

Evaluating changes in species distribution in the Eastern Bering Sea internal report

Daniel Vilas Lewis A.K. Barnett André E. Punt Zack S. Oyafuso
Lukas DeFilippo Margaret C. Siple Shannon Hennessey Leah S. Zacher
Stan Kotwicki

2024-09-10

Setup and settings

These lines of code set the working directory and load necessary libraries (installing them if needed).

```

#####
# Settings
#####

#libraries from cran to call or install/load
pack_cran<-c('shadowtext',
             'cowplot',
             'ggspatial',
             'raster',
             'rasterVis',
             'rgeos',
             'scales',
             'rnatural-earth',
             'grid',
             'ggplot2',
             'lubridate',
             'ragg',
             'rgdal',
             'dplyr',
             'sp',
             'ggtext',
             'magick',
             'shadowtext',
             'dplyr',
             'googledrive',
             'sf',
             'knitr',
             'data.table',
             'ggshadow',
             'devtools',
             'reshape2')

#install pacman to use p_load function - call library and if not installed, then install

```

```

if (!('pacman' %in% installed.packages())) {
  install.packages("pacman");library(pacman)} else {
  library(pacman)
}

#install akgfmaps to extract shapefile of Alaska
if (!('akgfmaps' %in% installed.packages())) {
  install_github('afsc-gap-products/akgfmaps');library(akgfmaps)

#install VAST
if (!('VAST' %in% installed.packages())) {
  install_github("james-thorson/VAST@main", INSTALL_opts="--no-staged-install");library(VAST)

#load/install packages
p_load(pack_cran,character.only = TRUE)

# Set the root directory for knitting to a work directory
dir <- '/Users/daniel/Work/Adapting Monitoring to a Changing Seascape/'
opts_knit$set(root.dir = dir)

```

Introduction

This report describes the analysis of species distribution in the Eastern Bering Sea (EBS) shelf, EBS slope, and Northern Bering Sea (NBS). We used observed and simulated observations from VAST models to explore various metrics and test the hypothesis that species shift their distribution during warm years to occupy northern and deeper regions in the area.

Methods

Observed data come from the fishery-independent bottom-trawl survey. Predicted data are derived from the fitted single-species VAST model, a Generalized Linear Mixed Model that represents the spatio-temporal distribution of species. We focused on four species to explore changes in their distribution: Pacific cod (*Gadus macrocephalus*), Alaskan pollock (*Gadus chalcogrammus*), snow crab (*Chionoecetes opilio*), and halibut (*Reinhardtius hippoglossoides*). We examined various representations of biomass (kg), density (kg/km²), and multiple environmental factors (depth, latitude, longitude), as well as prey data and metrics.

The metrics used in this analysis include:

Center of gravity

The equation for the center of gravity *COG* (x,y) or (longitude,latitude) of a species is given by:

$$COG = \left(\frac{\sum_{i=1}^n dens_i lon_i}{\sum_{i=1}^n dens_i}, \frac{\sum_{i=1}^n dens_i lat_i}{\sum_{i=1}^n dens_i} \right)$$

where: $dens_i$ is the density of the i -th observation, lon_i is the longitude of the i -th observation, lat_i is the latitude of the i -th observation, n is the total number of observations.

Bathymetrical center of gravity

The equation for the center of the bathymetrical center of gravity (COGdepth) of a species is given by:

$$COGdepth = \frac{\sum_{i=1}^n dens_i depth_i}{\sum_{i=1}^n dens_i}$$

where: $depth_i$ is the depth of the i -th observation

Effective area occupied

The equation for the effective area occupied EAO of a species is given by:

