267e-02  
## 17582 1.494153e-02  
## 17588 3.329142e-02  
## 17591 9.251089e-02  
## 17592 5.656129e-02  
## 17593 -3.206965e-01  
## 17598 1.949588e-01  
## 17600 1.470793e-01  
## 17602 8.275064e-02  
## 17604 -8.797901e-03  
## 17605 1.017719e-01  
## 17606 1.450250e-01  
## 17611 1.021860e-01  
## 17612 -5.381649e-02  
## 17614 -9.366024e-02  
## 17616 9.141201e-02  
## 17617 1.988497e-01  
## 17618 -1.492982e-01  
## 17620 3.322417e-02  
## 17621 -3.108175e-02  
## 17622 -4.764822e-01  
## 17625 4.858350e-02  
## 17628 1.095664e-01  
## 17634 -4.097124e-05  
## 17635 -5.147131e-01  
## 17638 -8.125019e-02  
## 17640 4.590663e-02  
## 17656 -1.510509e-01  
## 17657 -2.768614e-02  
## 17658 3.499266e-01  
## 17662 -2.033141e-01  
## 17663 -6.008674e-02  
## 17664 2.807814e-01  
## 17665 -1.497248e-01  
## 17666 -1.135776e-01  
## 17673 -1.632151e-01  
## 17675 2.061909e-01  
## 17676 -4.060199e-03  
## 17680 2.833336e-01  
## 17684 2.042679e-02  
## 17687 2.194608e-01  
## 17688 -2.861715e-01  
## 17689 1.158268e-01  
## 17690 -7.376450e-01  
## 17692 1.559271e-01  
## 17694 2.147340e-01  
## 17695 1.272053e-01  
## 17696 -3.233896e-01  
## 17697 -3.507579e-02  
## 17742 -6.613457e-02  
## 17743 -1.966644e-01  
## 17748 -1.278815e-01  
## 17749 -9.689210e-02  
## 17752 -5.728565e-01  
## 17753 -1.453657e-01  
## 17755 1.182317e-01  
## 17756 1.845961e-01  
## 17760 7.700988e-02  
## 17764 -5.253402e-01  
## 17766 2.315892e-01  
## 17769 -1.835129e-02  
## 17772 6.459173e-02  
## 17773 -1.392239e-01  
## 17776 2.825849e-01  
## 17777 2.839658e-02  
## 17793 -2.791957e-02  
## 17795 -2.402575e-01  
## 17805 2.002560e-02  
## 17844 -1.710445e-01  
## 17847 -1.643221e-01  
## 17848 -2.457777e-01  
## 17850 9.604675e-02  
## 17851 -2.273464e-01  
## 17857 -1.332940e-01  
## 17858 1.002560e-01  
## 17859 1.974242e-01  
## 17861 6.764210e-02  
## 17863 2.042187e-02  
## 17864 -1.996616e-01  
## 17865 7.786137e-03  
## 17866 -2.445951e-01  
## 17868 1.551449e-01  
## 17872 1.609524e-01  
## 17875 -9.679317e-01  
## 17878 1.674424e-02  
## 17882 -4.528443e-02  
## 17883 -1.221851e-01  
## 17885 6.208154e-02  
## 17887 -2.124491e-01  
## 17888 -1.640096e-01  
## 17889 2.953456e-01  
## 17890 2.767713e-01  
## 17891 1.459656e-01  
## 17892 7.761564e-02  
## 17893 4.486387e-02  
## 17904 2.425101e-02  
## 17905 -1.669020e-02  
## 17925 3.407153e-01  
## 17929 4.048631e-02  
## 17930 1.510586e-02  
## 17931 -4.676607e-02  
## 17932 6.877370e-02  
## 17953 5.494683e-01  
## 17963 -1.656092e-02  
## 17976 5.406567e-02  
## 17982 -2.946042e-02  
## 17984 1.216512e-01  
## 17986 -8.292751e-02  
## 17989 2.774927e-01  
## 17990 8.500161e-02  
## 17993 -7.858752e-02  
## 17994 2.355197e-01  
## 17995 -9.671035e-02  
## 17996 5.719810e-03  
## 17997 -5.241184e-03  
## 17999 -2.810468e-04  
## 18000 -3.450050e-02  
## 18001 -1.092086e-01  
## 18002 -4.028743e-02  
## 18003 -1.036118e-02  
## 18022 3.794101e-01  
## 18030 -1.598867e-01  
## 18034 -5.134237e-02  
## 18035 3.867805e-02  
## 18040 5.165066e-02  
## 18041 5.634415e-02  
## 18044 -6.725850e-02  
## 18045 -1.689225e-01  
## 18046 9.096618e-02  
## 18051 5.801859e-02  
## 18052 -5.291139e-02  
## 18067 -2.186313e-02  
## 18071 -5.570592e-02  
## 18082 6.440390e-02  
## 18103 -1.994496e-01  
## 18107 1.320936e-01  
## 18115 -3.202423e-01  
## 18128 -1.643456e-01  
## 18129 7.053776e-02  
## 18132 -1.537192e-01  
## 18136 -1.129127e-01  
## 18142 2.804373e-01  
## 18161 -6.031112e-02  
## 18177 -3.459866e-01  
## 18180 -4.354657e-02  
## 18181 -1.299341e-01  
## 18182 -4.440099e-01  
## 18185 -3.780038e-01  
## 18188 -1.642227e-01  
##   
## with conditional variances for "sal.id"

