

MED Gear Comparison Analysis

Prepared by: T.C. Wainwright, Newport Research Station, Northwest Fisheries Science Center, National Marine Fisheries Service

Updated: 08 Oct. 2018, 10:00, using R version 3.4.4 (2018-03-15)

NOTE: This is a work of the U.S. Government and is not subject to copyright protection in the United States. Foreign copyrights may apply.

NOTE: References throughout this document to trade names does not imply endorsement by the National Marine Fisheries Service, NOAA.

Introduction

This document presents full statistical results and describes the R code used to conduct the analysis in Wainwright et al. “*Effect of a mammal excluder device on catches of small pelagic fishes*”. Data and scripts for the analysis are available in the R package ‘GearComparisonAnalysis2018’ available on [GitHub](#).

NOTE: “R” is open source software freely available from the [R Project](#).

Part 1: Read and Summarize data

First, read in the data.

```
#data.dir <- '.' #Directory where the data files resides
MMEDdata <- read.csv(system.file('extdata', 'AllSppCounts&Lengths.csv',
                                package = "GearComparisonAnalysis2018"))
# print(summary(MMEDdata))
```

Fix some data problems. Create consistent haul numbers, filter out “experimental” hauls, and change Chinook and coho salmon “species” names to reflect age groups. Some species are renamed to make nicer plot labels, and months and times are recoded.

```
# Haul ID is last three characters of station code:
MMEDdata$Station <- as.character(MMEDdata$Station) #remove factor levels
MMEDdata$Haul <- with(MMEDdata, substr(Station, nchar(Station)-2, nchar(Station)))
# Excluder code (Y or N) is last character of station code:
MMEDdata$Excluder <- with(MMEDdata, substr(Station, nchar(Station), nchar(Station)))
# Filter out "experimental" hauls:
MMEDdata <- MMEDdata[-grep('*X$', MMEDdata$Haul), ]
# Recode MMED types to Standard short labels
### NOTE: work around database error where some records with Excluder code "N"
### have MMED code "Yes; Up"
MMEDdata$MMED <- as.character(MMEDdata$MMED) #remove factor levels
MMEDdata$MMED[MMEDdata$Excluder == "N"] <- "None"
MMEDdata$MMED[MMEDdata$Excluder == "Y" & grepl("Up", MMEDdata$MMED)] <- "Up"
MMEDdata$MMED[MMEDdata$Excluder == "Y" & grepl("Down", MMEDdata$MMED)] <- "Down"
# Fix a few species names, add age-classes for Chinook & coho
MMEDdata$Species <- toupper(as.character(MMEDdata$Species)) #all upper case
MMEDdata$Species[MMEDdata$Species %in% 'CALIFORNIA MARKET SQUID'] <- 'MARKET SQUID'
MMEDdata$Species[grepl('SMELT', MMEDdata$Species)] <- 'SMELT SPP.'
MMEDdata$SpecAge <- as.character(MMEDdata$SpecAge)
.index <- MMEDdata$SpecAge %in% 'CHINOOK SALMON'
.newnames <- paste('CHINOOK', MMEDdata$AgeGp[.index])
```

```

MMEDdata$SpecAge[.index] <- .newnames
.index <- MMEDdata$SpecAge %in% 'COHO SALMON'
.newnames <- paste('COHO', MMEDdata$AgeGp[.index])
MMEDdata$SpecAge[.index] <- .newnames
MMEDdata$SpecAge[MMEDdata$SpecAge %in%
  c('CHINOOK subadult/adult', 'CHINOOK mixed age juvenile')] <- 'CHINOOK subadult'
MMEDdata$SpecAge[MMEDdata$SpecAge %in% 'COHO subadult/adult'] <- 'COHO subadult'
MMEDdata$SpecAge <- factor(as.character(MMEDdata$SpecAge))
# print(summary(MMEDdata))

```

Divide the hauls into analytic “blocks” which are combinations of Date X Location. There are 12 blocks (“A” through “L”), with the following number of hauls in each:

- May 2011: block A : 10, B : 10
- July 2011: C : 4, D : 4, E : 10
- Jun-Jul 2014: F : 4, G : 4, H : 8
- July 2015: I : 8, J : 8, K : 8, L : 8

```

# sequential along the sorted haul numbers,
blocks <- as.factor(c(rep('A',10), rep('B',10), rep('C', 4), rep('D', 4),
  rep('E',10), rep('F', 4), rep('G', 4), rep('H', 8),
  rep('I', 8), rep('J', 8), rep('K', 8), rep('L', 8)))
sortHauls <- sort(unique(MMEDdata$Haul))
MMEDdata$Block <- blocks[match(MMEDdata$Haul, sortHauls)]
# print(with(MMEDdata, t(apply(as.character(Block), list(Haul,MMED), unique))))

```

Then, create a summary table of total catch by species and gear type (None==“without MED”, Down==“with downward MED”, Up==“with upward MED”), ...

```

tab1 <- with(MMEDdata, tapply(Number, list(SpecAge, MMED), sum, na.rm=T))
tab1[is.na(tab1)] <- 0 #Missing values are actually zero counts
tab1 <- cbind(tab1, Total=apply(tab1, 1, sum))
print(tab1[, c('Down', 'Up', 'None', 'Total')])

```

##	Down	Up	None	Total
## AMERICAN SHAD	6	0	6	12
## BLACK ROCKFISH	0	1	5	6
## BLUE SHARK	0	0	1	1
## CABEZON	0	1	0	1
## CHINOOK subadult	28	36	59	123
## CHINOOK subyearling	191	713	1092	1996
## CHINOOK yearling	90	223	364	677
## CHUM SALMON	54	0	173	227
## COHO subadult	41	25	61	127
## COHO yearling	38	24	194	256
## COMB JELLIES	0	0	0	0
## CUTTHROAT TROUT	0	0	4	4
## EGGYOLK JELLY	29	2	41	72
## FISH	0	2	0	2
## FLATFISHES	0	1	3	4
## HORMIPHORA CUCUMIS	0	0	0	0
## HYBRID STEELHEAD AND CUTTHROAT TROUT	0	0	1	1
## JACK MACKEREL	0	9	13	22
## LINGCOD	0	0	1	1
## LION'S MANE JELLY	0	0	3	3
## MARKET SQUID	2160	258	8285	10703

## MOON JELLY	10259	2	1377	11638
## NORTHERN ANCHOVY	2	200	3073	3275
## OCEAN SUNFISH	0	1	0	1
## PACIFIC CHUB MACKEREL	0	3	2	5
## PACIFIC HERRING	1755	58	563	2376
## PACIFIC POMPAÑO	7	0	5	12
## PACIFIC SANDDAB	0	0	4	4
## PACIFIC SANDFISH	0	0	1	1
## PACIFIC SARDINE	0	45	75	120
## PACIFIC SPINY DOGFISH	0	3	5	8
## PACIFIC STAGHORN SCULPIN	0	1	0	1
## PACIFIC TOMCOD	0	15	22	37
## PINK SALMON	0	1	1	2
## REX SOLE	0	0	1	1
## SALPS	0	0	0	0
## SEA NETTLE	21	1285	1422	2728
## SHINER PERCH	1	0	0	1
## SMELT SPP.	120	464	2221	2805
## SOCKEYE SALMON	0	0	22	22
## STARRY FLOUNDER	0	3	2	5
## STEELHEAD	1	3	6	10
## THRESHER SHARK	0	0	1	1
## TOPE	0	0	7	7
## WATER JELLY	53291	528	71472	125291
## WESTERN RIVER LAMPREY	6	2	16	24
## WOLF-EEL	6	2	29	37
## YELLOWTAIL ROCKFISH	0	0	1	1

... and, do the same for frequency of catch.

```
.tmp.all <- with(MMEDdata, table(SpecAge, Haul))
.tmp.std <- with(MMEDdata[MMEDdata$MMED=='None', ], table(SpecAge, Haul))
.tmp.up <- with(MMEDdata[MMEDdata$MMED=='Up', ], table(SpecAge, Haul))
.tmp.dwn <- with(MMEDdata[MMEDdata$MMED=='Down', ], table(SpecAge, Haul))
tab2 <- cbind(apply(.tmp.dwn>0, 1, sum),      # Num. occurrences in STD
              apply(.tmp.up>0, 1, sum),      # . . .
              apply(.tmp.std>0, 1, sum),     # . . .
              apply(.tmp.all>0, 1, sum))     # Total Num. occurrences
colnames(tab2) <- c('Down', 'Up', 'None', 'Total')
print(tab2)
```

##	Down	Up	None	Total
## AMERICAN SHAD	2	0	3	5
## BLACK ROCKFISH	0	1	2	3
## BLUE SHARK	0	0	1	1
## CABEZON	0	1	0	1
## CHINOOK subadult	13	19	27	59
## CHINOOK subyearling	14	18	31	63
## CHINOOK yearling	13	16	37	66
## CHUM SALMON	12	0	21	33
## COHO subadult	14	11	25	50
## COHO yearling	13	15	32	60
## COMB JELLIES	0	2	2	4
## CUTTHROAT TROUT	0	0	3	3
## EGGYOLK JELLY	15	2	8	25

## FISH	0	1	0	1
## FLATFISHES	0	1	4	5
## HORMIPHORA CUCUMIS	0	0	1	1
## HYBRID STEELHEAD AND CUTTHROAT TROUT	0	0	1	1
## JACK MACKEREL	0	3	2	5
## LINGCOD	0	0	1	1
## LION'S MANE JELLY	0	0	1	1
## MARKET SQUID	16	16	36	68
## MOON JELLY	16	2	15	33
## NORTHERN ANCHOVY	2	8	10	20
## OCEAN SUNFISH	0	1	0	1
## PACIFIC CHUB MACKEREL	0	2	2	4
## PACIFIC HERRING	4	2	10	16
## PACIFIC POMPAÑO	6	0	1	7
## PACIFIC SANDDAB	0	0	3	3
## PACIFIC SANDFISH	0	0	1	1
## PACIFIC SARDINE	0	3	4	7
## PACIFIC SPINY DOGFISH	0	3	5	8
## PACIFIC STAGHORN SCULPIN	0	1	0	1
## PACIFIC TOMCOD	0	4	5	9
## PINK SALMON	0	1	1	2
## REX SOLE	0	0	1	1
## SALPS	0	6	5	11
## SEA NETTLE	6	17	26	49
## SHINER PERCH	1	0	0	1
## SMELT SPP.	7	2	9	18
## SOCKEYE SALMON	0	0	7	7
## STARRY FLOUNDER	0	2	2	4
## STEELHEAD	1	3	3	7
## THRESHER SHARK	0	0	1	1
## TOPE	0	0	3	3
## WATER JELLY	16	17	29	62
## WESTERN RIVER LAMPREY	5	2	9	16
## WOLF-EEL	4	2	14	20
## YELLOWTAIL ROCKFISH	0	0	1	1

Also, look at species by cruise to see which are “regularly caught.” The final two columns give the number of cruises with nonzero catch (“Ngt0”) and catch > 1 (“Ngt1”) for each species.

```
tab3 <- with(MMEDdata, tapply(Number, list(SpecAge, Cruise), sum, na.rm=T))
tab3[is.na(tab3)] <- 0 #Missing values are actually zero counts
ngt0 <- apply(tab3>0, 1, sum)
ngt1 <- apply(tab3>1, 1, sum)
tab3 <- cbind(tab3, Ngt0=ngt0, Ngt1=ngt1)
print(tab3)
```

##	41	43	50	53	Ngt0	Ngt1
## AMERICAN SHAD	0	1	0	11	2	1
## BLACK ROCKFISH	0	1	1	4	3	1
## BLUE SHARK	0	1	0	0	1	0
## CABEZON	0	0	1	0	1	0
## CHINOOK subadult	21	42	11	49	4	4
## CHINOOK subyearling	8	1451	53	484	4	4
## CHINOOK yearling	472	9	25	171	4	4
## CHUM SALMON	19	2	1	205	4	3

## COHO subadult	6	31	34	56	4	4
## COHO yearling	124	16	20	96	4	4
## COMB JELLIES	0	0	0	0	0	0
## CUTTHROAT TROUT	3	1	0	0	2	1
## EGGYOLK JELLY	1	0	1	70	3	1
## FISH	2	0	0	0	1	1
## FLATFISHES	1	0	0	3	2	1
## HORMIPHORA CUCUMIS	0	0	0	0	0	0
## HYBRID STEELHEAD AND CUTTHROAT TROUT	0	0	0	1	1	0
## JACK MACKEREL	0	22	0	0	1	1
## LINGCOD	1	0	0	0	1	0
## LION'S MANE JELLY	0	0	3	0	1	1
## MARKET SQUID	126	899	65	9613	4	4
## MOON JELLY	0	1	1	11636	3	1
## NORTHERN ANCHOVY	0	2827	445	3	3	3
## OCEAN SUNFISH	0	1	0	0	1	0
## PACIFIC CHUB MACKEREL	4	0	1	0	2	1
## PACIFIC HERRING	2	2	608	1764	4	4
## PACIFIC POMPAÑO	0	0	0	12	1	1
## PACIFIC SANDDAB	0	0	0	4	1	1
## PACIFIC SANDFISH	0	0	0	1	1	0
## PACIFIC SARDINE	111	8	0	1	3	2
## PACIFIC SPINY DOGFISH	1	2	2	3	4	3
## PACIFIC STAGHORN SCULPIN	0	0	1	0	1	0
## PACIFIC TOMCOD	0	0	37	0	1	1
## PINK SALMON	0	2	0	0	1	1
## REX SOLE	0	0	0	1	1	0
## SALPS	0	0	0	0	0	0
## SEA NETTLE	0	1570	1111	47	3	3
## SHINER PERCH	0	0	0	1	1	0
## SMELT SPP.	0	0	2582	223	2	2
## SOCKEYE SALMON	22	0	0	0	1	1
## STARRY FLOUNDER	0	0	3	2	2	2
## STEELHEAD	3	2	0	5	3	3
## THRESHER SHARK	0	0	0	1	1	0
## TOPE	0	7	0	0	1	1
## WATER JELLY	0	377	507	124407	3	3
## WESTERN RIVER LAMPREY	0	2	2	20	3	3
## WOLF-EEL	0	6	1	30	3	2
## YELLOWTAIL ROCKFISH	1	0	0	0	1	0

Examining these tables, for analysis we select the species that have total catch of at least 100 and occurred more than once in at least 3 cruises.

```
sel.spec <- rownames(tab1)[tab1[, "Total"] >= 100]
sel.spec <- sel.spec[sel.spec %in% rownames(tab3[tab3[, "Ngt1"] >= 3, ])]
print(sel.spec)
```

```
## [1] "CHINOOK subadult"      "CHINOOK subyearling" "CHINOOK yearling"
## [4] "CHUM SALMON"          "COHO subadult"      "COHO yearling"
## [7] "MARKET SQUID"         "NORTHERN ANCHOVY"   "PACIFIC HERRING"
## [10] "SEA NETTLE"           "WATER JELLY"
```

Part2: Summary plots - CPUE by time

To estimate CPUE for each species in each haul, we need to convert the raw data that has counts by size into total counts for each species in each haul, then we need to summarize effort.

Generating the total catch summary is just a tabulation of numbers by size:

```
# Total number by Haul (rows) and Species
MMEDcnt <- with(MMEDdata, tapply(Number, list(Haul, SpecAge), FUN=sum, simplify=T))
MMEDcnt[is.na(MMEDcnt)] <- 0
MMEDcnt <- as.data.frame(MMEDcnt)
## print(summary(MMEDcnt)) ### DEBUG ###
```

Then, build a parallel structure of the haul data. This includes location, date, time, effort (as distance towed), and gear information.

```
MMEDhauls <- data.frame(HaulID = rownames(MMEDcnt))
MMEDhauls$Station <- with(MMEDdata, tapply(as.character(Station), list(Haul), FUN=unique))
MMEDhauls$Date <- with(MMEDdata, tapply(as.character(Date), list(Haul), FUN=unique))
MMEDhauls$Effort <- with(MMEDdata, tapply(Distance, list(Haul), FUN=unique))
MMEDhauls$Gear <- factor(with(MMEDdata, tapply(as.character(MMED), list(Haul), FUN=unique)),
                        levels=c('None', 'Up', 'Down'))
MMEDhauls$Block <- with(MMEDdata, tapply(as.character(Block), list(Haul), FUN=unique))
```

Finally, generate CPUE plots for selected species.

For this, we create an artificial “time” scale for plotting, creating equal intervals within blocks. Then, CPUE is computed as counts for each species divided by effort.

```
MMEDhauls$PlotTime <- match(MMEDhauls$Block, LETTERS[1:13]) - 1 +
  c((1:10)/11, (1:10)/11, (1:4)/5, (1:4)/5, (1:10)/11, (1:4)/5, (1:4)/5, (1:8)/9,
    (1:8)/9, (1:8)/9, (1:4)/5, (1:4)/5, (1:8)/9)
## print(summary(MMEDhauls)) ### DEBUG ###
MMEDcpue=sweep(MMEDcnt, 1, MMEDhauls$Effort, '/')
## print(summary(MMEDcpue)) ### DEBUG ###
```

Before running the analysis, set up some plotting configurations:

```
bw <- FALSE          #Flag for black-and-white figures
BLACK <- 'black'
BLUE <- if(bw) 'black' else 'blue'  #color code for blue
RED <- if(bw) 'black' else 'red'    #color code for red
lndscp <- FALSE      #flag for landscape figures
plcol <- if(lndscp) 3 else 2      #number of columns for multi-plots
```

A special plotting function is created, so it can be re-used later.

```
cpue.plot <- function(t, y, dot.col=1, log.zero=FALSE, ...) {
  minpos <- 0
  if (log.zero) {
    minpos <- min(y[y>0]) #minimum positive value
    if(any(y<=0)) {
      y[y<=0] <- minpos/2 # recode
    } # if(any...)
  } # if(log.zero)
  plot(t, y, log=ifelse(log.zero,"y",""), type='p',
       axes=F, ...)
  # label blocks at midpoint:
```

```

blocks <- seq(round(min(t)), round(max(t)))
axis(side=1, at=blocks, labels=NA) # Ticks at day boundaries
abline(v=blocks, col='blue')
axis(side=1, at=blocks[-1]-0.5,
      labels=sort(unique(as.character(MMEDhauls$Block))),
      tick=FALSE, cex.axis=0.8)
if (log.zero && any(y<minpos)) {
  tck<- axisTicks(range(y),log=FALSE) #default tick locations
  axis(side=2, at=c(minpos/2,tck), labels=c(0,tck))
} else {
  axis(side=2)
} # if (log.zero && ...
box()
} # cpue.plot()

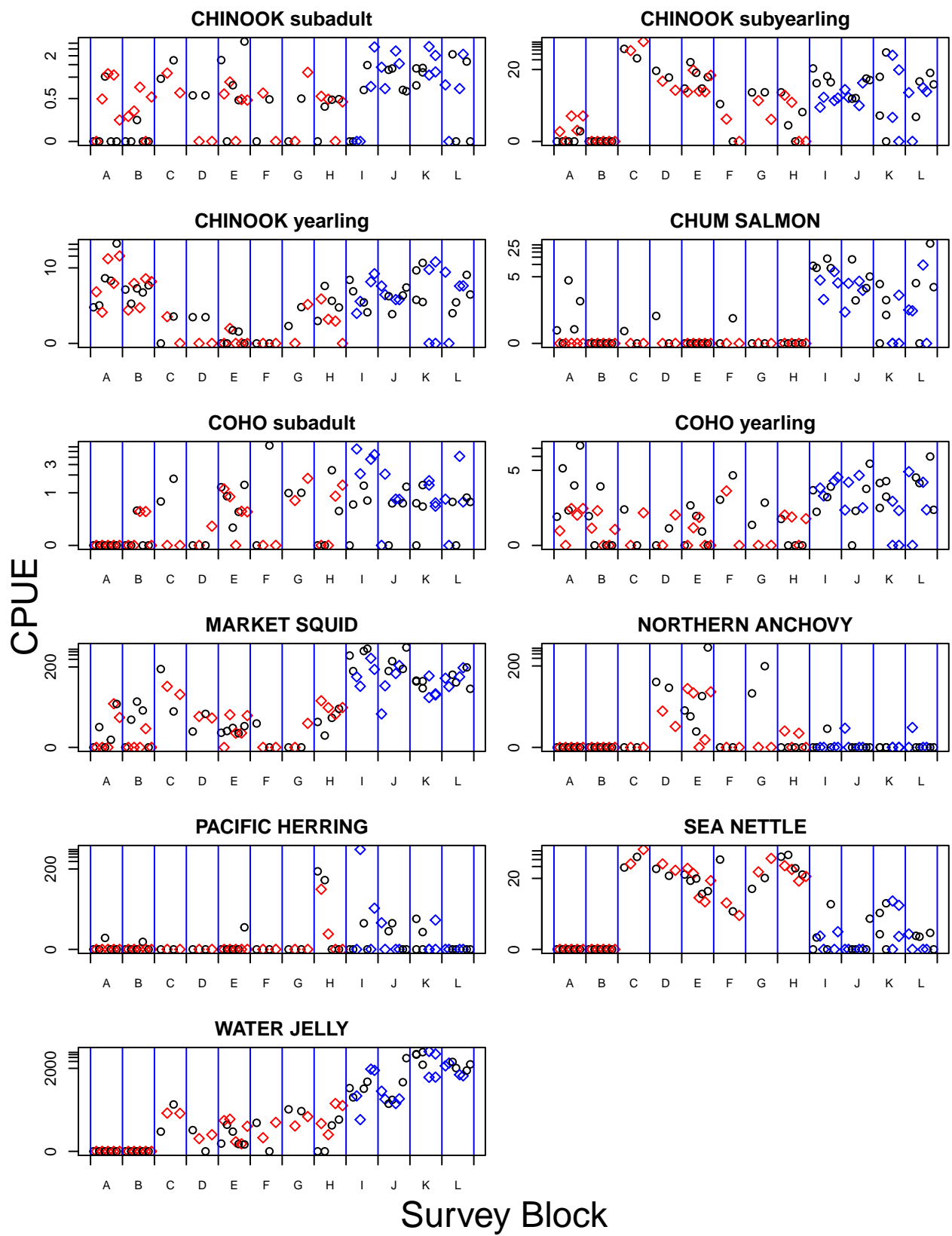
```