$$EAO = \frac{\text{total abundance}}{\text{mean density}}$$

where: $depth_i$ is the depth of the i -th observation

Results

Annual SBT values were extracted as the mean SBT value over cells from the ROMS Bering-10K model extrapolation grid. Annual mean SBT values were computed for multiple regions: EBS shelf, EBS slope, NBS and all (EBSshelf+EBSslope+NBS). We scaled annual SBT estimates to identify years warm and cold years as well as warm and cold periods. Identified cold and warm years were years with one standard deviation from the mean value. Cold and warm labeling was compared to previous temperature analysis in the area (Siple et al. unpublished) (2002-2005 warm; 2006-2013 cold; 2014:2022 warm)

```
#setwd
setwd(dir)

# Define the file ID from the shareable link
file_id <- "1WgzeMc47A13zwtuz_o0L6V3H6mImSbYt"

# Download the grid file
suppressMessages(
  drive_download(as_id(file_id), path = "./grid_EBS_NBS.RData", overwrite = TRUE))

#file_url <- "https://drive.google.com/uc?export=download&id=FILE_ID"
#download.file(file_url, destfile = "./grid_EBS_NBS.RData")

#load file grid
load('./grid_EBS_NBS.RData') #grid.ebs_year

#average SBT by region and year
temp_region<-aggregate(Temp ~ Year + region, grid.ebs_year, FUN=mean)
names(temp_region)<-c('year','region','temp')

#get average SBT for all region (EBSshelf+NBS+EBSslope)
alltemp<-aggregate(temp ~ year, temp_region, FUN=mean)
alltemp$region<-'all'
alltemp<-alltemp[,c("year","region","temp")]
```

```

#rbind
temp_region<-rbind(temp_region,alltemp)

# Scaling values by group
temp_region1 <- data.frame(temp_region %>%
                           group_by(region) %>%
                           mutate(scaledtemp = scale(temp)) %>%
                           ungroup())

#classify by cold/warm/normal years