#Calculate conﬁdence intervals for coeﬃcient estimate  
confint(model\_11)

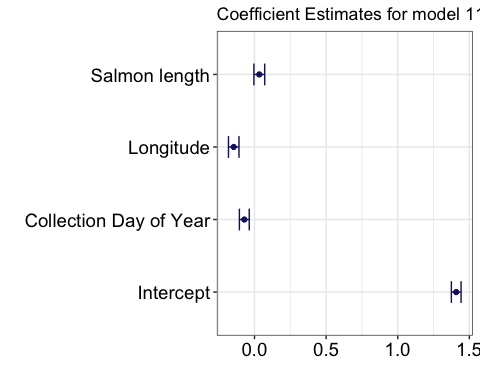
## Computing profile confidence intervals ...

## 2.5 % 97.5 %  
## .sig01 0.205272227 0.25963048  
## .sigma 0.140630587 0.17004989  
## (Intercept) 1.374474504 1.44215054  
## coll.doy\_standardized -0.105569467 -0.03672275  
## long\_standardized -0.181799245 -0.10839302  
## sal.length\_standardized -0.004343185 0.07109330

#Rearrange data for a plot of coefficients with standard errors  
#Make data frame for plot of coefficients  
confint\_fixed\_11 <- as.data.frame(confint(model\_11, parm = c("(Intercept)", "coll.doy\_standardized", "long\_standardized", "sal.length\_standardized")))

## Computing profile confidence intervals ...

##Compute profile confidence intervals  
confint\_fixed\_11 <- as.data.frame(confint\_fixed\_11)  
confint\_fixed\_11$term <- rownames(confint\_fixed\_11)  
confint\_fixed\_11$estimate <- fixef(model\_11)  
  
##ggplot  
ggplot(confint\_fixed\_11) +   
 theme\_bw() +  
 geom\_point(aes(y = term, x = estimate), colour = "midnightblue") +  
 geom\_errorbarh(aes(y = term, xmin = `2.5 %`, xmax = `97.5 %`), colour = "midnightblue", height = 0.3) +  
 scale\_y\_discrete(labels = c("Intercept", "Collection Day of Year", "Longitude", "Salmon length")) +  
 xlab("") +  
 ylab("") +   
 ggtitle("Coefficient Estimates for model 11") +  
 theme(axis.text = element\_text(size = 14, colour = "black"))

 ##Summarizing model 6

#Call coeﬃcient estimates for fixed effects  
fixef(model\_6)

## (Intercept) coll.doy\_standardized long\_standardized   
## 1.40569338 -0.07358628 -0.15719183

#Call coeﬃcient estimates for random effects  
ranef(model\_6)