Generate the individual plots:

```

par(mfrow=c(ceiling(length(sel.spec)/plcol), plcol), omi=c(0.5,0.5,0,0), mar=c(3,2,2,1))
for (sp in sel.spec) {
  cpue.plot(MMEDhauls$PlotTime, MMEDcpue[[sp]], log.zero=TRUE,
            col=c(BLACK,RED,BLUE)[as.numeric(MMEDhauls$Gear)],
            pch=1+4*(MMEDhauls$Gear!='None'), xlab="", ylab="", main=sp)
} # for (sp)
mtext('Survey Block', side=1, outer=TRUE, cex=1.5)
mtext('CPUE', side=2, outer=TRUE, cex=1.5)

```



Part 3: Overall Catch Ratio Statistics

In preliminary analyses, we considered a number of methods, including:

- *CPUE ratio estimate*. Ratio of mean CPUEs (Wilderbuer et al. 1998, North. Am. J. Fish. Manage. 18:11-18) using bootstrapped quantiles rather than a normal approximation.
- *Paired sample differences*. This method uses paired t-test statistics with a $\log(x+\text{offset})$ transform on CPUE.
- *Nonparametric paired differences*. This method computes the median catch ratio and approximate binomial quantiles based on eq. 10.3 in Efron (1982, “*The Jackknife, the Bootstrap, and Other Resampling Plans*”) which finds the nearest observation toward the tail from the nominal quantile.
- *GLM analysis of deviance*. This method computes a GLM blocked ANODEV model of `Catch ~ Block + Gear + offset()` with either a Poisson or a negative binomial distribution, where the offset is the effort (km towed) for each sample. (Because the model uses log links, including effort as an offset is similar to using a linear model $\log(\text{CPUE}) \sim \text{Block} + \text{Gear}$.)

After reviewing the methods, for the final analysis, we used only the GLM method.

3.1 The analysis

To make the coding easier, we define standard names for the gear types, and set the rounding parameters for easy-to-read results tables.

```
assign('gears', c('None', 'Up', 'Down'), 1)
options(scipen=3) # Favor non-scientific notation
```

Then, create a function to compute the GLM estimates. The *R* `glm()` method does not support negative binomial distributions, so we use the MASS library (Venables & Ripley 2002, “*Modern Applied Statistics with S*, 4th ed.”). The `glm.nb()` function is used to estimate the θ parameter; if that estimation fails, it reverts to the value specified in “init.theta”. Then (because `glm.nb()` fails for some species) `glm()` is used with the estimated θ to get the final statistics. This means that the distribution of the final catch ratio estimate does not include error in the estimation of θ .

The GLM model is on a log scale, so exponential transformations are needed to get the catch-ratio estimate and quantiles. Predictions are returned in `$Pred` for later plotting.

NOTE that we want the ratio MED/STD, so we use the negative of the log-scale `GearStd` coefficient.

```
GLMAnoDevEst <- function(dat, qprobs=0.5, nb=TRUE, init.theta=1,
                        diag.plt=FALSE, plt.lab='') {
  rslt <- matrix(NA, ncol=length(gears)-1, nrow=1+length(qprobs),
                dimnames=list(c("Mean", as.character(qprobs)), gears[2:length(gears)]))
  library(MASS)
  fit.data <- data.frame(Count=as.vector(dat[,1]),
                        Date=as.factor(dat$Date),
                        Station=as.factor(dat$Station),
                        Block=as.factor(dat$Block),
                        Gear=as.factor(dat$Gear),
                        Offset=as.vector(dat$Effort))
  if (nb) { # Negative Binomial fit
    cat('\n\tInitial fit to estimate theta\n')
    fit.init <- try(glm.nb(Count ~ Block + Gear + offset(log(Offset)),
                        data=fit.data, init.theta=init.theta,
                        control=list(epsilon=1e-03, maxit=500, trace=0)))
    if (inherits(fit.init, 'try-error')) {
```

```

print(fit.init)
warning("glm.nb failed to estimate theta; using default value")
theta.init=init.theta # initial estimate for herring from theta.ml
} else {
cat('Estimated theta: ', fit.init$theta, ', SE: ', fit.init$SE.theta, '\n')
if(is.finite(fit.init$SE.theta)) {
theta.init <- fit.init$theta
} else {
warning("glm.nb failed to estimate theta; using default value")
theta.init <- init.theta
}
} # if 'try-error'
cat('\n\tFinal fit with theta = ', theta.init, '\n')
fit.fin <- glm(Count ~ Block + Gear + offset(log(Offset)),
data=fit.data, family=negative.binomial(theta.init),
control=list(epsilon=1e-08, maxit=500, trace=FALSE))
} else { # Poisson fit
fit.fin <- glm(Count ~ Block + Gear + offset(log(Offset)),
data=fit.data, family=poisson,
control=list(epsilon=1e-08, maxit=500, trace=FALSE))
} # if (nb)
cat("\nFIT STATISTICS:")
# print(summary(fit.fin))
fit.anova <- anova(fit.fin, test="Chisq")
print(fit.anova)
geareffects <- paste('Gear',gears[2:length(gears)], sep='')
.lmn <- summary(fit.fin)$coefficients[geareffects, "Estimate"]
.lsd <- summary(fit.fin)$coefficients[geareffects, "Std. Error"]
.df <- fit.fin$df.residual
for (g in 1:length(.lmn)) {
if (.lsd[g] > 1000) { # Estimate blew up, just use the mean value
.mn <- exp(.lmn[g])
.qnt <- rep(NA, length(qprobs))
} else {
.mn <- exp(.lmn[g] + .lsd[g]^2 / 2)
.qnt <- exp(qt(qprobs, .df)*.lsd[g]+.lmn[g])
} # if (.lsd[g])
rslt[, g] <- c(.mn, .qnt)
} # for (g)
predCatch <- predict(fit.fin, type="response", se.fit=TRUE)
if (diag.plt) {
rs <- resid(fit.fin, type="deviance")
op <- par(omi=c(0,0,0.25,0), mfrow=c(1,2), mar=c(4,4,1,1))
plot(predCatch$fit, rs, xlab="Prediction", ylab="Deviance Resids")
qqnorm(rs, ylab="Deviance Resids")
qqline(rs)
mtext(paste(plt.lab, ifelse(nb, "Neg. Binomial", "Poisson"), sep=' - '),
side=3, outer=TRUE)
par(op)
} # if (diat.plt)
return(list(Smry=rslt, Pred=predCatch))
} # GLMAnoDevEst()

```

Before running the analysis, create two summary lists: one for the means and quantiles of all methods by

species, and one for storing GLM model results for plotting. The first list contains one element for each species, holding a matrix of summary results for estimates of the ratio A by each of the various estimation methods. Rows are the methods, columns are a six-number summary: Mean, Median, and quantiles (0.05, 0.25, 0.75, 0.95).

The routine then loops through the species, computing first the GLM Poisson catch ratio estimates, then the GLM negative-binomial estimates. The Poisson results are used to get a crude initial estimate of the neg-bin Θ parameter, which is used to initiate the refined estimate via `glm.nb()`. If the refined estimate fails, the initial estimate of Θ is used instead.

```
stat.sum <- list() # structure for storing summary results
GLM.pred <- list() # structure for storing GLM predictions
for (sp in sel.spec) {
  cat('\n***** ', sp, ' *****\n')
  .sumtbl <- array(NA, dim=c(2, 6, length(gears)-1),
    dimnames=list(c('GLM.Po', 'GLM.nb'),
      c('Mean', 'q0.05', 'q0.25', 'Median', 'q0.75', 'q0.95'),
      gears[2:length(gears)]))
  .qprobs <- c(0.05, 0.25, 0.50, 0.75, 0.95)
  cat("\n*** METHOD 1: GLM ANODEV, Poisson ***\n")
  .est <- GLMAnoDevEst(cbind(MMEDcnt[[sp]], MMEDhauls), .qprobs,
    nb=FALSE, diag.plt=TRUE, plt.lab=sp)
  print(.est$Smry)
  .sumtbl['GLM.Po', , ] <- .est$Smry

  cat("\n*** METHOD 2: GLM ANODEV, negative binomial ***\n")
  init.theta <- theta.ml(y=MMEDcnt[[sp]], mu=.est$Pred$fit,
    n=length(MMEDcnt[[sp]]), limit=100, trace=FALSE)
  cat('Initial Theta: ', init.theta, '\n')
  .est <- GLMAnoDevEst(cbind(MMEDcnt[[sp]], MMEDhauls), .qprobs,
    nb=TRUE, init.theta=init.theta, diag.plt=TRUE, plt.lab=sp)
  print(.est$Smry)
  .sumtbl['GLM.nb', , ] <- .est$Smry

  # Add species to summary lists
  GLM.pred[[sp]] <- .est$Pred
  .sumtbl <- round(.sumtbl, 3) # round statistical results
  stat.sum[[sp]] <- .sumtbl
} # for(sp)
```

```
##
## ***** CHINOOK subadult *****
##
## *** METHOD 1: GLM ANODEV, Poisson ***
##
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: Count
##
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev    Pr(>Chi)
```

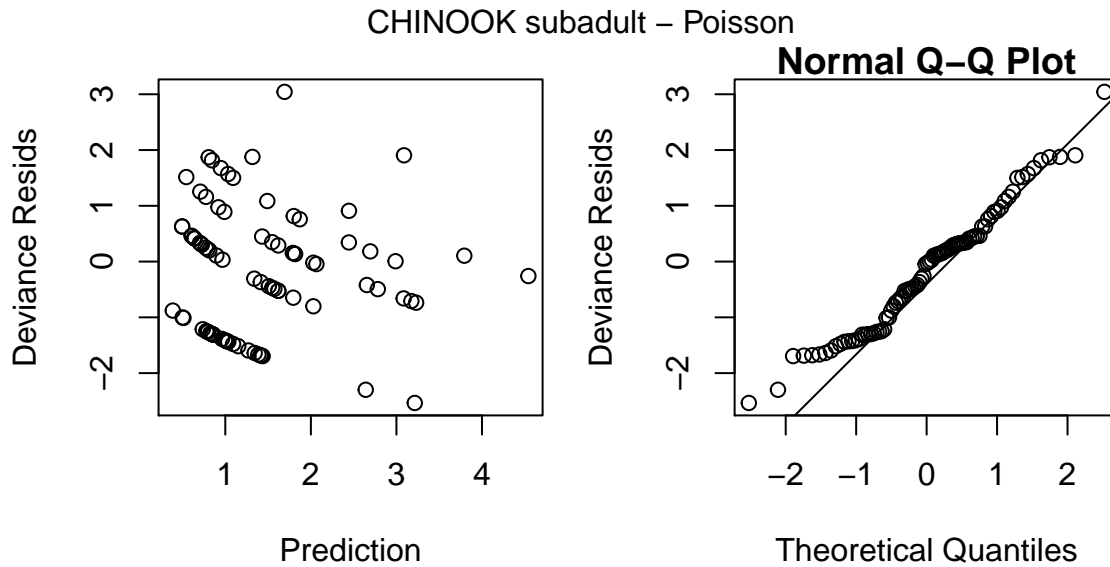
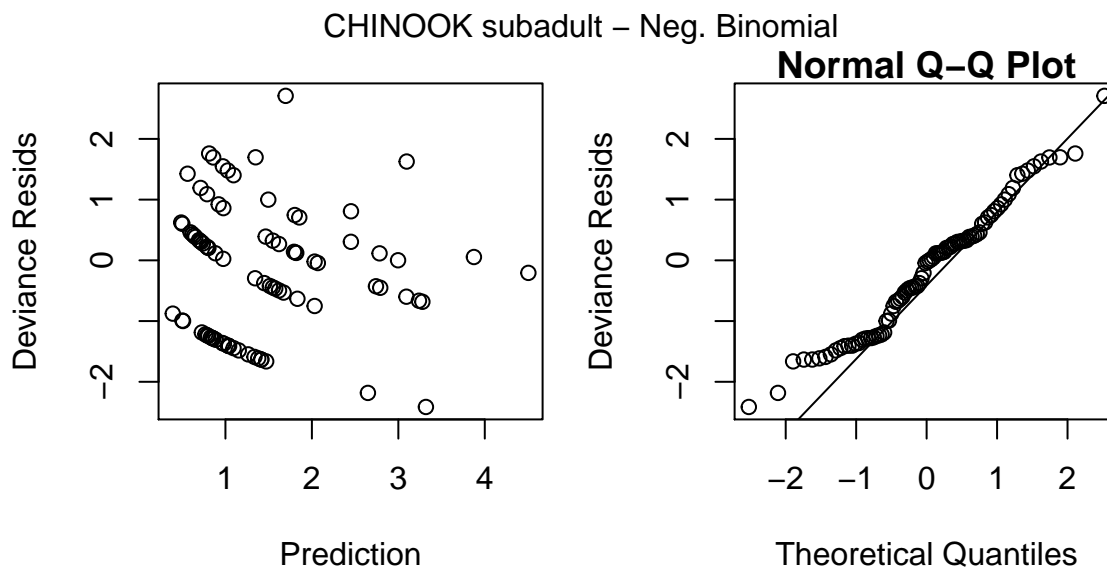


Figure 1:

```
## NULL                85      157.50
## Block 11    47.887      74      109.61 0.000001496 ***
## Gear  2      1.179      72      108.43      0.5545
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##           Up      Down
## Mean 1.0303668 1.4245452
## 0.05 0.6805217 0.8441632
## 0.25 0.8564966 1.1232247
## 0.5  1.0028455 1.3662921
## 0.75 1.1742010 1.6619597
## 0.95 1.4778356 2.2113664
##
## *** METHOD 2: GLM ANODEV, negative binomial ***
## Initial Theta: 11.84953
##
## Initial fit to estimate theta
## Estimated theta: 11.4745 , SE: 15.34951
##
## Final fit with theta = 11.4745
##
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: Negative Binomial(11.4745), link: log
##
## Response: Count
##
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                85      141.391
```



```
## Block 11    42.336         74    99.055 0.0003957 ***
## Gear   2     1.051         72    98.004 0.6575545
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##           Up      Down
## Mean 1.0744294 1.4517892
## 0.05 0.6471796 0.7688163
## 0.25 0.8540238 1.0819175
## 0.5  1.0329542 1.3676013
## 0.75 1.2493731 1.7287207
## 0.95 1.6486834 2.4327441
##
## ***** CHINOOK subyearling *****
##
## *** METHOD 1: GLM ANODEV, Poisson ***
##
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: Count
##
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev   Pr(>Chi)
## NULL              85      5379.7
## Block 11    4427.6      74      952.1 < 2.2e-16 ***
## Gear   2      22.5      72      929.5 0.00001275 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##           Up      Down
```

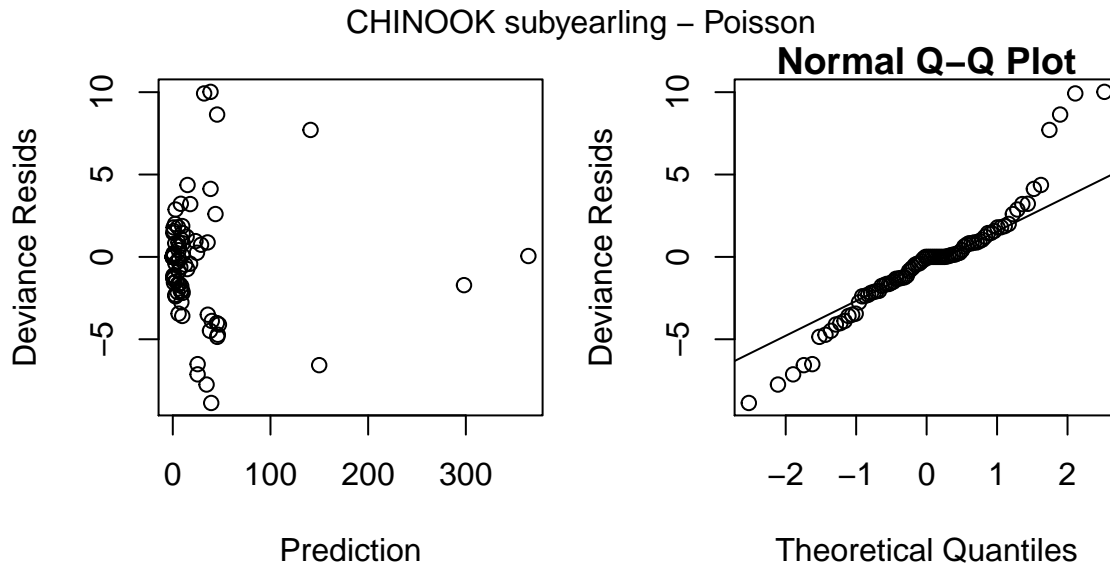
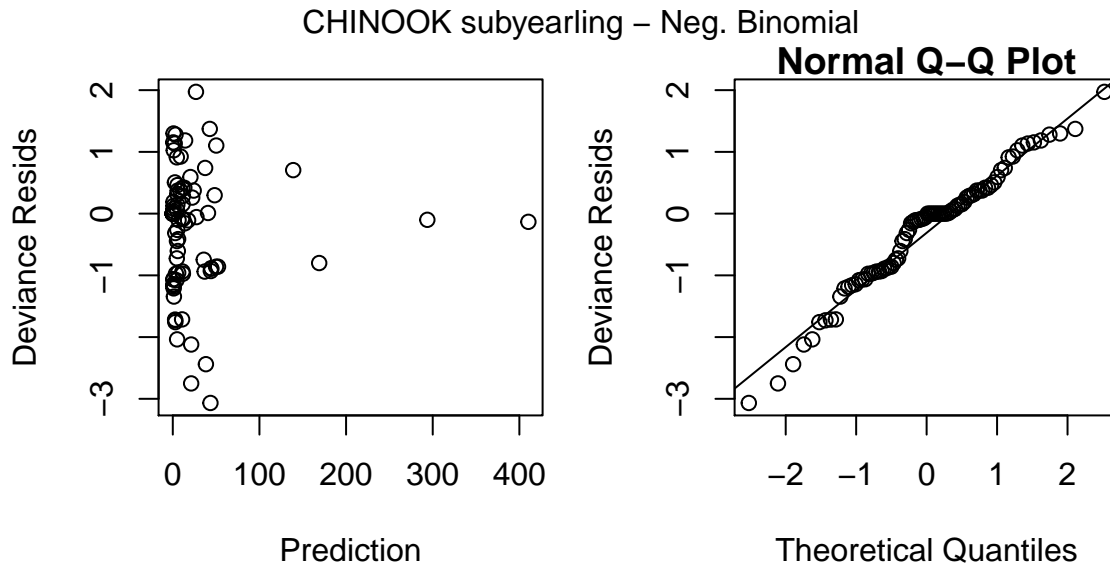