temp_region1$typeyear <- ifelse(temp_region1$scaledtemp > 1,
                                 "warm", ifelse(temp_region1$scaledtemp < -1,
                                                "cold", "normal"))

#cold/warm years
crys<-unique(temp_region1[which(temp_region1$typeyear == c('cold') &
                                 temp_region1$region=='all'), 'year'])
wyrs<-unique(temp_region1[which(temp_region1$typeyear == c('warm') &
                                 temp_region1$year %in% c(1982:2022) &
                                 temp_region1$region=='all'), 'year'])

# Define the years you want to bold
bold_years <- c(2002, 2004, 2008, 2010, 2012, 2016)

# Create custom labels function
custom_labels <- function(x) {
  sapply(x, function(year) {
    if (year %in% bold_years) {
      paste0("##", year, "##")
    } else {
      as.character(year)
    }
  })
}

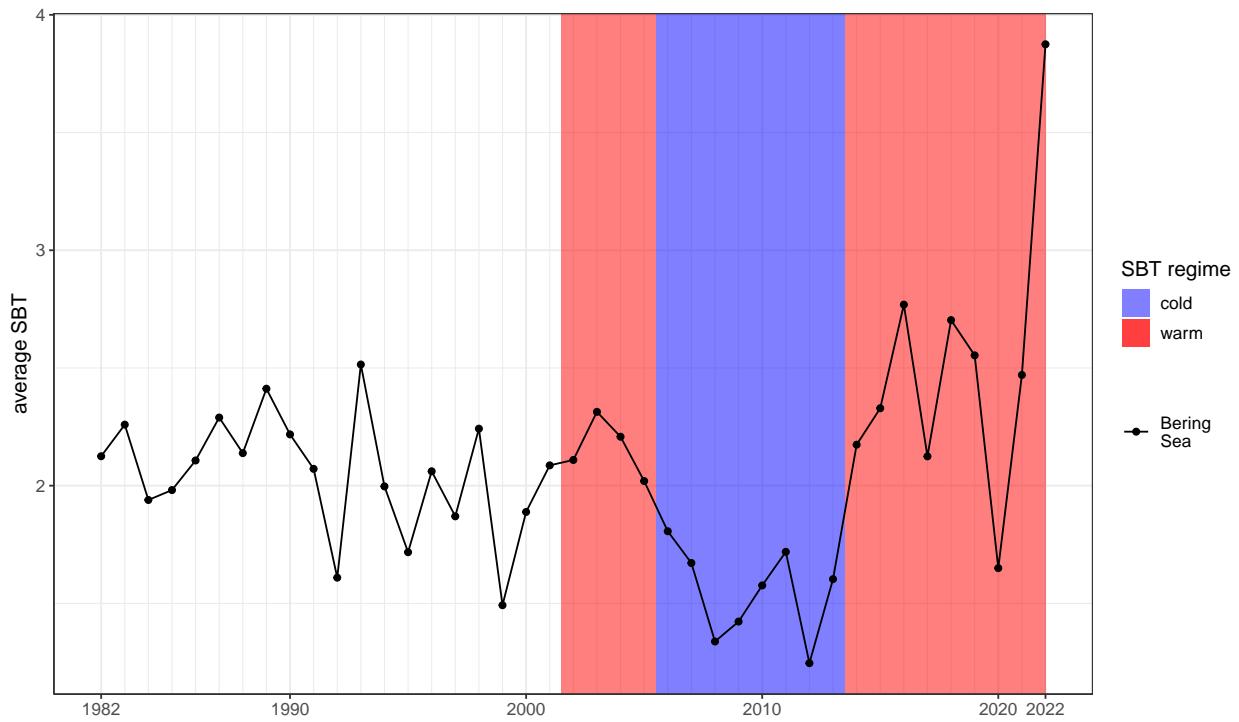
#plot
print(
ggplot() +
  geom_rect(aes(xmin = 2001.5, xmax = 2005.5, ymin = -Inf, ymax = Inf, fill = "red"),
            alpha = 0.5) +
  geom_rect(aes(xmin = 2013.5, xmax = 2022, ymin = -Inf, ymax = Inf, fill = "red"),
            alpha = 0.5) +
  geom_rect(aes(xmin = 2005.5, xmax = 2013.5, ymin = -Inf, ymax = Inf, fill = "blue"),
            alpha = 0.5) +
  geom_point(data = subset(temp_region1, region == 'all' & year %in% c(1982:2022)),
             aes(x = year, y = temp, color = region)) +
  geom_line(data = subset(temp_region1, region == 'all' & year %in% c(1982:2022)),
            aes(x = year, y = temp, color = region)) +
  scale_color_manual(values = c('all' = 'black'),
                     breaks = c('all'),
                     labels = c('Bering\nSea', name='')) +
  scale_fill_manual(values = c("red" = "red", "blue" = "blue"),
                    labels = c("cold", "warm"))