## $sal.id  
## (Intercept)  
## 17345 0.016148755  
## 17349 -0.263972703  
## 17353 -0.133493363  
## 17368 -0.142912071  
## 17369 -0.244831033  
## 17370 -0.329475495  
## 17372 -0.214231877  
## 17374 -0.197603237  
## 17382 0.335693391  
## 17383 0.039315167  
## 17392 -0.125538330  
## 17393 -0.116758266  
## 17394 -0.180882557  
## 17395 0.348402677  
## 17396 -0.148575723  
## 17397 -0.275807749  
## 17398 -0.151493008  
## 17400 -0.217586690  
## 17402 -0.190304302  
## 17405 -0.043732664  
## 17406 -0.315631060  
## 17407 -0.316264788  
## 17408 0.242538056  
## 17409 0.212760781  
## 17410 0.206204476  
## 17415 0.122626438  
## 17416 0.091418215  
## 17421 0.157517155  
## 17425 -0.024488977  
## 17428 0.050966419  
## 17429 0.406125069  
## 17430 0.261525186  
## 17431 0.082107371  
## 17432 -0.057590665  
## 17433 -0.058994548  
## 17434 0.305836290  
## 17440 -0.147939868  
## 17441 0.080340432  
## 17442 0.213816432  
## 17443 -0.006577806  
## 17446 0.100733257  
## 17448 0.329799342  
## 17449 -0.118765067  
## 17455 0.189725098  
## 17456 0.357336119  
## 17461 0.115831860  
## 17462 0.202105577  
## 17464 -0.014267832  
## 17465 -0.087341660  
## 17466 -0.124076111  
## 17467 -0.028012680  
## 17468 -0.035611397  
## 17469 -0.135841914  
## 17470 0.230297211  
## 17474 0.203547431  
## 17478 0.026139498  
## 17479 -0.097561606  
## 17480 0.187845651  
## 17483 0.050669725  
## 17485 0.183313411  
## 17486 0.233688018  
## 17490 -0.142601054  
## 17493 0.071002464  
## 17495 -0.032037005  
## 17498 -0.050230091  
## 17501 0.381925340  
## 17504 0.376058961  
## 17506 -0.035693348  
## 17516 0.020627520  
## 17519 -0.066971393  
## 17523 0.309243724  
## 17529 0.208402202  
## 17534 -0.074895461  
## 17535 -0.130685433  
## 17538 0.018461353  
## 17540 0.295934308  
## 17541 0.141977255  
## 17543 0.421842460  
## 17544 0.291060527  
## 17548 0.106300751  
## 17550 0.121759478  
## 17551 -0.003809655  
## 17552 0.247650528  
## 17553 -0.021997059  
## 17554 -0.063563482  
## 17555 -0.130717191  
## 17559 -0.075696453  
## 17563 0.299704720  
## 17571 -0.150338726  
## 17575 0.072854298  
## 17581 -0.055657395  
## 17582 0.022200965  
## 17588 0.050166841  
## 17591 0.141826603  
## 17592 0.057719910  
## 17593 -0.282936032  
## 17598 0.225207499  
## 17600 0.154595505  
## 17602 0.085369472  
## 17604 -0.005231441  
## 17605 0.138655931  
## 17606 0.194965021  
## 17611 0.083957148  
## 17612 -0.052573538  
## 17614 -0.123357197  
## 17616 0.074648762  
## 17617 0.228544045  
## 17618 -0.160215440  
## 17620 0.022681832  
## 17621 -0.019452820  
## 17622 -0.404265551  
## 17625 0.043831042  
## 17628 0.116621090  
## 17634 -0.033665504  
## 17635 -0.512596923  
## 17638 -0.106549222  
## 17640 0.012820744  
## 17656 -0.173060505  
## 17657 -0.048110877  
## 17658 0.347623672  
## 