Figure 3:

```
## Mean 0.9848031 0.6494879
## 0.05 0.9025556 0.5537613
## 0.25 0.9497254 0.6071303
## 0.5 0.9834959 0.6466797
## 0.75 1.0184673 0.6888055
## 0.95 1.0716949 0.7551895
##
## *** METHOD 2: GLM ANODEV, negative binomial ***
## Initial Theta: 1.310679
##
## Initial fit to estimate theta
## Estimated theta: 1.340598 , SE: 0.2734801
##
## Final fit with theta = 1.340598
##
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: Negative Binomial(1.3406), link: log
##
## Response: Count
##
## Terms added sequentially (first to last)
##
##      Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                85      357.53
## Block 11  267.174      74      90.36 < 2e-16 ***
## Gear  2    4.698      72      85.66 0.06016 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##           Up      Down
## Mean 0.8942809 0.5101777
```



```
## 0.05 0.5381177 0.2964244
## 0.25 0.7104730 0.3983538
## 0.5 0.8596332 0.4878714
## 0.75 1.0401087 0.5975053
## 0.95 1.3732482 0.8029653
##
## ***** CHINOOK yearling *****
##
## *** METHOD 1: GLM ANODEV, Poisson ***
##
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: Count
##
## Terms added sequentially (first to last)
##
##      Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                85    1456.63
## Block 11      766.09      74      690.54 < 2e-16 ***
## Gear   2         6.53      72      684.01 0.03822 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##           Up      Down
## Mean 0.8063947 1.1275647
## 0.05 0.6918107 0.8631587
## 0.25 0.7558489 1.0043845
## 0.5 0.8031664 1.1143906
## 0.75 0.8534461 1.2364452
## 0.95 0.9324461 1.4387464
```

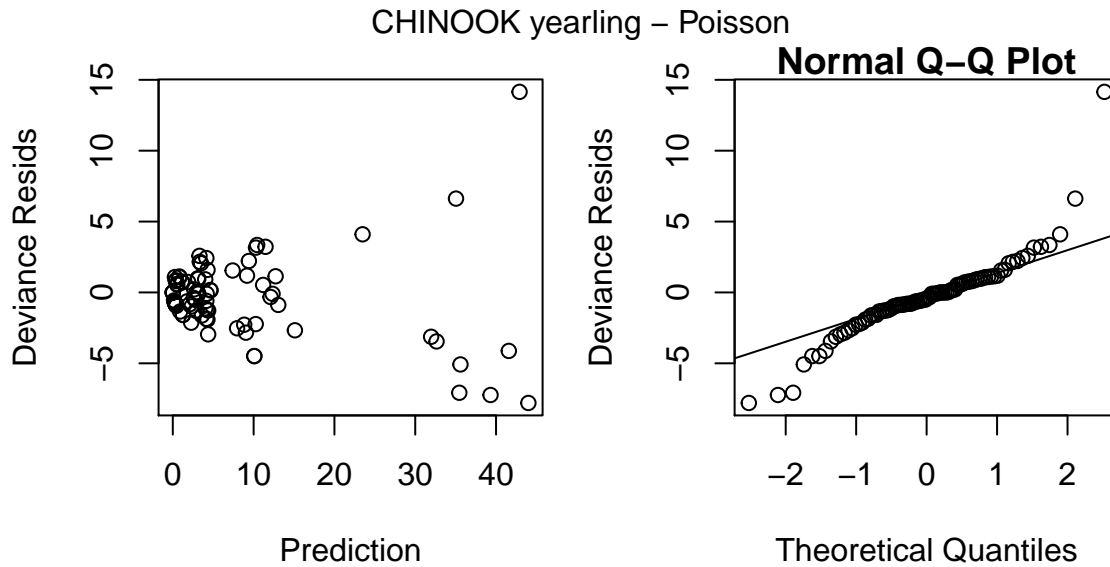


Figure 5:

```
##
## *** METHOD 2: GLM ANODEV, negative binomial ***
## Initial Theta: 1.5042
##
## Initial fit to estimate theta
## Estimated theta: 1.503116 , SE: 0.3096866
##
## Final fit with theta = 1.503116
##
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: Negative Binomial(1.5031), link: log
##
## Response: Count
##
## Terms added sequentially (first to last)
##
##      Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                        85    219.731
## Block 11  138.594         74     81.138  <2e-16 ***
## Gear    2    1.083         72     80.055   0.5341
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##           Up      Down
## Mean 0.7867095 1.2389568
## 0.05 0.4741009 0.7050806
## 0.25 0.6254745 0.9574429
## 0.5  0.7563934 1.1809969
## 0.75 0.9147153 1.4567488
## 0.95 1.2067706 1.9781479
##
```

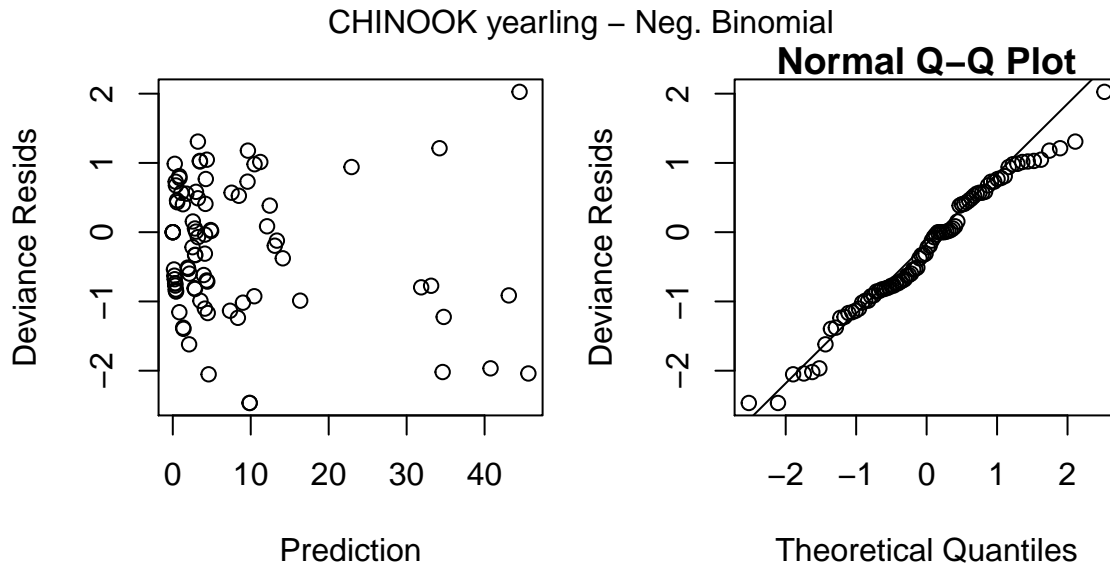
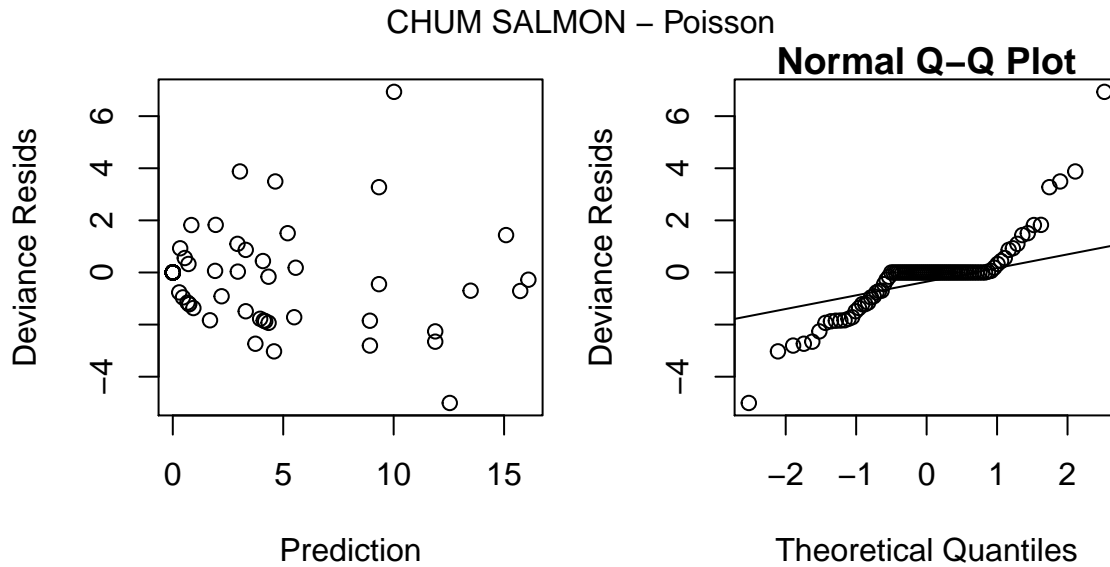



Figure 6:

```
## ***** CHUM SALMON *****
##
## *** METHOD 1: GLM ANODEV, Poisson ***
## Warning: glm.fit: fitted rates numerically 0 occurred
##
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: Count
##
## Terms added sequentially (first to last)
##
##      Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                        85      882.42
## Block 11      613.58          74      268.84 < 2.2e-16 ***
## Gear   2       71.07          72      197.77 3.696e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##           Up      Down
## Mean 8.724456e-09 0.3905847
## 0.05          NA 0.2961159
## 0.25          NA 0.3463782
## 0.5           NA 0.3857016
## 0.75          NA 0.4294894
## 0.95          NA 0.5023904
##
## *** METHOD 2: GLM ANODEV, negative binomial ***
## Initial Theta: 1.259582
##
```



```
## Initial fit to estimate theta
## Estimated theta: 1.259586 , SE: 0.3875694
##
## Final fit with theta = 1.259586
##
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: Negative Binomial(1.2596), link: log
##
## Response: Count
##
## Terms added sequentially (first to last)
##
##      Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                85    292.744
## Block 11    221.763      74    70.981 < 2.2e-16 ***
## Gear  2     25.384      72    45.597 1.216e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##              Up      Down
## Mean 2.679148e-09 0.3863633
## 0.05          NA 0.2359931
## 0.25          NA 0.3092175
## 0.5           NA 0.3721880
## 0.75          NA 0.4479821
## 0.95          NA 0.5869828
##
## ***** COHO subadult *****
##
## *** METHOD 1: GLM ANODEV, Poisson ***
##
```

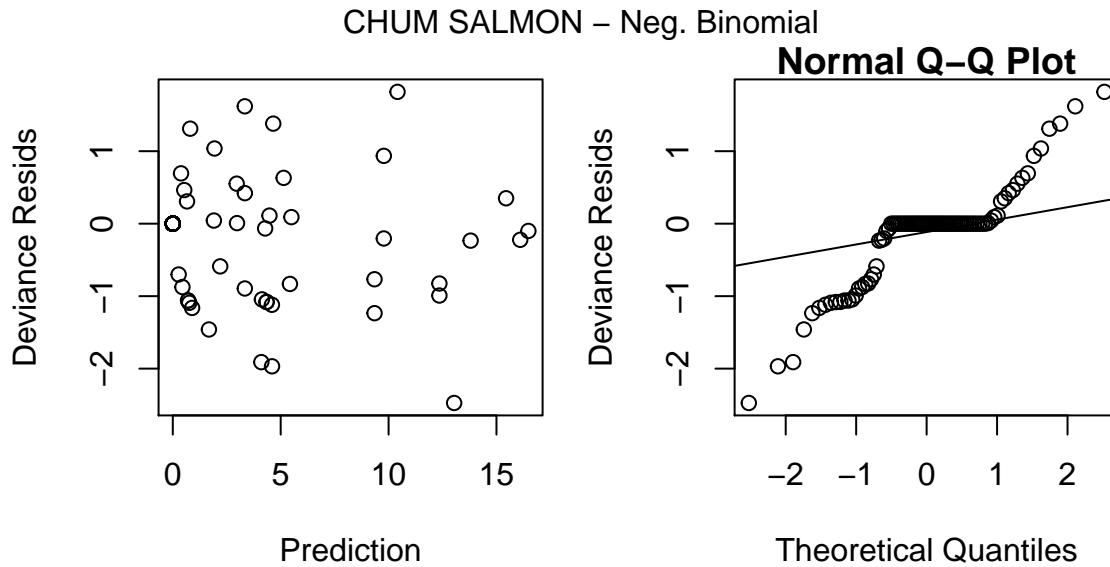


Figure 8:

```
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: Count
##
## Terms added sequentially (first to last)
##
##      Df Deviance Resid. Df Resid. Dev   Pr(>Chi)
## NULL                85      251.80
## Block 11  118.686      74      133.11 < 2.2e-16 ***
## Gear   2    18.565      72      114.55 0.00009303 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##           Up      Down
## Mean 0.6194021 3.043432
## 0.05 0.3968357 1.758135
## 0.25 0.5073904 2.369548
## 0.5  0.6005476 2.907803
## 0.75 0.7108086 3.568325
## 0.95 0.9088332 4.809254
##
## *** METHOD 2: GLM ANODEV, negative binomial ***
## Initial Theta:  3.830574
##
## Initial fit to estimate theta
## Estimated theta:  3.686582 , SE:  2.051033
##
## Final fit with theta =  3.686582
##
## FIT STATISTICS:Analysis of Deviance Table
```

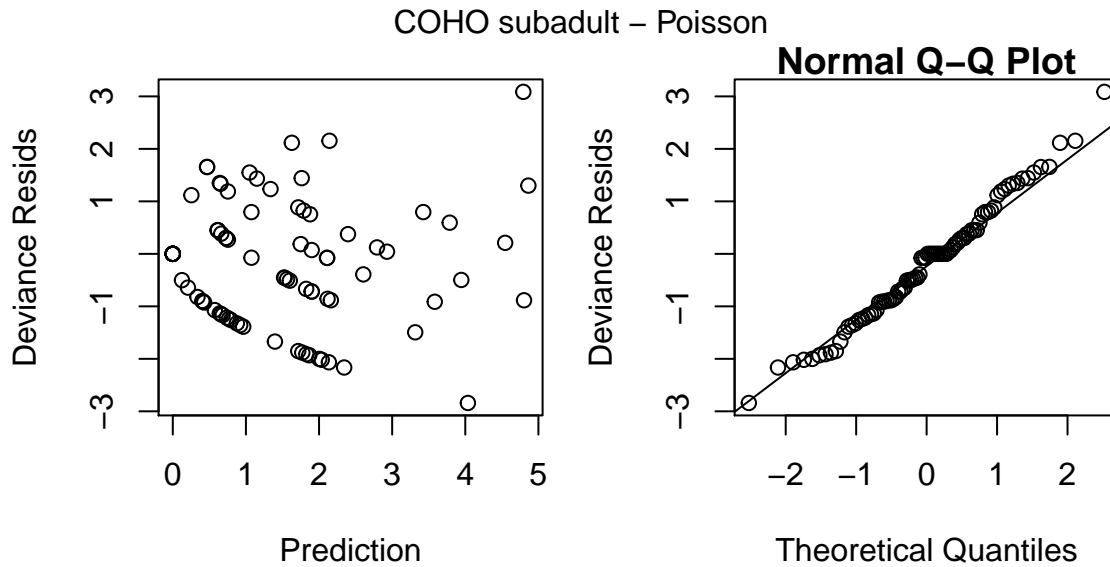


Figure 9:

```
##
## Model: Negative Binomial(3.6866), link: log
##
## Response: Count
##
## Terms added sequentially (first to last)
##
##      Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                85    180.195
## Block 11    87.323      74    92.872 5.602e-14 ***
## Gear   2    10.248      72    82.624 0.005957 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##      Up      Down
## Mean 0.6639356 2.872783
## 0.05 0.3817730 1.484663
## 0.25 0.5157354 2.114405
## 0.5  0.6338960 2.694712
## 0.75 0.7791285 3.434287
## 0.95 1.0525212 4.890992
##
## ***** COHO yearling *****
##
## *** METHOD 1: GLM ANODEV, Poisson ***
##
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: Count
##
```

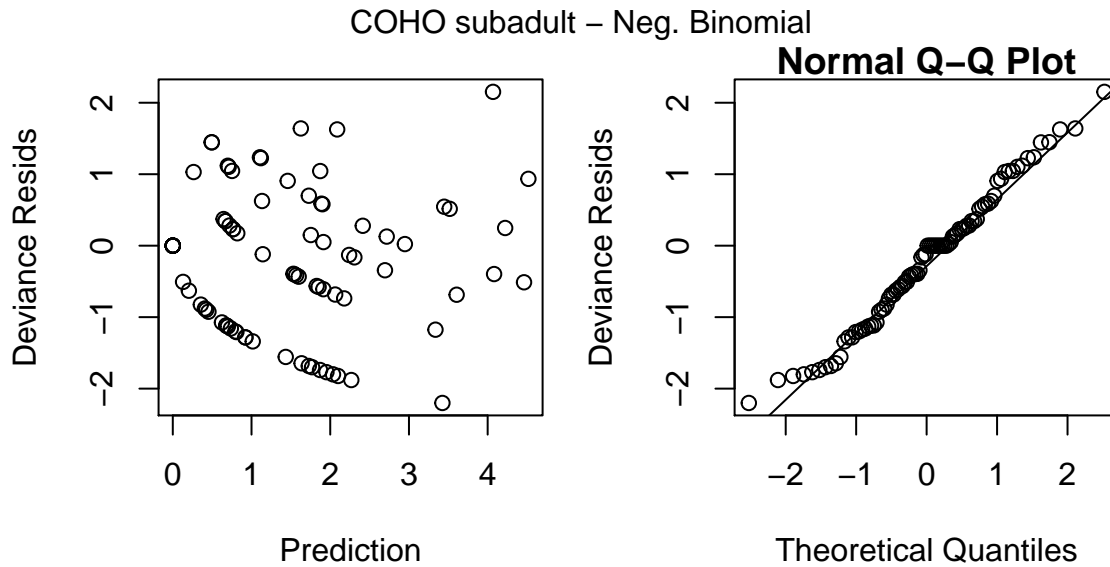


Figure 10:

```
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                85      588.09
## Block 11  220.599      74      367.49 < 2.2e-16 ***
## Gear   2    86.535      72      280.96 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##           Up      Down
## Mean 0.1860394 0.7047559
## 0.05 0.1255099 0.4868875
## 0.25 0.1562238 0.5985192
## 0.5  0.1815321 0.6895519
## 0.75 0.2109403 0.7944304
## 0.95 0.2625601 0.9765742
##
## *** METHOD 2: GLM ANODEV, negative binomial ***
## Initial Theta:  1.554399
##
## Initial fit to estimate theta
## Estimated theta:  1.504939 , SE:  0.3966753
##
## Final fit with theta =  1.504939
##
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: Negative Binomial(1.5049), link: log
##
## Response: Count
##
## Terms added sequentially (first to last)
```