```

```

        name = "SBT regime") +
theme_bw() +
scale_x_continuous(limits = c(1982, 2022), minor_breaks = c(1982:2022),
                   breaks = c(1982,1990,2000,2010,2020,2022)) +
theme(text = element_text(size = 12)) +
labs(x = '', y = 'average SBT')
)

```



```

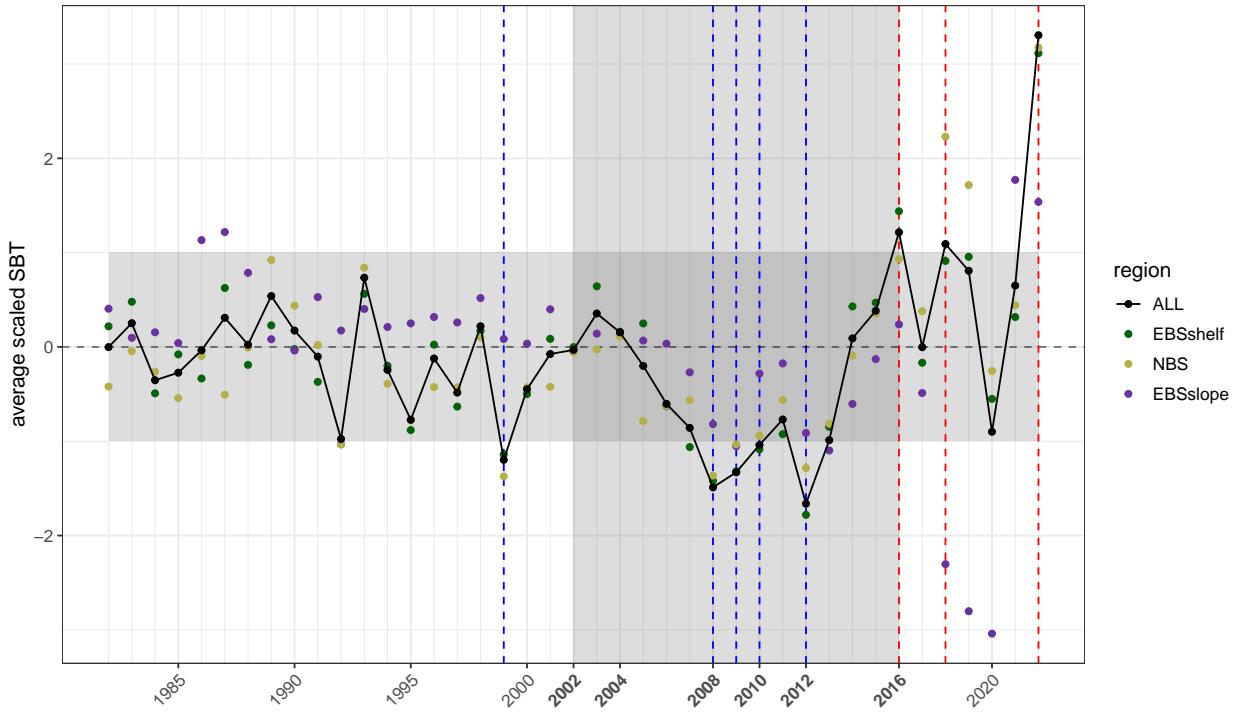
#plot
print(
ggplot() +
geom_rect(aes(xmin = 1982, xmax = 2022, ymin = -1, ymax = 1),
           linetype = 'dashed', alpha = 0.2) +
geom_rect(aes(xmin = 2002, xmax = 2016, ymin = -Inf, ymax = Inf),
           alpha = 0.2) +
geom_vline(xintercept = cyrs, linetype = 'dashed', color = 'blue') +
geom_vline(xintercept = wyrs, linetype = 'dashed', color = 'red') +
geom_hline(yintercept = 0, alpha = 0.5, linetype = 'dashed') +
geom_point(data = subset(temp_region1, year %in% 1982:2022) ,
           aes(x = year, y = scaledtemp, color = region)) +
geom_line(data = subset(temp_region1, region == 'all' & year %in% 1982:2022),
           aes(x = year, y = scaledtemp, color = region)) +
scale_color_manual(values = c('EBSslope' = '#6a329f',
                             'EBSshelf' = '#046407',
                             'NBS' = '#B4AF46',
                             'EBS+NBS' = '#',
                             'all' = 'black'),
                     breaks = c('all', 'EBS+NBS', 'EBSshelf', 'NBS', 'EBSslope'),
                     labels = c('ALL', 'EBS+NBS', 'EBSshelf', 'NBS', 'EBSslope')) +