17662 -0.211452039  
## 17663 -0.061230462  
## 17664 0.298889496  
## 17665 -0.110224238  
## 17666 -0.143066352  
## 17673 -0.188586242  
## 17675 0.193370504  
## 17676 -0.019338067  
## 17680 0.306318380  
## 17684 0.024692186  
## 17687 0.231170508  
## 17688 -0.283969443  
## 17689 0.126087160  
## 17690 -0.733385292  
## 17692 0.135553997  
## 17694 0.260726295  
## 17695 0.127886997  
## 17696 -0.331927521  
## 17697 -0.057091564  
## 17742 -0.106570946  
## 17743 -0.182012044  
## 17748 -0.141238079  
## 17749 -0.137824114  
## 17752 -0.552929554  
## 17753 -0.109163314  
## 17755 0.089806752  
## 17756 0.194374315  
## 17760 0.111261776  
## 17764 -0.475670558  
## 17766 0.215327845  
## 17769 -0.052007626  
## 17772 0.077567013  
## 17773 -0.153783689  
## 17776 0.284028483  
## 17777 0.008814310  
## 17793 -0.032583151  
## 17795 -0.251772072  
## 17805 0.040798229  
## 17844 -0.187003280  
## 17847 -0.160891008  
## 17848 -0.237191771  
## 17850 0.109795053  
## 17851 -0.189676195  
## 17857 -0.131651075  
## 17858 0.125957718  
## 17859 0.219525776  
## 17861 0.084409577  
## 17863 -0.007985752  
## 17864 -0.224242457  
## 17865 0.001481756  
## 17866 -0.207395700  
## 17868 0.144702818  
## 17872 0.140399424  
## 17875 -1.002047553  
## 17878 0.012527084  
## 17882 -0.065459187  
## 17883 -0.144357073  
## 17885 0.043691050  
## 17887 -0.182591388  
## 17888 -0.157818505  
## 17889 0.264850625  
## 17890 0.274109111  
## 17891 0.161144843  
## 17892 0.075070618  
## 17893 0.035694548  
## 17904 0.014862064  
## 17905 0.002542233  
## 17925 0.326820646  
## 17929 0.046930250  
## 17930 0.046480499  
## 17931 0.038932656  
## 17932 0.106980881  
## 17953 0.565928168  
## 17963 -0.002966620  
## 17976 0.049759772  
## 17982 -0.049627497  
## 17984 0.110230150  
## 17986 -0.094736579  
## 17989 0.279356612  
## 17990 0.099104386  
## 17993 -0.085619015  
## 17994 0.252664408  
## 17995 -0.086156568  
## 17996 0.021324854  
## 17997 -0.030672429  
## 17999 -0.035622576  
## 18000 -0.066891415  
## 18001 -0.121178434  
## 18002 -0.060393445  
## 18003 -0.038495596  
## 18022 0.385157074  
## 18030 -0.085608021  
## 18034 -0.019662962  
## 18035 0.012949974  
## 18040 -0.009248741  
## 18041 0.045841890  
## 18044 -0.087518316  
## 18045 -0.190045804  
## 18046 0.107577599  
## 18051 0.134583748  
## 18052 -0.046608058  
## 18067 -0.048820899  
## 18071 -0.074434675  
## 18082 0.056499082  
## 18103 -0.220275322  
## 18107 0.138837455  
## 18115 -0.286010018  
## 18128 -0.165566478  
## 18129 0.051701298  
## 18132 -0.171088418  
## 18136 -0.127644854  
## 18142 0.273226035  
## 18161 -0.062322298  
## 18177 -0.366629111  
## 18180 -0.050722592  
## 18181 -0.146770735  
## 18182 -0.461096244  
## 18185 -0.386442368  
## 18188 -0.142855215  
##   
## with conditional variances for "sal.id"