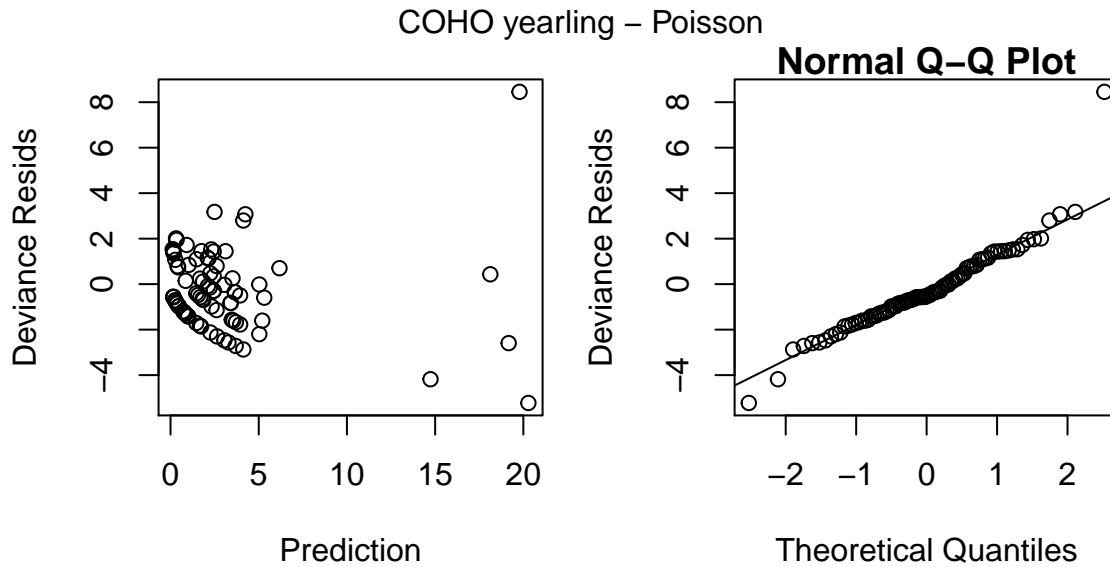


Figure 11:

```
##
##
##      Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                        85    171.099
## Block 11    72.115         74     98.983 3.228e-09 ***
## Gear   2    11.723         72     87.260 0.006241 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##           Up      Down
## Mean 0.3397104 0.7189217
## 0.05 0.1718611 0.3510543
## 0.25 0.2473126 0.5139029
## 0.5  0.3174407 0.6674197
## 0.75 0.4074544 0.8667963
## 0.95 0.5863376 1.2688893
##
## ***** MARKET SQUID *****
##
## *** METHOD 1: GLM ANODEV, Poisson ***
##
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: Count
##
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                        85    41497
## Block 11    31995         74     9502 < 2.2e-16 ***
```

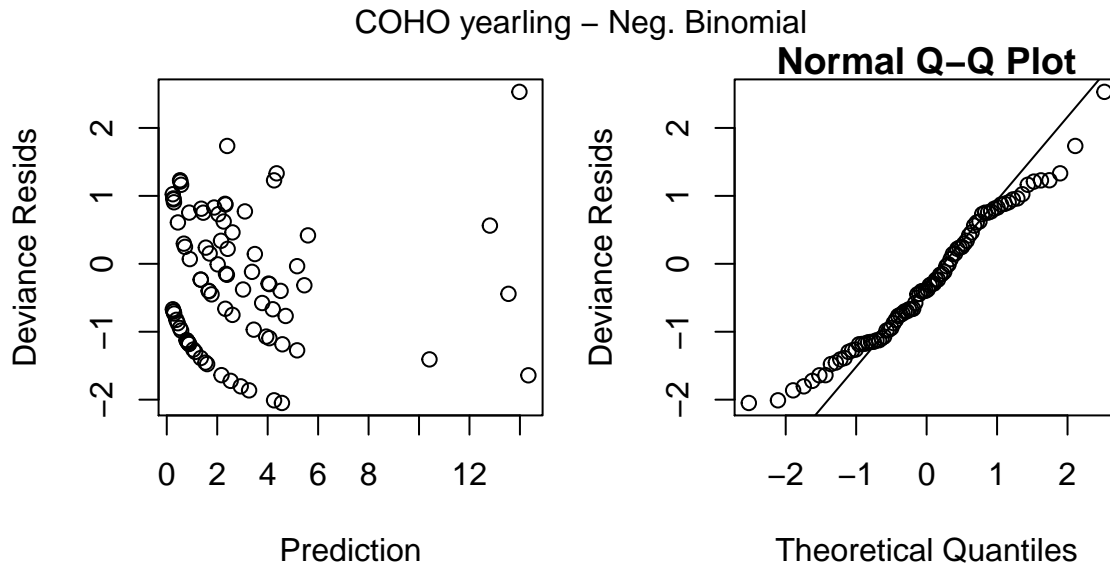


Figure 12:

```
## Gear    2      2833      72      6670 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##           Up      Down
## Mean 0.3499610 0.3202018
## 0.05 0.3099832 0.3073303
## 0.25 0.3326078 0.3148456
## 0.5  0.3490732 0.3201062
## 0.75 0.3663537 0.3254546
## 0.95 0.3930925 0.3334131
##
## *** METHOD 2: GLM ANODEV, negative binomial ***
## Initial Theta: 0.7905173
##
## Initial fit to estimate theta
## Estimated theta: 0.9008459 , SE: 0.1607128
##
## Final fit with theta = 0.9008459
##
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: Negative Binomial(0.9008), link: log
##
## Response: Count
##
## Terms added sequentially (first to last)
##
##           Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                85      433.55
## Block 11      329.48      74      104.08 <2e-16 ***
## Gear    2       4.46      72       99.61 0.1631
```

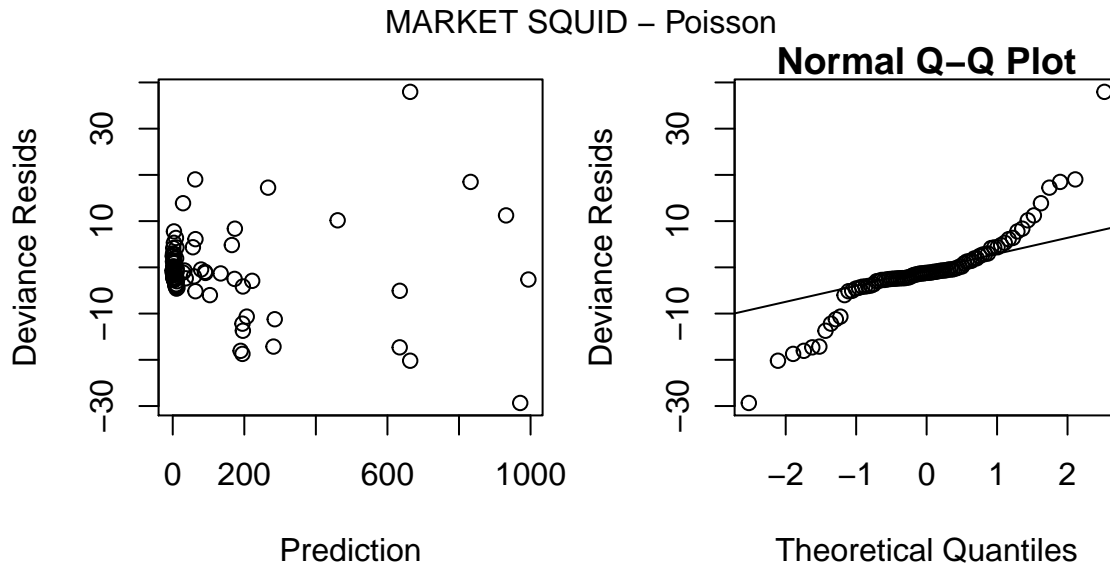
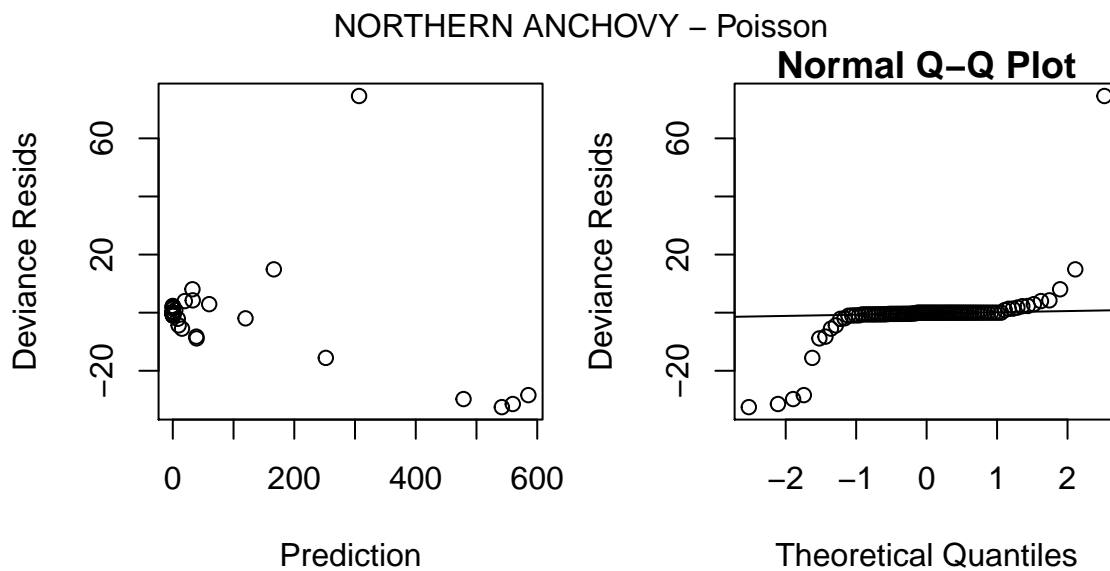
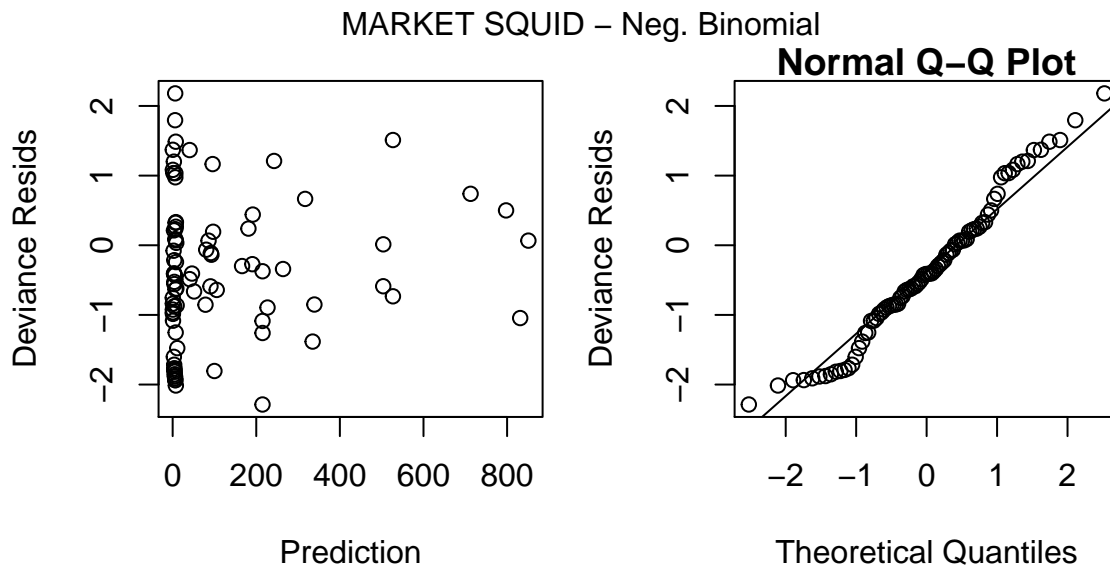


Figure 13:

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##           Up      Down
## Mean 0.8765406 0.4845325
## 0.05 0.4519912 0.2228432
## 0.25 0.6444103 0.3356991
## 0.5  0.8218839 0.4446347
## 0.75 1.0482347 0.5889203
## 0.95 1.4944830 0.8871711
##
## ***** NORTHERN ANCHOVY *****
##
## *** METHOD 1: GLM ANODEV, Poisson ***
##
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: Count
##
## Terms added sequentially (first to last)
##
##      Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                85      22390
## Block 11      9361.5      74  13029 < 2.2e-16 ***
## Gear   2      2929.5      72  10100 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##           Up      Down
## Mean 0.06762749 4.6141497
## 0.05 0.05972477 0.2828815
```

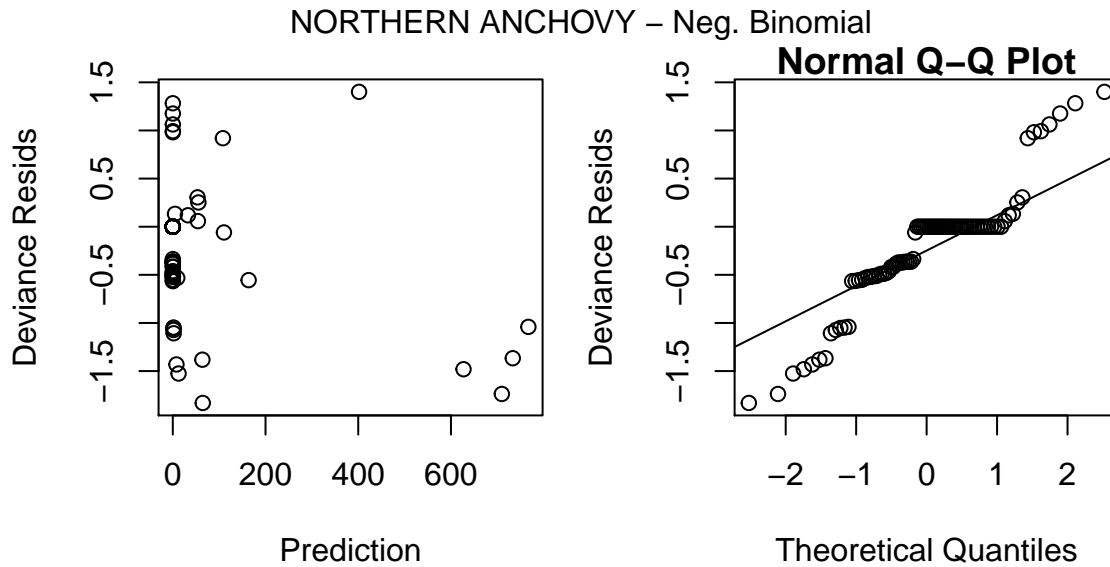


Figure 16:

```
## 0.25 0.06419196 0.9494745
## 0.5 0.06744764 2.1785820
## 0.75 0.07086843 4.9987855
## 0.95 0.07616913 16.7781196
##
## *** METHOD 2: GLM ANODEV, negative binomial ***
## Initial Theta: 0.3260892
##
## Initial fit to estimate theta
## Estimated theta: 0.314627 , SE: 0.09187918
##
## Final fit with theta = 0.314627
##
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: Negative Binomial(0.3146), link: log
##
## Response: Count
##
## Terms added sequentially (first to last)
##
##      Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                85    253.813
## Block 11    210.880      74    42.933 < 2.2e-16 ***
## Gear 2      6.113      72    36.821 0.002488 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##           Up      Down
## Mean 0.09891328 3.8901959
## 0.05 0.03385024 0.4174618
## 0.25 0.05841659 1.1492074
```

```

## 0.5 0.08493225 2.3016355
## 0.75 0.12348353 4.6097213
## 0.95 0.21310000 12.6898443
##
## ***** PACIFIC HERRING *****
##
## *** METHOD 1: GLM ANODEV, Poisson ***
##
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: Count
##
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                        85      19043.0
## Block 11  10470.5          74      8572.5 < 2.2e-16 ***
## Gear  2    2986.8          72      5585.7 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##           Up      Down
## Mean 0.11263741 233.0833
## 0.05 0.08864923 126.3052
## 0.25 0.10160500 175.7400
## 0.5  0.11156983 220.4245
## 0.75 0.12251196 276.4707
## 0.95 0.14041665 384.6789
##
## *** METHOD 2: GLM ANODEV, negative binomial ***
## Initial Theta: 0.09446641
##
## Initial fit to estimate theta
## [1] "Error in while ((it <- it + 1) < limit && abs(del) > eps) { : \n missing value where TRUE/FALSE"
## attr(,"class")
## [1] "try-error"
## attr(,"condition")
## <simpleError in while ((it <- it + 1) < limit && abs(del) > eps) { t0 <- abs(t0) del <- score(
## Warning in GLMAnoDevEst(cbind(MMEDcnt[[sp]], MMEDhauls), .qprobs, nb =
## TRUE, : glm.nb failed to estimate theta; using default value
##
## Final fit with theta = 0.09446641
##
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: Negative Binomial(0.0945), link: log
##
## Response: Count
##
## Terms added sequentially (first to last)

```

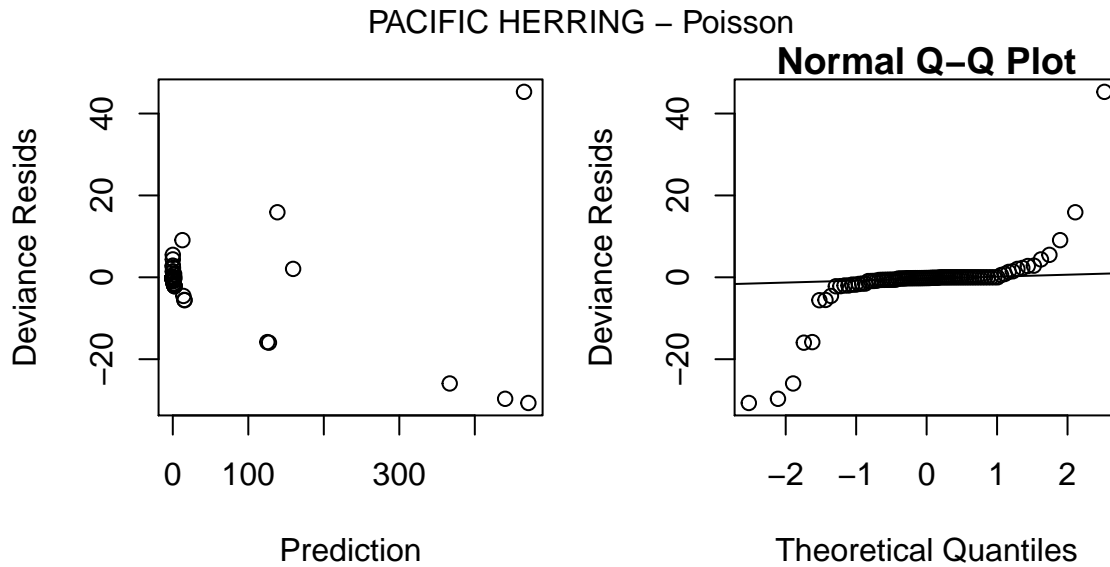


Figure 17:

```
##
##
##      Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                        85      98.474
## Block 11      70.323        74      28.151 < 2.2e-16 ***
## Gear   2       3.531        72      24.620 0.0006172 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##           Up      Down
## Mean 0.07043164 2.9876844
## 0.05 0.01206889 0.7724318
## 0.25 0.02781045 1.5053250
## 0.5  0.04930240 2.3789116
## 0.75 0.08740338 3.7594672
## 0.95 0.20140445 7.3264980
##
## ***** SEA NETTLE *****
##
## *** METHOD 1: GLM ANODEV, Poisson ***
##
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: Count
##
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                        85      5737.1
## Block 11      4848.0        74      889.0 <2e-16 ***
```

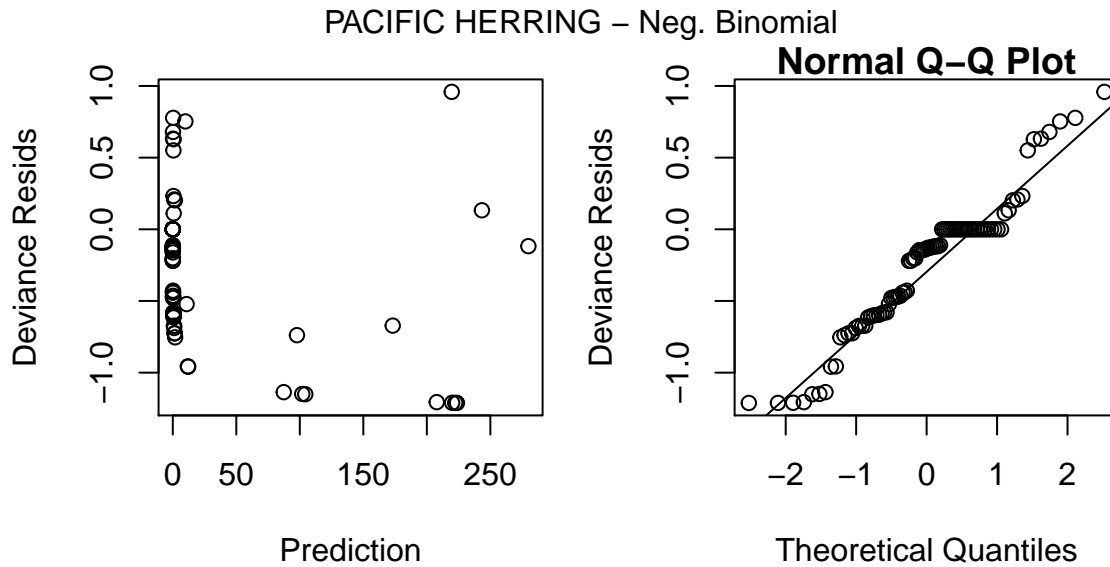


Figure 18:

```
## Gear      2      0.7      72      888.4      0.7119
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##           Up      Down
## Mean 0.9945518 0.8239794
## 0.05 0.9317755 0.4837972
## 0.25 0.9680875 0.6467325
## 0.5  0.9938081 0.7892015
## 0.75 1.0202120 0.9630551
## 0.95 1.0599704 1.2873971
##
## *** METHOD 2: GLM ANODEV, negative binomial ***
## Initial Theta:  2.154861
##
## Initial fit to estimate theta
## Estimated theta:  2.16212 , SE:  0.5290027
##
## Final fit with theta =  2.16212
##
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: Negative Binomial(2.1621), link: log
##
## Response: Count
##
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                    85      632.74
## Block 11      559.76      74      72.98 <2e-16 ***
## Gear      2       1.65      72      71.33  0.4406
```

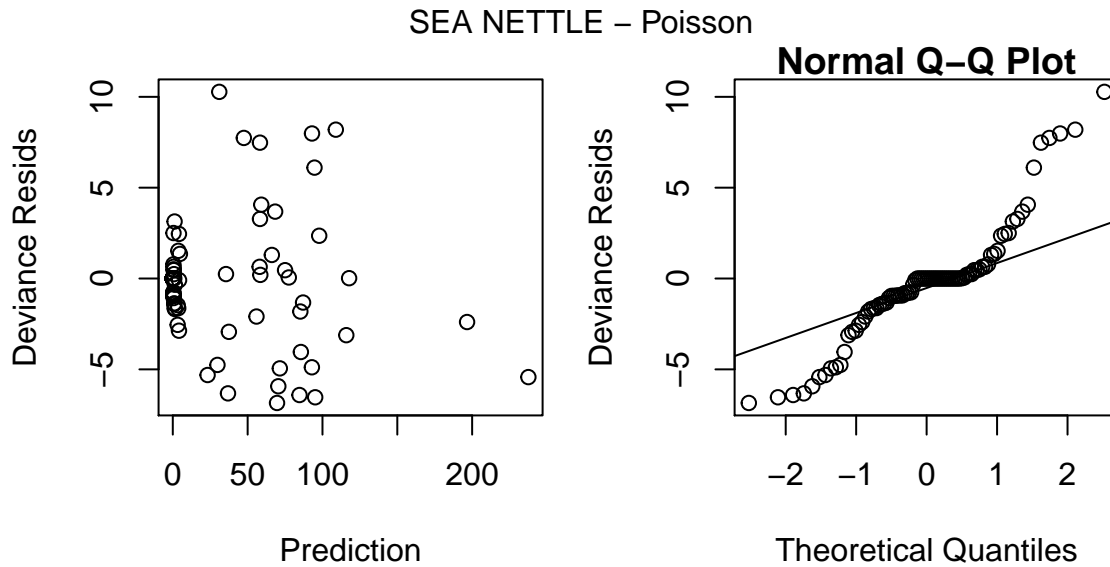


Figure 19:

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##           Up      Down
## Mean 0.9357340 0.6465416
## 0.05 0.6115211 0.2874541
## 0.25 0.7738936 0.4400243
## 0.5  0.9095491 0.5892529
## 0.75 1.0689838 0.7890904
## 0.95 1.3528226 1.2079109
##
## ***** WATER JELLY *****
##
## *** METHOD 1: GLM ANODEV, Poisson ***
##
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: Count
##
## Terms added sequentially (first to last)
##
##      Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                85      547720
## Block 11      468382      74      79338 < 2.2e-16 ***
## Gear   2       3313      72      76025 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##           Up      Down
## Mean 1.634405 0.7221444
## 0.05 1.454432 0.7152668
```

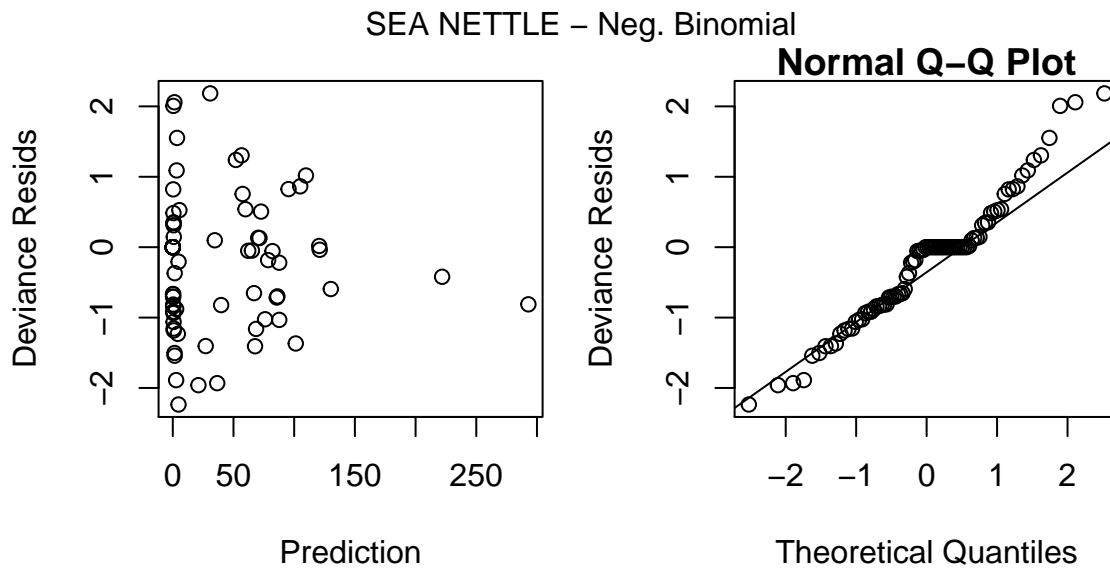


Figure 20:

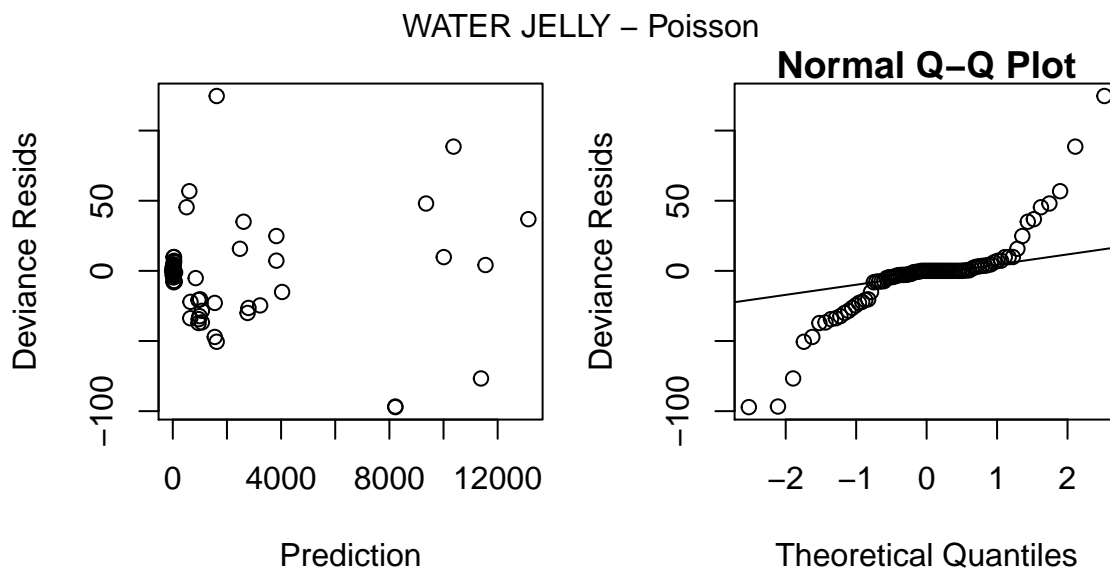


Figure 21:

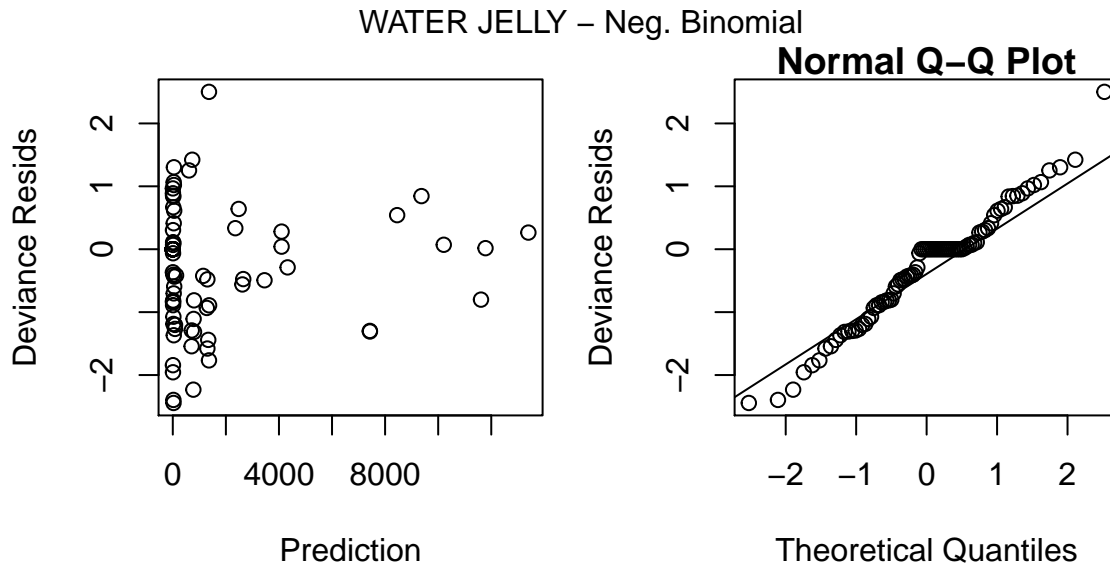


Figure 22:

```
## 0.25 1.556469 0.7193314
## 0.5 1.630563 0.7221325
## 0.75 1.708185 0.7249445
## 0.95 1.828025 0.7290641
##
## *** METHOD 2: GLM ANODEV, negative binomial ***
## Initial Theta: 0.85263
##
## Initial fit to estimate theta
## Estimated theta: 0.8668664 , SE: 0.1430373
##
## Final fit with theta = 0.8668664
##
## FIT STATISTICS:Analysis of Deviance Table
##
## Model: Negative Binomial(0.8669), link: log
##
## Response: Count
##
## Terms added sequentially (first to last)
##
##      Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                85      653.02
## Block 11      572.25      74      80.77 <2e-16 ***
## Gear    2        2.98      72      77.79 0.217
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##           Up      Down
## Mean 1.9209114 0.6862222
## 0.05 0.9583039 0.3424639
## 0.25 1.3884082 0.4960827
```



```
## 0.5 1.7904148 0.6396456
## 0.75 2.3088203 0.8247545
## 0.95 3.3450611 1.1947141
```

3.2 Summary of results

Method abbreviations for the summary are:

- GLM.Po - GLM blocked AnoDev with Poisson
- GLM.nb - GLM blocked AnoDev with negative binomial

```
print(stat.sum)
```

```
## $`CHINOOK subadult`
## , , Up
##
##           Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 1.030 0.681 0.856 1.003 1.174 1.478
## GLM.nb 1.074 0.647 0.854 1.033 1.249 1.649
##
## , , Down
##
##           Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 1.425 0.844 1.123 1.366 1.662 2.211
## GLM.nb 1.452 0.769 1.082 1.368 1.729 2.433
##
##
## $`CHINOOK subyearling`
## , , Up
##
##           Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 0.985 0.903 0.95 0.983 1.018 1.072
## GLM.nb 0.894 0.538 0.71 0.860 1.040 1.373
##
## , , Down
##
##           Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 0.649 0.554 0.607 0.647 0.689 0.755
## GLM.nb 0.510 0.296 0.398 0.488 0.598 0.803
##
##
## $`CHINOOK yearling`
## , , Up
##
##           Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 0.806 0.692 0.756 0.803 0.853 0.932
## GLM.nb 0.787 0.474 0.625 0.756 0.915 1.207
##
## , , Down
##
##           Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 1.128 0.863 1.004 1.114 1.236 1.439
## GLM.nb 1.239 0.705 0.957 1.181 1.457 1.978
##
```

```

##
## $`CHUM SALMON`
## , , Up
##
##      Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po    0    NA    NA    NA    NA    NA
## GLM.nb    0    NA    NA    NA    NA    NA
##
## , , Down
##
##      Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 0.391 0.296 0.346 0.386 0.429 0.502
## GLM.nb 0.386 0.236 0.309 0.372 0.448 0.587
##
##
## $`COHO subadult`
## , , Up
##
##      Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 0.619 0.397 0.507 0.601 0.711 0.909
## GLM.nb 0.664 0.382 0.516 0.634 0.779 1.053
##
## , , Down
##
##      Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 3.043 1.758 2.370 2.908 3.568 4.809
## GLM.nb 2.873 1.485 2.114 2.695 3.434 4.891
##
##
## $`COHO yearling`
## , , Up
##
##      Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 0.186 0.126 0.156 0.182 0.211 0.263
## GLM.nb 0.340 0.172 0.247 0.317 0.407 0.586
##
## , , Down
##
##      Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 0.705 0.487 0.599 0.690 0.794 0.977
## GLM.nb 0.719 0.351 0.514 0.667 0.867 1.269
##
##
## $`MARKET SQUID`
## , , Up
##
##      Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 0.350 0.310 0.333 0.349 0.366 0.393
## GLM.nb 0.877 0.452 0.644 0.822 1.048 1.494
##
## , , Down
##
##      Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 0.320 0.307 0.315 0.320 0.325 0.333

```

```

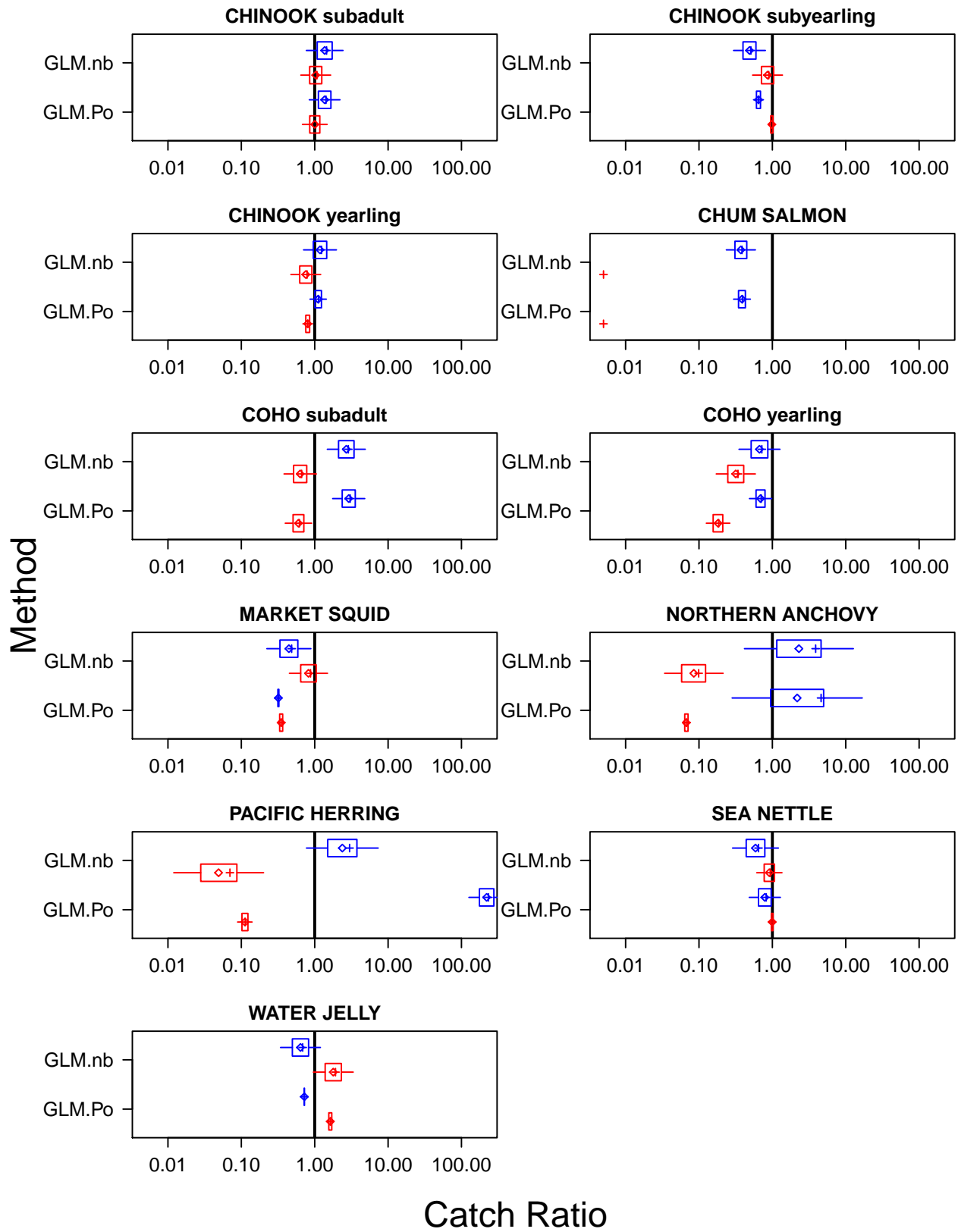
## GLM.nb 0.485 0.223 0.336 0.445 0.589 0.887
##
##
## $`NORTHERN ANCHOVY`
## , , Up
##
##      Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 0.068 0.060 0.064 0.067 0.071 0.076
## GLM.nb 0.099 0.034 0.058 0.085 0.123 0.213
##
## , , Down
##
##      Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 4.614 0.283 0.949 2.179 4.999 16.778
## GLM.nb 3.890 0.417 1.149 2.302 4.610 12.690
##
##
## $`PACIFIC HERRING`
## , , Up
##
##      Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 0.113 0.089 0.102 0.112 0.123 0.140
## GLM.nb 0.070 0.012 0.028 0.049 0.087 0.201
##
## , , Down
##
##      Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 233.083 126.305 175.740 220.424 276.471 384.679
## GLM.nb 2.988 0.772 1.505 2.379 3.759 7.326
##
##
## $`SEA NETTLE`
## , , Up
##
##      Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 0.995 0.932 0.968 0.994 1.020 1.060
## GLM.nb 0.936 0.612 0.774 0.910 1.069 1.353
##
## , , Down
##
##      Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 0.824 0.484 0.647 0.789 0.963 1.287
## GLM.nb 0.647 0.287 0.440 0.589 0.789 1.208
##
##
## $`WATER JELLY`
## , , Up
##
##      Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 1.634 1.454 1.556 1.631 1.708 1.828
## GLM.nb 1.921 0.958 1.388 1.790 2.309 3.345
##
## , , Down
##

```

```
##           Mean q0.05 q0.25 Median q0.75 q0.95
## GLM.Po 0.722 0.715 0.719 0.722 0.725 0.729
## GLM.nb 0.686 0.342 0.496 0.640 0.825 1.195
```