```

```

theme_bw() +
scale_x_continuous(breaks = c(1985, 1990, 1995, 2000, bold_years, 2020),
                   minor_breaks = c(1982:2022),
                   labels = custom_labels, limits = c(1982, 2022)) +
theme(axis.text.x = element_markdown(angle = 45, hjust=1),
      axis.text.y = element_text(), text = element_text(size = 12)) +
labs(x = '', y = 'average scaled SBT')
)

```



Average scaled annual SBT across regions. Horizontal rectangular shade represents the 1 standard deviation that includes normal SBT estimates. Vertical rectangular shade represents the time period in which slope model was fitted. Bold x-axis text indicates years with slope observed data. Dashed blue lines represent cold years and dashed red lines represent warm years. Point colors represent regions.

Metrics

- Note that this analysis doesn't account for catchability differences between the EBS+NBS shelf survey data and EBS slope survey data, which are being analyzed in Rohan *et al.* (in progress).

Observed CPUE

Observed catch per unit effort (CPUE) data by year and region for selected species.

```

# Define the files
file_id <- '1B0mhHm7j9VCILPAhgs_aPAmnKMi2rfp'

# Download observed data file
suppressMessages(

```

```

drive_download(as_id(file_id), path = "./obs_depth_density.RData", overwrite = TRUE)

#load file observed dataframe input
load('./obs_depth_density.RData')

#survey as factors and rename
obs_df$survey_name<-as.factor(obs_df$survey_name)
levels(obs_df$survey_name)<-c('EBSshelf','EBSslope','NBS')

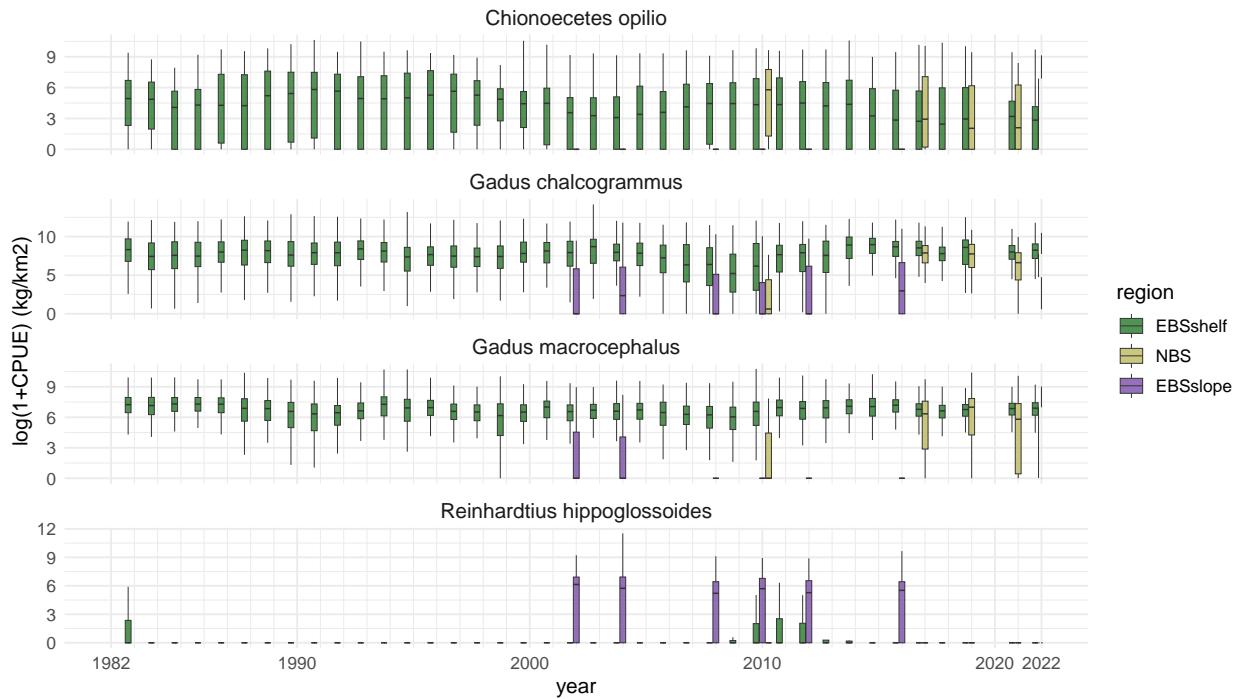
#selected species
sel_sp<-c("Gadus chalcogrammus", #Alaskan pollock
          "Gadus macrocephalus", #pacific cod
          "Reinhardtius hippoglossoides", #Greenland turbot
          "Chionoecetes opilio") #snow crab

#filter by selected species
obs_df1<-subset(obs_df,species %in% sel_sp & year %in% c(1982:2022))

#plot obs cpue
suppressWarnings(print(
  ggplot(obs_df1, aes(x = year, y=log(1+cpue_kgkm2),group=interaction(year,survey_name),
                 fill=survey_name)) +
    geom_boxplot(alpha = 0.7,outliers = FALSE,
                 position=position_dodge(width=0.8,preserve = "single"),size=0.2) +
    theme_minimal()+
    labs(y='log(1+CPUE) (kg/km2)',title = 'Observed CPUE data')+
    scale_fill_manual(values = c('EBSshelf' = '#046407',
                                'NBS' = '#B4AF46',
                                'EBSslope'='#6a329f'),
                      breaks = c('EBSshelf', 'NBS','EBSslope'),
                      labels = c('EBSshelf', 'NBS','EBSslope'),name='region') +
    theme(strip.text = element_text(size=12),text= element_text(size=12))+ 
    scale_x_continuous(limits = c(1982, 2022),minor_breaks = c(1982:2022),
                       breaks = c(1982,1990,2000,2010,2020,2022)) +
    facet_wrap(~species,scales='free_y',ncol = 1)
))