#Calculate conﬁdence intervals for coeﬃcient estimate  
confint(model\_6)

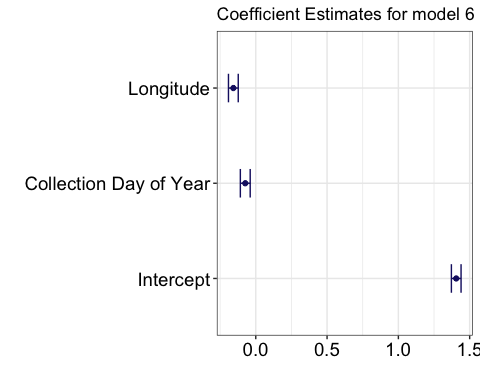
## Computing profile confidence intervals ...

## 2.5 % 97.5 %  
## .sig01 0.2071951 0.26175136  
## .sigma 0.1405957 0.17000163  
## (Intercept) 1.3718305 1.43968577  
## coll.doy\_standardized -0.1081468 -0.03906564  
## long\_standardized -0.1914520 -0.12286438

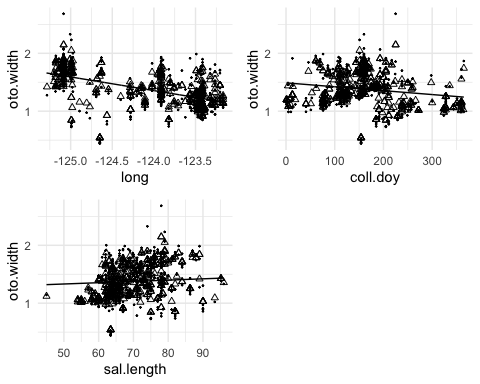
#Rearrange data for a plot of coefficients with standard errors  
#Make data frame for plot of coefficients  
confint\_fixed\_6 <- as.data.frame(confint(model\_6, parm = c("(Intercept)", "coll.doy\_standardized", "long\_standardized")))

## Computing profile confidence intervals ...

##Compute profile confidence intervals  
confint\_fixed\_6 <- as.data.frame(confint\_fixed\_6)  
confint\_fixed\_6$term <- rownames(confint\_fixed\_6)  
confint\_fixed\_6$estimate <- fixef(model\_6)  
  
##ggplot  
ggplot(confint\_fixed\_6) +   
 theme\_bw() +  
 geom\_point(aes(y = term, x = estimate), colour = "midnightblue") +  
 geom\_errorbarh(aes(y = term, xmin = `2.5 %`, xmax = `97.5 %`), colour = "midnightblue", height = 0.3) +  
 scale\_y\_discrete(labels = c("Intercept", "Collection Day of Year", "Longitude")) +  
 xlab("") +  
 ylab("") +   
 ggtitle("Coefficient Estimates for model 6") +  
 theme(axis.text = element\_text(size = 14, colour = "black"))

 #Section 8: Plotting a Model with the Data ##Plot model 11

#Create data frame for plot of model 11 predictions with longitude on the x axis  
oto.width\_vs\_long\_data\_11 <- data.frame(long = seq(-125.29, -123.16, 0.001), coll.doy\_standardized = median(herring\_data$coll.doy\_standardized), sal.length\_standardized = median(herring\_data$sal.length\_standardized))  
  
#Add standardized long data to data frame  
oto.width\_vs\_long\_data\_11$long\_standardized <- ((oto.width\_vs\_long\_data\_11$long - mean(oto.width\_vs\_long\_data\_11$long))/sd(oto.width\_vs\_long\_data\_11$long))  
  
#Generate model predictions at population level   
oto.width\_vs\_long\_data\_11$fit\_11\_pop <- predict(model\_11, newdata = oto.width\_vs\_long\_data\_11, re.form = ~0)  
  
#Generate model predictions at salmon id level  
herring\_data$fit\_11\_sal.id <- predict(model\_11)  
  
#Plot the model vs. longitude  
model\_11\_long <- ggplot(herring\_data) +  
 theme\_minimal() +   
 geom\_point(aes(long, oto.width), shape = 1, size = 0.2) +  
 geom\_line(aes(long, fit\_11\_pop), data = oto.width\_vs\_long\_data\_11) +   
 geom\_point(aes(long, fit\_11\_sal.id, group = sal.id), shape = 2)  
  
#Create data frame for plot of model 11 predictions with coll.doy on the x axis  
oto.width\_vs\_coll.doy\_data\_11 <- data.frame(coll.doy = seq(1, 365, 1), long\_standardized = median(herring\_data$long\_standardized), sal.length\_standardized = median(herring\_data$sal.length\_standardized))  
  