Next, generate summary figures of catch ratio estimates. In the plots, the estimated mean is marked with “+” and the median with a diamond; boxes span the quartiles, and whiskers extend to the 5% and 95% quantiles.

```
par(mfrow=c(ceiling(length(stat.sum)/plcol), plcol), oml=c(0.5,0.5,0,0), mar=c(3,4,2,1))
for (sp in names(stat.sum)) {
  .sumtbl <- stat.sum[[sp]]
  .sumtbl[.sumtbl==Inf] <- 99 # recode infinite values as +99
  .minx <- 5e-3
  .sumtbl[.sumtbl<.minx] <- .minx # recode zeros as small pos. value (for log scale plots)
  .nstats <- dim(.sumtbl)[1]
  .ngears <- dim(.sumtbl)[3]
  .ny <- .ngears*.nstats # number of elements along y axis.
  .gclr <- rep(c('red','blue','green3')[1:.ngears], .nstats) # gear color codes
  .xmax <- ceiling(max(.sumtbl[, 'q0.75', ], na.rm=TRUE)) # make sure the quartiles are covered
  .xmax <- max(.xmax, 2) #make sure upper bound is above 1
  .xmax <- min(.xmax, 10) #truncate high values so plot is readable
  .xlim <- c(.minx, 1/.minx)
  plot(t(.sumtbl[, 'Mean', ]), log='x', 1:.ny, col=.gclr, pch=3, cex=0.75, axes=F,
       xlim=.xlim, ylim=c(0.5, .ny+0.5),
       main=sp, xlab='', ylab='')
  box()
  abline(v=1, lwd=2, col='black')
  axis(side=1, cex.axis=1.2)
  axis(side=2, at=seq(1,.ny,.ngears)+1/.ngears, labels=dimnames(.sumtbl)[[1]], las=2, cex.axis=1.2)
  points(t(.sumtbl[, 'Median', ]), 1:.ny, col=.gclr, pch=5, cex=0.75)
  rect(t(.sumtbl[, 'q0.25', ]), (1:.ny)-0.35, t(.sumtbl[, 'q0.75', ]), (1:.ny)+0.35, border=.gclr)
  segments(t(.sumtbl[, 'q0.05', ]), 1:.ny, t(.sumtbl[, 'q0.25', ]), 1:.ny, col=.gclr, lwd=1)
  segments(t(.sumtbl[, 'q0.75', ]), 1:.ny, t(.sumtbl[, 'q0.95', ]), 1:.ny, col=.gclr, lwd=1)
} # for (sp)
mtext('Catch Ratio', outer=T, side=1, line=1, cex=1.5)
mtext('Method', outer=T, side=2, line=1, cex=1.5)
```

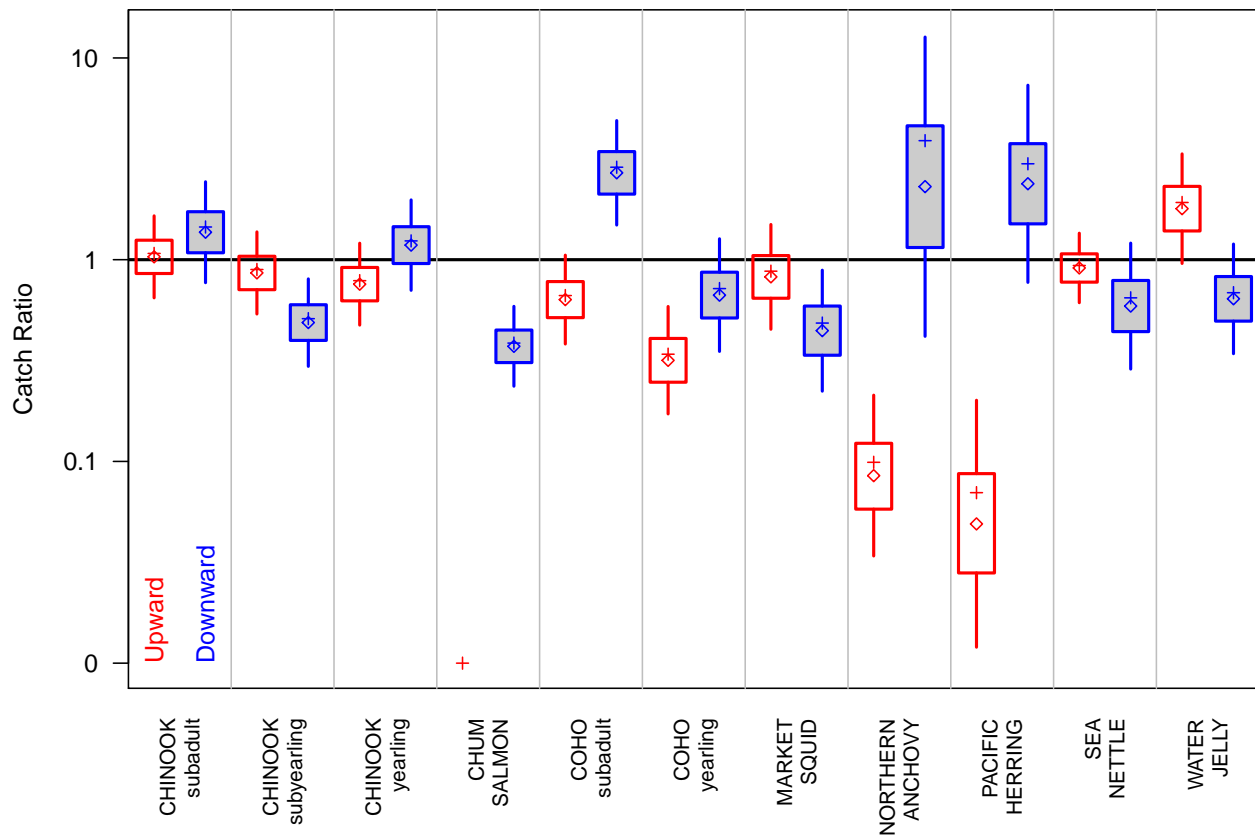


And a single panel summary of just the GLM.nb results for both excluder types.

```

par(mfrow=c(1,1), omi=c(0.5,0.5,0,0), mar=c(4,4,1,1))
.sumbtbl <- simplify2array(stat.sum)[ "GLM.nb", , , ] # array: probs x gear x species
.sumbtbl[.sumtbl==Inf] <- 99 # recode infinite values as +99
.miny <- 1e-2
.sumbtbl[.sumtbl<.minx] <- .miny # recode zeros as small pos. value (for log scale plots)
.ngears <- dim(.sumtbl)[2]
.nspecs <- dim(.sumtbl)[3]
.nx <- .ngears*.nspecs # number of elements along y axis.
.gbox <- rep(c('red','blue','green3')[1:.ngears], .nspecs) # box colors
.gpnt <- rep(c('red','blue','grey80')[1:.ngears], .nspecs) # point colors
.gfill <- rep(c(NA,'grey80','green3')[1:.ngears], .nspecs) # fill colors
.maxy <- ceiling(max(.sumtbl[ 'q0.95', , ], na.rm=TRUE))
.ylim <- c(.miny, .maxy)
plot(1:.nx, .sumtbl['Mean', , ], type='n', log='y', xaxs='i', axes=F,
     ylim=.ylim, xlim=c(0.5, .nx+0.5),
     xlab='', ylab='Catch Ratio')
box()
abline(h=1, lwd=2, col='black')
# label "fake zero" as zero:
axis(side=2, at=c(.miny,0.10,1,10,100), labels=c('0','0.1','1','10','100'), las=2)
.labs <- dimnames(.sumtbl)[[3]]
.labs <- sub(' ', '\n', .labs)
axis(side=1, at=seq(1, .nx, .ngears) + 1/.ngears, tick=FALSE, labels=.labs,
     las=2, cex.axis=0.8)
abline(v=seq(.ngears+1, .nx, .ngears) - 1/.ngears, col="grey")
rect((1:.nx)-0.35, .sumtbl['q0.25', , ], (1:.nx)+0.35, .sumtbl['q0.75', , ],
     border=.gbox, col=.gfill, lwd=2)
segments(1:.nx, .sumtbl['q0.05', , ], 1:.nx, .sumtbl['q0.25', , ], col=.gbox, lwd=2)
segments(1:.nx, .sumtbl['q0.75', , ], 1:.nx, .sumtbl['q0.95', , ], col=.gbox, lwd=2)
points(1:.nx, .sumtbl['Median', , ], col=.gpnt, pch=5, cex=0.75)
points(1:.nx, .sumtbl['Mean', , ], col=.gpnt, pch=3, cex=0.75)
text(1, .miny, "Upward", col="red", adj=c(0,0.5), srt=90)
text(2, .miny, "Downward", col="blue", adj=c(0,0.5), srt=90)

```



Part 4: Size-selectivity analysis

We analyze size-frequencies for the same set of species selected above, but do not subdivide salmon species by age group. For an unbiased analysis, we first compute the size subsampling ratio (individuals measured / number caught) for each species in each haul, and compute adjusted counts by size. We then bin sizes into 5 mm intervals, and eliminate a couple size outliers for anchovy and water jelly (appear to be erroneous data that are overly influential in the analysis).

```
# Restrict to species selected above, but no age groups for salmon
len.spec <- c("CHINOOK SALMON", "CHUM SALMON", "COHO SALMON",
             "MARKET SQUID", "NORTHERN ANCHOVY", "PACIFIC HERRING",
             "SEA NETTLE", "WATER JELLY")
lenData <- MMEDdata[ , c("Cruise", "MMED", "Species", "Length",
                        "Number", "Distance", "Haul")]
lenData <- lenData[lenData$Species %in% len.spec, ]
# Add subsample ratio:
# Total number by Haul (rows) and Species (cols)
cnt <- with(lenData, tapply(Number, list(Haul, Species), FUN=sum, simplify=T))
cnt[is.na(cnt)] <- 0
cnt <- as.data.frame(cnt)
# Total measured by Haul & species
meas <- with(lenData, tapply(!is.na(Length), list(Haul, Species), FUN=sum, simplify=T))
meas[is.na(meas)] <- 0
meas <- as.data.frame(meas)
# Subsampling ratio by Haul & Species:
ssr <- meas / cnt
cat('\nSubsampling Ratios:\n')
```

##

```
## Subsampling Ratios:
```

```
print(summary(ssr))
```

```
## CHINOOK SALMON      CHUM SALMON      COHO SALMON      MARKET SQUID
## Min.   :0.1328      Min.   :0.6923      Min.   :0.6618      Min.   :0.008602
## 1st Qu.:0.7674      1st Qu.:0.8000      1st Qu.:1.0000      1st Qu.:0.114096
## Median :0.9045      Median :1.0000      Median :1.0000      Median :0.660256
## Mean   :0.8346      Mean   :0.9054      Mean   :0.9851      Mean   :0.542583
## 3rd Qu.:1.0000      3rd Qu.:1.0000      3rd Qu.:1.0000      3rd Qu.:1.000000
## Max.   :1.0000      Max.   :1.0000      Max.   :1.0000      Max.   :1.000000
## NA's   :2          NA's   :53         NA's   :11         NA's   :18
## NORTHERN ANCHOVY    PACIFIC HERRING      SEA NETTLE      WATER JELLY
## Min.   :0.00253      Min.   :0.01261      Min.   :0.06829      Min.   :0.000679
## 1st Qu.:0.33095      1st Qu.:0.46711      1st Qu.:0.21333      1st Qu.:0.010392
## Median :0.92857      Median :1.00000      Median :0.41463      Median :0.140222
## Mean   :0.67990      Mean   :0.75391      Mean   :0.53817      Mean   :0.287572
## 3rd Qu.:1.00000      3rd Qu.:1.00000      3rd Qu.:1.00000      3rd Qu.:0.528846
## Max.   :1.00000      Max.   :1.00000      Max.   :1.00000      Max.   :1.000000
## NA's   :66         NA's   :70         NA's   :37         NA's   :24
```

```
# Adjusted Numbers (expanded by ssr)
lenData <- lenData[!is.na(lenData$Length), ] #remove non-measured counts
lenData$AdjNum <- lenData$Number / unlist(apply(lenData[c("Haul","Species")], 1,
function(x){ssr[x["Haul"], x["Species"]]}))
# Length bin size (mm)
binsize <- 5
lenData$LenBin <- binsize * round(lenData$Length/binsize)
# Remove size outliers for anchovy & water jelly (likely data errors)
lenData <- lenData[!((lenData$Species == "NORTHERN ANCHOVY") &
(lenData$Length < 100)), ]
lenData <- lenData[!((lenData$Species == "WATER JELLY") &
(lenData$Length > 150)), ]
cat('\nSummary of Length Data:\n')
```

```
##
```

```
## Summary of Length Data:
```

```
print(summary(lenData))
```

```
##      Cruise      MMED      Species      Length
## Min.   :41.00      Length:4385      Length:4385      Min.   : 15.0
## 1st Qu.:43.00      Class :character      Class :character      1st Qu.: 73.0
## Median :50.00      Mode  :character      Mode  :character      Median :115.0
## Mean   :47.99
## 3rd Qu.:53.00
## Max.   :53.00
##      Number      Distance      Haul      AdjNum
## Min.   : 1.000      Min.   :1.129      Length:4385      Min.   : 1.000
## 1st Qu.: 1.000      1st Qu.:1.506      Class :character      1st Qu.: 1.179
## Median : 1.000      Median :1.860      Mode  :character      Median : 3.154
## Mean   : 1.702      Mean   :2.396
## 3rd Qu.: 2.000      3rd Qu.:3.513
## Max.   :33.000      Max.   :4.906
##      LenBin
## Min.   : 15.0
```



```
## 1st Qu.: 75.0
## Median :115.0
## Mean   :137.1
## 3rd Qu.:160.0
## Max.   :795.0
```

Next, run the Size-Frequency (SF) analysis for each species. Because the length data is sparse for most species, we analyze only the bulk data across all haul samples, ignoring the data blocks. First tabulate the distributions across all samples by species and gear type. Then, apply both a Wilcox-Mann-Whitney test and a Kolmogorov-Smirnov test for gear differences. **NOTE** the several warnings that p-values are approximate for the K-S test. We'll ignore the warnings, assuming they're close enough as we're concerned with the patterns of size-selectivity, and only note the highly significant results.

Then, conduct full gear Size-Selectivity Analysis, fitting a smooth curve (3rd-order polynomial) to the size-specific catch ratio data using a binomial GLM with logit link function, similar to models in Krag et al. (2014 PLOS One), Herrmann et al. (2017 Fish. Res.), & Kotwicki et al. (2017 Fish. Res.). The model is first fit to the full data set, then the error distribution is approximated via a double bootstrap (Millar 1993 Fish. Bull.) resampling both among hauls and among fish within hauls.

Because the two MED orientations were used in different years and locations, we can't compare either MED to all the samples from the standard net, so we separate this analysis into two groups: the upward MED tests in years 2011 & 2014, and the downward MED tests in 2015.

```
boot_GLM3P <- function(sdat, nrep=10, binsz=5, L.pr=NULL) {
  fit.model <- function(sdat) {
    NumTotL <- with(sdat, tapply(AdjNum, list(LenBin, MMED), sum,
                                   na.rm=TRUE, default=0))
    EffTotL <- with(sdat, tapply(Distance, list(LenBin, MMED), sum,
                                   na.rm=TRUE, default=0))

    cpue <- NumTotL/EffTotL
    cpue[is.na(cpue)] <- 0
    STD <- match("None", colnames(cpue))
    TST <- match("Up", colnames(cpue))
    if (is.na(TST)) TST <- match("Down", colnames(cpue))
    std <- cpue[ , STD]
    tst <- cpue[ , TST]
    p.L12 <- std / (std + tst)
    # Binomial weights based on number measured in both gears:
    Nmeas <- with(sdat, tapply(Number, list(LenBin, MMED), sum,
                                   na.rm=TRUE, default=0))
    wts <- Nmeas[ , STD] + Nmeas[ , TST]
    L <- as.numeric(names(p.L12))
    old.opt <- options(warn = -1) # suppress warnings about non-integer values
    fit.glm <- glm(p.L12 ~ L + I(L^2) + I(L^3), family=binomial, weights=wts)
    options(old.opt)
    return(fit.glm)
  } # fit.model()

  # Fit the model to the original (full) dataset:
  fit.full <- fit.model(sdat)
  # Predictions of full model, with rough SE's
  if (is.null(L.pr)) L.pr <- seq(min(sdat$LenBin), max(sdat$LenBin), 5)
  pred.full <- predict(fit.full, newdata=data.frame(L=L.pr, wts=1.0),
                      type="response")
  names(pred.full) <- L.pr
  # Bootstrap predictions:
```

```

bs <- matrix(NA, nrow=length(L.pr), ncol=nrep, dimnames=list(L.pr, NULL))
rep <- 0
while (rep < nrep) {
  hauls <- unique(sdat$Haul)
  hauls.samp <- sample(hauls, length(hauls), replace=TRUE)
  .data <- data.frame()
  for (h in hauls.samp) {
    .hdata <- sdat[sdat$Haul == h, ]
    ssr <- with(.hdata, sum(Number)/sum(AdjNum)) # subsample rate
    L.ex <- with(.hdata, rep(LenBin, Number)) #expand Number to indiv. lengths
    if(length(L.ex) > 1) {
      L.smp <- sample(L.ex, length(L.ex), replace=TRUE) #resample lengths
    } else {
      L.smp <- L.ex # sample() doesn't work for length 1 vector
    }
    new.freq <- as.data.frame(table(L.smp))
    .ndata <- data.frame(MMED=unique(.hdata$MMED),
                        Haul=unique(.hdata$Haul),
                        Distance=mean(.hdata$Distance),
                        LenBin=as.numeric(levels(new.freq$L.smp)),
                        Number=new.freq$Freq,
                        AdjNum=new.freq$Freq/ssr)
    .data <- rbind(.data, .ndata)
  } # for (h)
  names(.data) <- names(sdat)
  fit.rep <- fit.model(.data)
  rep <- rep+1
  bs[, rep] <- predict(fit.rep, newdata=data.frame(L=L.pr, wts=1.0),
                      type="response")
} # for (rep)
rownames(bs) <- L.pr
bs.mn <- apply(bs, 1, mean, na.rm=FALSE)
bs.q <- t(apply(bs, 1, quantile, probs=c(0,0.05,0.25,0.50,0.75,0.95,1),
              na.rm=FALSE))
return(list(gam=fit.full, pred=pred.full, boot=bs,
            boot.sum=data.frame(mean=bs.mn, q=bs.q)))
} # boot_GLM3P()

for (excl in c("Up", "Down")) {
  cat('\n***** Excluder: ', excl, ' *****\n')
  lD <- if(excl %in% "Up") {
    lenData[lenData$Cruise %in% c(41,43,50), ]
  } else {
    lenData[lenData$Cruise %in% 53, ]
  }
  lD$Species <- factor(as.character(lD$Species))
  .tab <- with(lD, tapply(AdjNum, list(Species), sum, na.rm=T))
  cat('Total Adjusted Catch:\n')
  print(.tab)
  lf.sel.spec <- names(.tab)[.tab >= 100]
  lenFreq <- with(lD, tapply(AdjNum, list(LenBin, MMED, Species), sum, na.rm=TRUE))
  lenFreq[is.na(lenFreq)] <- 0
  .mfrow <- if(lndscp) c(3,3) else c(3,2)

```

```

par(mfrow=.mfrow, omi=c(0.5,0.5,0,0.5), mar=c(3,3,2,3))
for (sp in lf.sel.spec) {
  cat('\n*****\n', sp, '*****\n')
  if (sp %in% dimnames(lenFreq)[[3]]) {
    .dat <- lenFreq[, , sp]
    .maxN <- max(.dat, na.rm=T)
    .len <- 1D[1D$Species %in% sp,
      c("MMED", "Haul", "Distance", "LenBin", "Number", "AdjNum")]
    .maxL <- max(.len$LenBin, na.rm=T)
    .minL <- min(.len$LenBin, na.rm=T)
    .len.std <- .len[.len$MMED=='None', ]
    .len.mmed <- .len[.len$MMED==excl, ]
    .x <- rep(.len.std$LenBin, .len.std$Number)
    .y <- rep(.len.mmed$LenBin, .len.mmed$Number)
    # Run analysis only if > 40 measurements in each gear:
    if ((length(.x) > 40) & (length(.y) > 40)) {
      # Wilcox & KS test for overall difference in size-frequencies
      print(wilcox.test(.x, .y, alt="two.sided"))
      print(ks.test(.x, .y))
      # GLM fit of Catch Ratio to size:
      ##mod.fit <- boot_GLM3P(sdat=.len, nrep=50, binsz=binsize) ### Testing ###
      mod.fit <- boot_GLM3P(sdat=.len, nrep=1000, binsz=binsize) ### Production ###
      cat("\tSummary of GAM fit: \n")
      print(summary(mod.fit$gam))
      cat("\tSummary of bootstrap fits: \n")
      print(summary(mod.fit$boot.sum))
      p.pred <- mod.fit$pred
      L.pred <- as.numeric(names(p.pred))
      # Convert probability to Catch Ratio:
      CR.obs <- 1/mod.fit$gam$model$p.L12 - 1
      CR.pred <- 1/p.pred - 1
      CR.boot <- 1/mod.fit$boot.sum - 1
      CR.boot[CR.boot > 1000] <- 1000 #recode infinite values
      CR.boot[CR.boot < 1/1000] <- 1/1000 #recode zero values
      plot(as.numeric(rownames(.dat)), -.dat[, 'None'], type='h', col=BLACK, lwd=1,
        xlim=c(.minL, .maxL), ylim=c(-.maxN, .maxN), axes=F,
        xlab='', ylab='')
      box()
      axis(side=1, lwd=0, lwd.ticks=1)
      axis(side=2, at=pretty(c(-max(.dat, na.rm=T), max(.dat, na.rm=T))),
        labels=abs(pretty(c(-max(.dat, na.rm=T), max(.dat, na.rm=T)))))
      points(as.numeric(rownames(.dat)), .dat[, excl], type='h',
        col={if(excl %in% "Up") RED else BLUE}, lwd=1)
      mtext(sp, side=3, cex=0.75, line=0)
      abline(h=0, lwd=1)
      # add Catch Ratio plot on right axis
      par(new=TRUE)
      plot(L.pred, CR.pred, log="y", type='l', lwd=3, axes=FALSE, bty="n",
        ylim=c(1/50, 50), xlab="", ylab="")
      axis(side=4, at=c(0.02, 0.2, 1, 5, 50),
        labels=c("0.02", "0.2", "1", "5", "50"))
      lines(L.pred, CR.boot[, "q.50."], lty=1, col='gray50', lwd=2) # bs median
      lines(L.pred, CR.boot[, "q.5."], lty=2, col='gray50', lwd=2) # bs lower 5%
    }
  }
}

```

```

        lines(L.pred, CR.boot[ , "q.95."], lty=2, col='gray50', lwd=2) # bs upper 5%
    } else {
        cat('\n Insufficient data \n')
    } # if (length...)
} else {
    cat('\n\tNO LENGTH DATA FOR ', sp, '\n')
} # if (sp %in% ...)
} # for(sp)
mtext('Size (mm)', side=1, outer=TRUE, at=c(0.5,0.5), cex=1.5)
mtext('Observed Catch Per Km', side=2, outer=TRUE, at=c(0.5,0.5), cex=1.5)
mtext('Estimated Catch Ratio', side=4, outer=TRUE, at=c(0.5,0.5), cex=1.5)
} # for (excl)

```

```
## Warning in ks.test(.x, .y): p-value will be approximate in the presence of
## ties
```

```
## Warning in ks.test(.x, .y): cannot compute exact p-value with ties
```

```
## Warning in ks.test(.x, .y): p-value will be approximate in the presence of
## ties
```

```
## Warning in ks.test(.x, .y): p-value will be approximate in the presence of
## ties
```

```
## Warning in ks.test(.x, .y): p-value will be approximate in the presence of
## ties
```

```
## Warning in ks.test(.x, .y): p-value will be approximate in the presence of
## ties
```

```
## Warning in ks.test(.x, .y): p-value will be approximate in the presence of
## ties
```

```
## Warning in ks.test(.x, .y): cannot compute exact p-value with ties
```

```
## Warning in ks.test(.x, .y): cannot compute exact p-value with ties
```

```
## Warning in ks.test(.x, .y): p-value will be approximate in the presence of
## ties
```

```
## Warning in ks.test(.x, .y): p-value will be approximate in the presence of
## ties
```

```
##
```

```
## ***** Excluder: Up *****
```

```
## Total Adjusted Catch:
```

## CHINOOK SALMON	CHUM SALMON	COHO SALMON	MARKET SQUID
## 7806.36006	24.34091	271.59921	1930.73203
## NORTHERN ANCHOVY	PACIFIC HERRING	SEA NETTLE	WATER JELLY
## 4545.80019	788.94286	5328.01283	2321.85789

```
##
```

```
## ***** CHINOOK SALMON *****
```

```
##
```

```
## Wilcoxon rank sum test with continuity correction
```

```
##
```

```
## data: .x and .y
```

```
## W = 557520, p-value = 0.3175
```

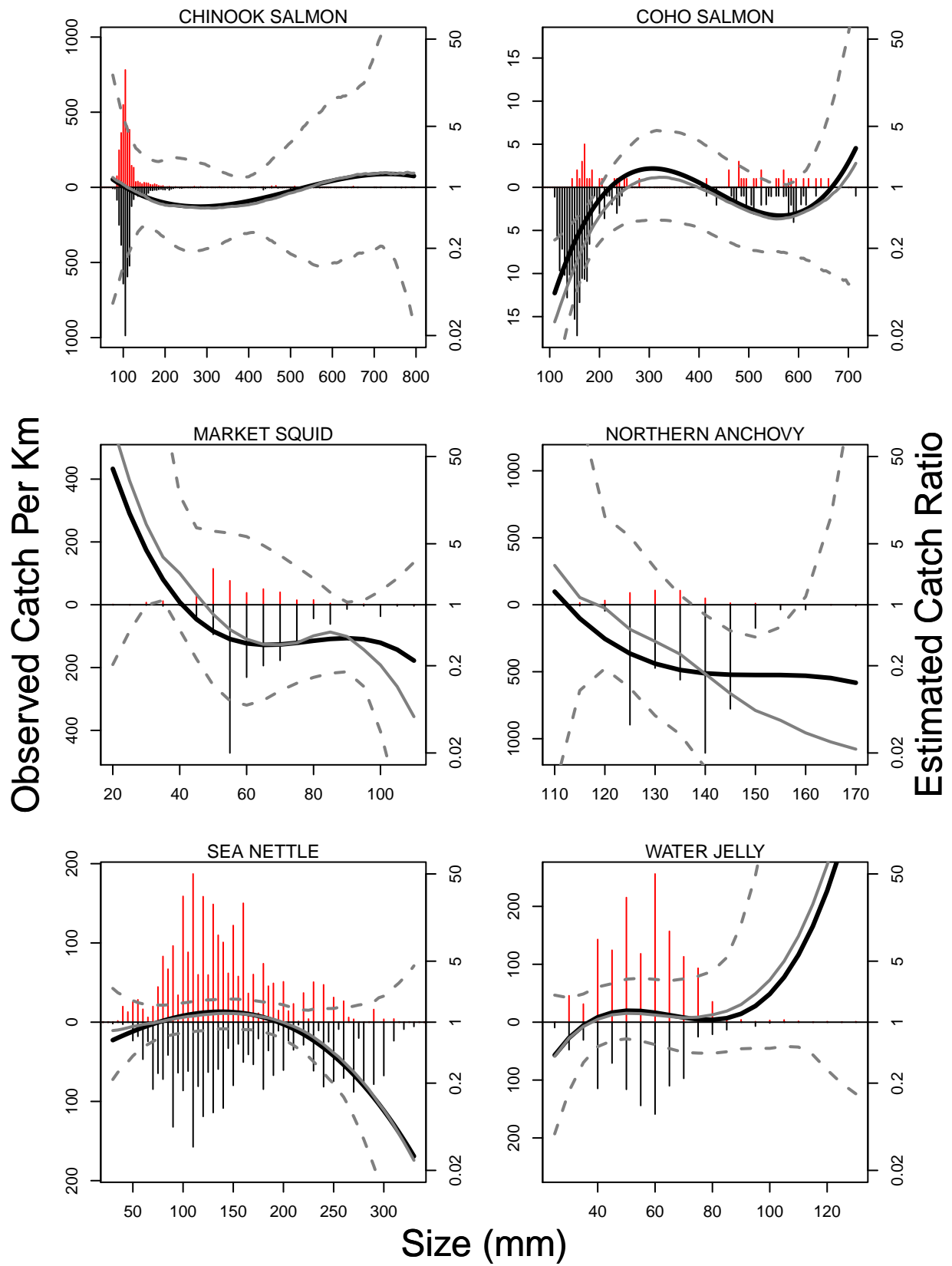


Figure 23:

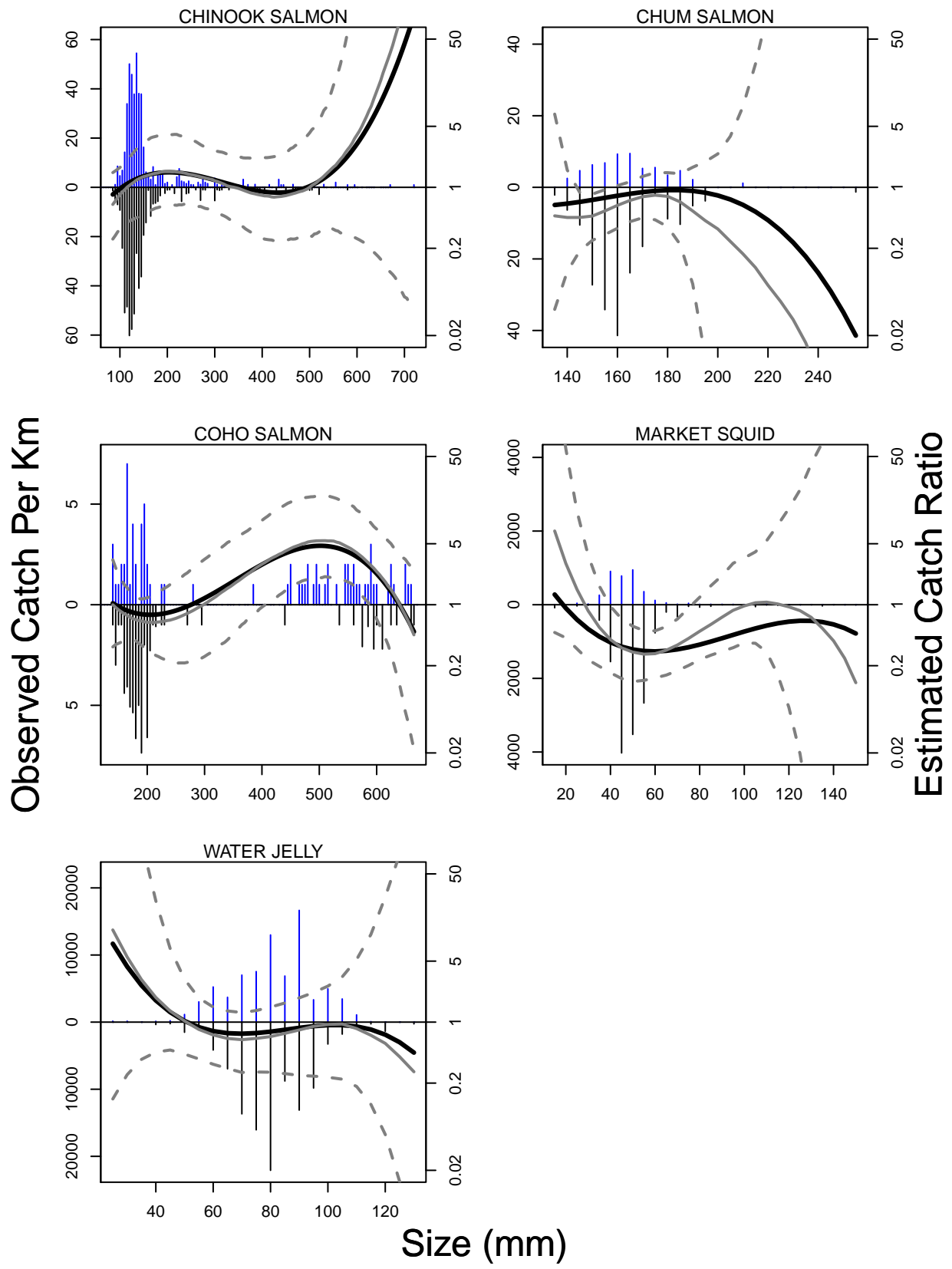


Figure 24:

```

## alternative hypothesis: true location shift is not equal to 0
##
##
## Two-sample Kolmogorov-Smirnov test
##
## data: .x and .y
## D = 0.028936, p-value = 0.7764
## alternative hypothesis: two-sided
##
## Summary of GAM fit:
##
## Call:
## glm(formula = p.L12 ~ L + I(L^2) + I(L^3), family = binomial,
##      weights = wts)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.4824  -1.0048   0.0389   1.0590   3.2229
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -9.376e-01  3.682e-01  -2.546  0.0109 *
## L            1.175e-02  4.754e-03   2.471  0.0135 *
## I(L^2)       -2.881e-05  1.573e-05  -1.832  0.0670 .
## I(L^3)        1.893e-08  1.423e-08   1.330  0.1835
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 132.65  on 87  degrees of freedom
## Residual deviance: 120.58  on 84  degrees of freedom
## AIC: 275.2
##
## Number of Fisher Scoring iterations: 4
##
## Summary of bootstrap fits:
##      mean           q.0.           q.5.           q.25.
## Min.    :0.4153   Min.    :0.00000000   Min.    :0.00037   Min.    :0.1427
## 1st Qu.:0.4516   1st Qu.:0.00008268   1st Qu.:0.08257   1st Qu.:0.2562
## Median :0.5340   Median :0.03281611   Median :0.25485   Median :0.4261
## Mean    :0.5251   Mean    :0.06707862   Mean    :0.22078   Mean    :0.3839
## 3rd Qu.:0.5991   3rd Qu.:0.12160065   3rd Qu.:0.33530   3rd Qu.:0.4951
## Max.    :0.6191   Max.    :0.23309205   Max.    :0.43201   Max.    :0.5450
##      q.50.           q.75.           q.95.           q.100.
## Min.    :0.4019   Min.    :0.5742   Min.    :0.7347   Min.    :0.8629
## 1st Qu.:0.4397   1st Qu.:0.6301   1st Qu.:0.8032   1st Qu.:0.9305
## Median :0.5373   Median :0.6597   Median :0.8407   Median :0.9650
## Mean    :0.5282   Mean    :0.6621   Mean    :0.8382   Mean    :0.9594
## 3rd Qu.:0.6163   3rd Qu.:0.6944   3rd Qu.:0.8711   3rd Qu.:0.9949
## Max.    :0.6339   Max.    :0.7389   Max.    :0.9741   Max.    :1.0000
##
## ***** COHO SALMON *****
##

```

```

## Wilcoxon rank sum test with continuity correction
##
## data: .x and .y
## W = 2478.5, p-value = 0.000001831
## alternative hypothesis: true location shift is not equal to 0
##
##
## Two-sample Kolmogorov-Smirnov test
##
## data: .x and .y
## D = 0.37991, p-value = 0.00002893
## alternative hypothesis: two-sided
##
## Summary of GAM fit:
##
## Call:
## glm(formula = p.L12 ~ L + I(L^2) + I(L^3), family = binomial,
##      weights = wts)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.3673  -0.9730   0.1931   1.2725   2.2525
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  9.004e+00  2.250e+00   4.001 0.000063 ***
## L           -7.567e-02  2.202e-02  -3.437 0.000588 ***
## I(L^2)        1.904e-04  6.037e-05   3.154 0.001610 **
## I(L^3)       -1.457e-07  5.075e-08  -2.872 0.004084 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 106.243  on 63  degrees of freedom
## Residual deviance:  87.423  on 60  degrees of freedom
## AIC: 143.29
##
## Number of Fisher Scoring iterations: 4
##
## Summary of bootstrap fits:
##      mean      q.0.      q.5.      q.25.
## Min.   :0.3905  Min.   :0.0000319  Min.   :0.006056  Min.   :0.0798
## 1st Qu.:0.4795  1st Qu.:0.0702695  1st Qu.:0.199964  1st Qu.:0.3520
## Median :0.5802  Median :0.1080961  Median :0.316143  Median :0.4776
## Mean   :0.5954  Mean   :0.1436925  Mean   :0.343229  Mean   :0.4924
## 3rd Qu.:0.6716  3rd Qu.:0.2094687  3rd Qu.:0.462386  3rd Qu.:0.6002
## Max.   :0.9443  Max.   :0.4396558  Max.   :0.800402  Max.   :0.9394
##      q.50.      q.75.      q.95.      q.100.
## Min.   :0.3466  Min.   :0.5493  Min.   :0.7006  Min.   :0.8401
## 1st Qu.:0.4845  1st Qu.:0.6151  1st Qu.:0.7396  1st Qu.:0.8886
## Median :0.5944  Median :0.7115  Median :0.8349  Median :0.9745
## Mean   :0.6046  Mean   :0.7052  Mean   :0.8215  Mean   :0.9477
## 3rd Qu.:0.6824  3rd Qu.:0.7565  3rd Qu.:0.8813  3rd Qu.:0.9964

```



```

## Max. :0.9721 Max. :0.9861 Max. :0.9948 Max. :1.0000
##
## ***** MARKET SQUID *****
##
## Wilcoxon rank sum test with continuity correction
##
## data: .x and .y
## W = 23156, p-value = 4.907e-08
## alternative hypothesis: true location shift is not equal to 0
##
## Two-sample Kolmogorov-Smirnov test
##
## data: .x and .y
## D = 0.23432, p-value = 0.00006958
## alternative hypothesis: two-sided
##
## Summary of GAM fit:
##
## Call:
## glm(formula = p.L12 ~ L + I(L^2) + I(L^3), family = binomial,
## weights = wts)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -2.9502 -0.9026 0.2298 1.1338 2.3714
##
## Coefficients:
## Estimate Std. Error z value Pr(>|z|)
## (Intercept) -10.81427430 3.97277650 -2.722 0.00649 **
## L 0.47533570 0.19060262 2.494 0.01264 *
## I(L^2) -0.00623093 0.00292930 -2.127 0.03341 *
## I(L^3) 0.00002659 0.00001440 1.846 0.06482 .
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 57.182 on 17 degrees of freedom
## Residual deviance: 35.140 on 14 degrees of freedom
## AIC: 91.578
##
## Number of Fisher Scoring iterations: 5
##
## Summary of bootstrap fits:
## mean q.0. q.5.
## Min. :0.1454 Min. :0.00000000 Min. :0.00000
## 1st Qu.:0.3799 1st Qu.:0.00000016 1st Qu.:0.08424
## Median :0.6177 Median :0.01911270 Median :0.17290
## Mean :0.5370 Mean :0.05469790 Mean :0.20649
## 3rd Qu.:0.6726 3rd Qu.:0.08900865 3rd Qu.:0.33345
## Max. :0.8516 Max. :0.20218859 Max. :0.48321
## q.25. q.50. q.75. q.95.
## Min. :0.0001559 Min. :0.008919 Min. :0.1531 Min. :0.4712

```

```

## 1st Qu.:0.2161372 1st Qu.:0.367903 1st Qu.:0.5233 1st Qu.:0.8177
## Median :0.3126480 Median :0.694412 Median :0.7902 Median :0.8880
## Mean :0.3736615 Mean :0.566555 Mean :0.6763 Mean :0.8343
## 3rd Qu.:0.5809514 3rd Qu.:0.740554 3rd Qu.:0.8384 3rd Qu.:0.9259
## Max. :0.8319384 Max. :0.950403 Max. :0.9892 Max. :0.9995
## q.100.
## Min. :0.8515
## 1st Qu.:0.9688
## Median :0.9871
## Mean :0.9715
## 3rd Qu.:0.9988
## Max. :1.0000
##
## ***** NORTHERN ANCHOVY *****
##
## Wilcoxon rank sum test with continuity correction
##
## data: .x and .y
## W = 24497, p-value = 3.043e-09
## alternative hypothesis: true location shift is not equal to 0
##
##
## Two-sample Kolmogorov-Smirnov test
##
## data: .x and .y
## D = 0.32536, p-value = 6.01e-09
## alternative hypothesis: two-sided
##
## Summary of GAM fit:
##
## Call:
## glm(formula = p.L12 ~ L + I(L^2) + I(L^3), family = binomial,
## weights = wts)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -1.58746 -1.04605 -0.00453 1.13477 1.85831
##
## Coefficients:
## Estimate Std. Error z value Pr(>|z|)
## (Intercept) -106.87895097 145.70040268 -0.734 0.463
## L 2.15694582 3.19635961 0.675 0.500
## I(L^2) -0.01426172 0.02328293 -0.613 0.540
## I(L^3) 0.00003143 0.00005632 0.558 0.577
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 26.876 on 11 degrees of freedom
## Residual deviance: 18.727 on 8 degrees of freedom
## AIC: 54.403
##
## Number of Fisher Scoring iterations: 4
##
## Summary of bootstrap fits:

```

```

##          mean          q.0.          q.5.          q.25.
## Min.    :0.3842   Min.    :0.000000   Min.    :0.0000092   Min.    :0.02057
## 1st Qu.:0.5985   1st Qu.:0.000000   1st Qu.:0.0890132   1st Qu.:0.42845
## Median :0.7971   Median :0.003532   Median :0.2755877   Median :0.77532
## Mean    :0.7210   Mean    :0.086990   Mean    :0.3127651   Mean    :0.59951
## 3rd Qu.:0.8845   3rd Qu.:0.189695   3rd Qu.:0.5666291   3rd Qu.:0.84124
## Max.    :0.9059   Max.    :0.301063   Max.    :0.7013107   Max.    :0.86644
##          q.50.          q.75.          q.95.          q.100.
## Min.    :0.2615   Min.    :0.6849   Min.    :0.8434   Min.    :1
## 1st Qu.:0.6570   1st Qu.:0.7986   1st Qu.:0.9491   1st Qu.:1
## Median :0.8611   Median :0.9295   Median :0.9923   Median :1
## Mean    :0.7687   Mean    :0.8885   Mean    :0.9650   Mean    :1
## 3rd Qu.:0.9548   3rd Qu.:0.9960   3rd Qu.:1.0000   3rd Qu.:1
## Max.    :0.9784   Max.    :1.0000   Max.    :1.0000   Max.    :1
##
## ***** PACIFIC HERRING *****
##
## Insufficient data
##
## ***** SEA NETTLE *****
##
## Wilcoxon rank sum test with continuity correction
##
## data: .x and .y
## W = 161020, p-value = 0.9677
## alternative hypothesis: true location shift is not equal to 0
##
##
## Two-sample Kolmogorov-Smirnov test
##
## data: .x and .y
## D = 0.064328, p-value = 0.1919
## alternative hypothesis: two-sided
##
## Summary of GAM fit:
##
## Call:
## glm(formula = p.L12 ~ L + I(L^2) + I(L^3), family = binomial,
##      weights = wts)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.3007  -0.8560   0.1096   0.9071   2.5380
##
## Coefficients:
##              Estimate      Std. Error z value Pr(>|z|)
## (Intercept)  0.870833569  0.8311327335   1.048   0.295
## L           -0.0140396513  0.0180557455  -0.778   0.437
## I(L^2)        0.0000232098  0.0001210210   0.192   0.848
## I(L^3)        0.0000001328  0.0000002502   0.531   0.596
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 113.782  on 55  degrees of freedom

```

```

## Residual deviance: 69.779 on 52 degrees of freedom
## AIC: 232.81
##
## Number of Fisher Scoring iterations: 4
##
## Summary of bootstrap fits:
##      mean      q.0.      q.5.      q.25.
## Min.   :0.4439   Min.   :0.0000203   Min.   :0.1844   Min.   :0.4034
## 1st Qu.:0.4660   1st Qu.:0.0459369   1st Qu.:0.3536   1st Qu.:0.4273
## Median :0.5235   Median :0.2253066   Median :0.3718   Median :0.4585
## Mean   :0.5835   Mean   :0.1653057   Mean   :0.3633   Mean   :0.5444
## 3rd Qu.:0.6903   3rd Qu.:0.2600649   3rd Qu.:0.3890   3rd Qu.:0.6492
## Max.   :0.8788   Max.   :0.2991730   Max.   :0.4222   Max.   :0.9032
##      q.50.      q.75.      q.95.      q.100.
## Min.   :0.4423   Min.   :0.4809   Min.   :0.5434   Min.   :0.6137
## 1st Qu.:0.4662   1st Qu.:0.5060   1st Qu.:0.5656   1st Qu.:0.6523
## Median :0.5229   Median :0.5738   Median :0.6593   Median :0.8163
## Mean   :0.6013   Mean   :0.6477   Mean   :0.7184   Mean   :0.8191
## 3rd Qu.:0.7146   3rd Qu.:0.7688   3rd Qu.:0.8592   3rd Qu.:0.9969
## Max.   :0.9745   Max.   :0.9926   Max.   :0.9994   Max.   :1.0000
##
## ***** WATER JELLY *****
##
## Wilcoxon rank sum test with continuity correction
##
## data: .x and .y
## W = 30119, p-value = 0.0192
## alternative hypothesis: true location shift is not equal to 0
##
##
## Two-sample Kolmogorov-Smirnov test
##
## data: .x and .y
## D = 0.099297, p-value = 0.1566
## alternative hypothesis: two-sided
##
## Summary of GAM fit:
##
## Call:
## glm(formula = p.L12 ~ L + I(L^2) + I(L^3), family = binomial,
##      weights = wts)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.15870  -0.60214   0.04482   0.72553   1.81500
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  5.73520424  2.98626868   1.921  0.0548 .
## L           -0.29910317  0.15325488  -1.952  0.0510 .
## I(L^2)        0.00479145  0.00251495   1.905  0.0568 .
## I(L^3)       -0.00002448  0.00001324  -1.849  0.0644 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 29.520  on 19  degrees of freedom
## Residual deviance: 23.455  on 16  degrees of freedom
## AIC: 82.13
##
## Number of Fisher Scoring iterations: 5
##
## Summary of bootstrap fits:
##      mean          q.0.          q.5.
## Min.   :0.08564   Min.   :0.00000   Min.   :0.000000
## 1st Qu.:0.23021   1st Qu.:0.00000   1st Qu.:0.001156
## Median :0.43963   Median :0.02536   Median :0.218125
## Mean   :0.36922   Mean   :0.04753   Mean   :0.156761
## 3rd Qu.:0.46346   3rd Qu.:0.09677   3rd Qu.:0.249615
## Max.   :0.68159   Max.   :0.13608   Max.   :0.344979
##      q.25.          q.50.          q.75.          q.95.
## Min.   :0.0000058   Min.   :0.001015   Min.   :0.0234   Min.   :0.6083
## 1st Qu.:0.0681473   1st Qu.:0.186907   1st Qu.:0.3256   1st Qu.:0.6566
## Median :0.3457250   Median :0.443058   Median :0.5213   Median :0.6773
## Mean   :0.2599618   Mean   :0.348806   Mean   :0.4443   Mean   :0.7106
## 3rd Qu.:0.3666425   3rd Qu.:0.465834   3rd Qu.:0.5575   3rd Qu.:0.7336
## Max.   :0.5528675   Max.   :0.711209   Max.   :0.8316   Max.   :0.9501
##      q.100.
## Min.   :0.7591
## 1st Qu.:0.8330
## Median :0.9331
## Mean   :0.9108
## 3rd Qu.:0.9997
## Max.   :1.0000
##
## ***** Excluder:  Down *****
## Total Adjusted Catch:
##      CHINOOK SALMON      CHUM SALMON      COHO SALMON      MARKET SQUID
##      1002.0864          255.3848          155.4833          17572.2192
## NORTHERN ANCHOVY  PACIFIC HERRING      SEA NETTLE      WATER JELLY
##      3.0000          2400.1818          47.0000          183119.7267
##
## ***** CHINOOK SALMON *****
##
## Wilcoxon rank sum test with continuity correction
##
## data:  .x and .y
## W = 34425, p-value = 0.000005302
## alternative hypothesis: true location shift is not equal to 0
##
##
## Two-sample Kolmogorov-Smirnov test
##
## data:  .x and .y
## D = 0.17929, p-value = 0.0001577
## alternative hypothesis: two-sided
##

```

```

## Summary of GAM fit:
##
## Call:
## glm(formula = p.L12 ~ L + I(L^2) + I(L^3), family = binomial,
##      weights = wts)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.27913  -1.05892   0.02342   0.95216   1.90680
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.645e+00  1.030e+00   1.597   0.1102
## L           -2.376e-02  1.329e-02  -1.787   0.0739 .
## I(L^2)       8.571e-05  4.855e-05   1.766   0.0775 .
## I(L^3)      -8.979e-08  5.197e-08  -1.728   0.0840 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 79.036  on 74  degrees of freedom
## Residual deviance: 74.352  on 71  degrees of freedom
## AIC: 175.45
##
## Number of Fisher Scoring iterations: 5
##
## Summary of bootstrap fits:
##      mean      q.0.      q.5.
## Min.   :0.1179   Min.   :0.00000000   Min.   :0.00000
## 1st Qu.:0.3515   1st Qu.:0.00002495   1st Qu.:0.04395
## Median :0.4375   Median :0.05867008   Median :0.21848
## Mean   :0.4082   Mean   :0.07143674   Mean   :0.18835
## 3rd Qu.:0.5201   3rd Qu.:0.12710540   3rd Qu.:0.29302
## Max.   :0.6068   Max.   :0.26964195   Max.   :0.40400
##      q.25.      q.50.      q.75.      q.95.
## Min.   :0.0000628   Min.   :0.002678   Min.   :0.04924   Min.   :0.6086
## 1st Qu.:0.1869646   1st Qu.:0.336251   1st Qu.:0.47913   1st Qu.:0.6698
## Median :0.3411622   Median :0.433289   Median :0.52974   Median :0.7643
## Mean   :0.2951066   Mean   :0.384269   Mean   :0.48908   Mean   :0.7509
## 3rd Qu.:0.4251438   3rd Qu.:0.517400   3rd Qu.:0.61758   3rd Qu.:0.7995
## Max.   :0.5222787   Max.   :0.610982   Max.   :0.69462   Max.   :0.9649
##      q.100.
## Min.   :0.8228
## 1st Qu.:0.8932
## Median :0.9509
## Mean   :0.9385
## 3rd Qu.:0.9995
## Max.   :1.0000
##
## ***** CHUM SALMON *****
##
## Wilcoxon rank sum test with continuity correction
##

```

```

## data: .x and .y
## W = 3579, p-value = 0.1798
## alternative hypothesis: true location shift is not equal to 0
##
##
## Two-sample Kolmogorov-Smirnov test
##
## data: .x and .y
## D = 0.14104, p-value = 0.4074
## alternative hypothesis: two-sided
##
## Summary of GAM fit:
##
## Call:
## glm(formula = p.L12 ~ L + I(L^2) + I(L^3), family = binomial,
##      weights = wts)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.3788  -0.6636   0.1496   0.5312   1.9372
##
## Coefficients:
##              Estimate      Std. Error z value Pr(>|z|)
## (Intercept) -15.301418766    67.256511774  -0.228    0.820
## L              0.324906462     1.156819324   0.281    0.779
## I(L^2)        -0.002174849     0.006600616  -0.329    0.742
## I(L^3)         0.000004691     0.000012502   0.375    0.707
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 14.167  on 14  degrees of freedom
## Residual deviance: 12.646  on 11  degrees of freedom
## AIC: 55.107
##
## Number of Fisher Scoring iterations: 6
##
## Summary of bootstrap fits:
##      mean      q.0.      q.5.      q.25.
## Min.   :0.5513  Min.   :0.000000  Min.   :0.000000  Min.   :0.4327
## 1st Qu.:0.6296  1st Qu.:0.000000  1st Qu.:0.001596  1st Qu.:0.4798
## Median :0.6931  Median :0.001589  Median :0.292737  Median :0.5178
## Mean   :0.6727  Mean   :0.104001  Mean   :0.242858  Mean   :0.5457
## 3rd Qu.:0.7157  3rd Qu.:0.253400  3rd Qu.:0.410867  3rd Qu.:0.5971
## Max.   :0.7554  Max.   :0.381989  Max.   :0.544924  Max.   :0.7877
##      q.50.      q.75.      q.95.      q.100.
## Min.   :0.5517  Min.   :0.6072  Min.   :0.6909  Min.   :0.8799
## 1st Qu.:0.6511  1st Qu.:0.7022  1st Qu.:0.8020  1st Qu.:0.9439
## Median :0.7066  Median :0.8982  Median :0.9889  Median :1.0000
## Mean   :0.7737  Mean   :0.8509  Mean   :0.9044  Mean   :0.9730
## 3rd Qu.:0.9530  3rd Qu.:1.0000  3rd Qu.:1.0000  3rd Qu.:1.0000
## Max.   :0.9993  Max.   :1.0000  Max.   :1.0000  Max.   :1.0000
##
## ***** COHO SALMON *****
##

```

```

## Wilcoxon rank sum test with continuity correction
##
## data: .x and .y
## W = 2252.5, p-value = 0.01996
## alternative hypothesis: true location shift is not equal to 0
##
##
## Two-sample Kolmogorov-Smirnov test
##
## data: .x and .y
## D = 0.31455, p-value = 0.001098
## alternative hypothesis: two-sided
##
## Summary of GAM fit:
##
## Call:
## glm(formula = p.L12 ~ L + I(L^2) + I(L^3), family = binomial,
##      weights = wts)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.8298  -0.7191  -0.3292   0.5503   2.4104
##
## Coefficients:
##              Estimate      Std. Error z value Pr(>|z|)
## (Intercept) -3.5712824495   3.4261844962  -1.042   0.2972
## L             0.0432693656   0.0322471088   1.342   0.1797
## I(L^2)       -0.0001484696   0.0000864551  -1.717   0.0859 .
## I(L^3)        0.0000001400   0.0000000711   1.969   0.0490 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 69.991  on 56  degrees of freedom
## Residual deviance: 55.305  on 53  degrees of freedom
## AIC: 104.59
##
## Number of Fisher Scoring iterations: 4
##
## Summary of bootstrap fits:
##      mean      q.0.      q.5.      q.25.
## Min.   :0.1682  Min.   :0.01635  Min.   :0.05331  Min.   :0.1098
## 1st Qu.:0.2191  1st Qu.:0.02205  1st Qu.:0.07360  1st Qu.:0.1421
## Median :0.3848  Median :0.03264  Median :0.14833  Median :0.2649
## Mean   :0.3897  Mean   :0.06204  Mean   :0.19165  Mean   :0.2968
## 3rd Qu.:0.5542  3rd Qu.:0.06221  3rd Qu.:0.28466  3rd Qu.:0.4480
## Max.   :0.6687  Max.   :0.25861  Max.   :0.46192  Max.   :0.5556
##      q.50.      q.75.      q.95.      q.100.
## Min.   :0.1557  Min.   :0.2131  Min.   :0.3225  Min.   :0.5047
## 1st Qu.:0.2050  1st Qu.:0.2759  1st Qu.:0.4195  1st Qu.:0.7169
## Median :0.3631  Median :0.4832  Median :0.6920  Median :0.8881
## Mean   :0.3788  Mean   :0.4730  Mean   :0.6204  Mean   :0.8227
## 3rd Qu.:0.5545  3rd Qu.:0.6509  3rd Qu.:0.7879  3rd Qu.:0.9530

```



```

## Max. :0.6889 Max. :0.8373 Max. :0.9785 Max. :1.0000
##
## ***** MARKET SQUID *****
##
## Wilcoxon rank sum test with continuity correction
##
## data: .x and .y
## W = 77344, p-value = 0.003271
## alternative hypothesis: true location shift is not equal to 0
##
## Two-sample Kolmogorov-Smirnov test
##
## data: .x and .y
## D = 0.12, p-value = 0.009663
## alternative hypothesis: two-sided
##
## Summary of GAM fit:
##
## Call:
## glm(formula = p.L12 ~ L + I(L^2) + I(L^3), family = binomial,
## weights = wts)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.71791  -0.94032   0.00664   0.87627   1.82742
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.730989505  1.529021884  -1.132   0.2576
## L             0.118148391  0.067601893   1.748   0.0805
## I(L^2)       -0.001458737  0.000946183  -1.542   0.1231
## I(L^3)        0.000005212  0.000004133   1.261   0.2074
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 38.511  on 24  degrees of freedom
## Residual deviance: 33.295  on 21  degrees of freedom
## AIC: 97.49
##
## Number of Fisher Scoring iterations: 4
##
## Summary of bootstrap fits:
##      mean      q.0.      q.5.      q.25.
## Min. :0.2096 Min. :0.000000 Min. :0.001043 Min. :0.03466
## 1st Qu.:0.5019 1st Qu.:0.000000 1st Qu.:0.040445 1st Qu.:0.33205
## Median :0.6064 Median :0.001539 Median :0.219507 Median :0.43500
## Mean :0.5916 Mean :0.126251 Mean :0.277480 Mean :0.46507
## 3rd Qu.:0.7063 3rd Qu.:0.264262 3rd Qu.:0.513869 3rd Qu.:0.64107
## Max. :0.7810 Max. :0.495004 Max. :0.666042 Max. :0.73809
##      q.50.      q.75.      q.95.      q.100.
## Min. :0.1250 Min. :0.3263 Min. :0.6753 Min. :0.9091

```

```

## 1st Qu.:0.5126 1st Qu.:0.6332 1st Qu.:0.7759 1st Qu.:0.9313
## Median :0.6362 Median :0.7462 Median :0.8542 Median :0.9523
## Mean :0.6104 Mean :0.7308 Mean :0.8522 Mean :0.9591
## 3rd Qu.:0.7470 3rd Qu.:0.8250 3rd Qu.:0.8972 3rd Qu.:0.9972
## Max. :0.8868 Max. :0.9966 Max. :1.0000 Max. :1.0000
##
## ***** PACIFIC HERRING *****
##
## Insufficient data
##
## ***** WATER JELLY *****
##
## Wilcoxon rank sum test with continuity correction
##
## data: .x and .y
## W = 79448, p-value = 0.8924
## alternative hypothesis: true location shift is not equal to 0
##
##
## Two-sample Kolmogorov-Smirnov test
##
## data: .x and .y
## D = 0.025505, p-value = 0.9995
## alternative hypothesis: two-sided
##
## Summary of GAM fit:
##
## Call:
## glm(formula = p.L12 ~ L + I(L^2) + I(L^3), family = binomial,
## weights = wts)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -3.5114 -0.5073 0.2832 0.9065 2.5335
##
## Coefficients:
## Estimate Std. Error z value Pr(>|z|)
## (Intercept) -7.018755017 3.943695235 -1.780 0.0751 .
## L 0.272045342 0.159724393 1.703 0.0885 .
## I(L^2) -0.003280210 0.002107257 -1.557 0.1196
## I(L^3) 0.000012696 0.000009052 1.403 0.1607
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 44.978 on 19 degrees of freedom
## Residual deviance: 40.236 on 16 degrees of freedom
## AIC: 110.61
##
## Number of Fisher Scoring iterations: 3
##
## Summary of bootstrap fits:
## mean q.0. q.5.

```

##	Min.	:0.2030	Min.	:0.000000	Min.	:0.0000742		
##	1st Qu.	:0.4954	1st Qu.	:0.000062	1st Qu.	:0.0364239		
##	Median	:0.5445	Median	:0.035775	Median	:0.2379796		
##	Mean	:0.5047	Mean	:0.079928	Mean	:0.2138525		
##	3rd Qu.	:0.5895	3rd Qu.	:0.140651	3rd Qu.	:0.3689207		
##	Max.	:0.6360	Max.	:0.235646	Max.	:0.4372077		
##	q.25.		q.50.		q.75.		q.95.	
##	Min.	:0.008695	Min.	:0.08069	Min.	:0.2884	Min.	:0.6761
##	1st Qu.	:0.329778	1st Qu.	:0.49756	1st Qu.	:0.5944	1st Qu.	:0.7576
##	Median	:0.395274	Median	:0.54995	Median	:0.6472	Median	:0.7980
##	Mean	:0.364505	Mean	:0.51097	Mean	:0.6311	Mean	:0.8094
##	3rd Qu.	:0.478733	3rd Qu.	:0.60298	3rd Qu.	:0.6796	3rd Qu.	:0.8386
##	Max.	:0.534651	Max.	:0.78761	Max.	:0.9442	Max.	:0.9975
##	q.100.							
##	Min.	:0.8592						
##	1st Qu.	:0.9087						
##	Median	:0.9400						
##	Mean	:0.9437						
##	3rd Qu.	:0.9953						
##	Max.	:1.0000						