#Add standardized coll.doy data to data frame  
oto.width\_vs\_coll.doy\_data\_11$coll.doy\_standardized <- ((oto.width\_vs\_coll.doy\_data\_11$coll.doy - mean(oto.width\_vs\_coll.doy\_data\_11$coll.doy))/sd(oto.width\_vs\_coll.doy\_data\_11$coll.doy))  
  
#Generate model predictions at population level   
oto.width\_vs\_coll.doy\_data\_11$fit\_11\_pop <- predict(model\_11, newdata = oto.width\_vs\_coll.doy\_data\_11, re.form = ~0)  
  
#Generate model predictions at salmon id level  
herring\_data$fit\_11\_sal.id <- predict(model\_11)  
  
#Plot the model vs.coll.doy  
model\_11\_coll.doy <- ggplot(herring\_data) +  
 theme\_minimal() +   
 geom\_point(aes(coll.doy, oto.width), shape = 1, size = 0.2) +  
 geom\_line(aes(coll.doy, fit\_11\_pop), data = oto.width\_vs\_coll.doy\_data\_11) +   
 geom\_point(aes(coll.doy, fit\_11\_sal.id, group = sal.id), shape = 2)  
  
#Create data frame for plot of model 11 predictions with sal.length on the x axis  
oto.width\_vs\_sal.length\_data\_11 <- data.frame(sal.length = seq(45, 96, 0.5), long\_standardized = median(herring\_data$long\_standardized), coll.doy\_standardized = median(herring\_data$coll.doy\_standardized))  
  
#Add standardized sal.length data to data frame  
oto.width\_vs\_sal.length\_data\_11$sal.length\_standardized <- ((oto.width\_vs\_sal.length\_data\_11$sal.length - mean(oto.width\_vs\_sal.length\_data\_11$sal.length))/sd(oto.width\_vs\_sal.length\_data\_11$sal.length))  
  
#Generate model predictions at population level   
oto.width\_vs\_sal.length\_data\_11$fit\_11\_pop <- predict(model\_11, newdata = oto.width\_vs\_sal.length\_data\_11, re.form = ~0)  
  
#Plot the model vs. sal.length  
model\_11\_sal.length <- ggplot(herring\_data) +  
 theme\_minimal() +   
 geom\_point(aes(sal.length, oto.width), shape = 1, size = 0.2) +  
 geom\_line(aes(sal.length, fit\_11\_pop), data = oto.width\_vs\_sal.length\_data\_11) +   
 geom\_point(aes(sal.length, fit\_11\_sal.id, group = sal.id), shape = 2)  
  
grid.arrange(model\_11\_long, model\_11\_coll.doy, model\_11\_sal.length, nrow = 2)



#Section 9: Discussion (400 words) ##A) Assessment of how well the model seems to ﬁt the data - model checking - visualization of the model with the data - ##B) Qualitative interpretation of the model - interpretation of results in relation to your research questions and hypotheses - general interpretation of conﬁdence intervals or standard errors for coeﬃcient estimates - ##C) Limitations of the model(s) that you have ﬁt - Limitations - Possible options for improvement - Possible approaches for further data analysis

##D) Results into context - citie at least two relevant scientiﬁc papers

## References

Schindler, D. E., M. D. Scheuerell, J. W. Moore, S. M. Gende, T. B. Francis, and W. J. Palen. 2003. Pacific Salmon and the Ecology of Coastal Ecosystems. Frontiers in Ecology and the Environment 1:31.

Stevenson, D. K., and S. E. Campana. 1992. Otolith microstructure examination and analysis. Can. Spec. Publ. Fish. Aquat. Sci. 117: 126 p. 1. Page Otolith Microstructure Examination and Analysis.

Bates, D., Maechler, M., Bolker, B., Walker, S., 2015. Fitting Linear Mixed-Effects Models Using lme4. Journal of Statistical Software, 67 (1), 1-48.