```

Observed CPUE data



Empirical depth distribution of occurrence by temperature regime

Species depth distribution of presences across warm and cold years, using observed CPUE data (kg/km^2) for selected species in the EBSshelf and EBSSlope. Only presence data are included.

```
# Define the file ID from the shareable link
file_id <- "1B0mhHHm7j9VCILPAhgs_aPAmnKMi2rfp"

# Download the file
suppressMessages(
  drive_download(as_id(file_id), path = "./obs_df.RData", overwrite = TRUE))

#load file observed dataframe input
load('./obs_df.RData') #obs_df

#survey as factors and rename
obs_df$survey_name<-as.factor(obs_df$survey_name)
levels(obs_df$survey_name)<-c('EBSshelf','EBSSlope','NBS')

#only presences, EBSshelf+EBSSlope and selected species
obs_df<-subset(obs_df, cpue_kgkm2!=0 &
  survey_name %in% c('EBSshelf','EBSSlope') &
  species %in% sel_sp)

#filter by cold and warm years
obs_df1<-obs_df[which(obs_df$year %in% c(cyrs,wyrs)),]

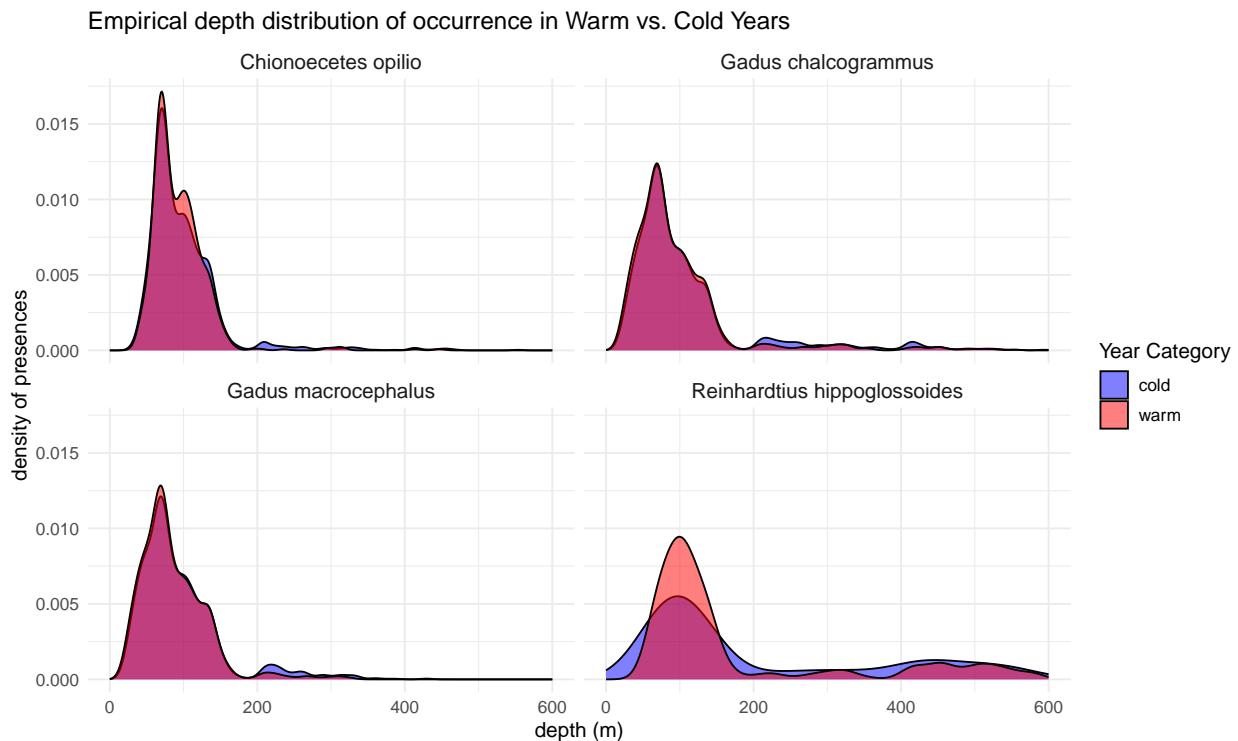
#year type
obs_df1$year_type<-ifelse(obs_df1$year %in% wyrs, "warm", 'cold')
```

```

#filtered to <600m
filt_obs_df1<-subset(obs_df1,depth_m<=600)

#plot Kernel density estimation for depth distributions
print(
ggplot(filt_obs_df1, aes(x = depth_m, fill = year_type)) +
  geom_density(alpha = 0.5) +
  labs(x = 'depth (m)', y = 'density of presences',
       title = 'Empirical depth distribution of occurrence in Warm vs. Cold Years') +
  scale_fill_manual(values = c('warm' = 'red',
                               'cold' = 'blue'), name = 'Year Category') +
  theme_minimal()+
  scale_x_continuous(limits = c(0,600))+ 
  theme(strip.text = element_text(size=12),text= element_text(size=12))+ 
  facet_wrap(~species)
)

```



Observed CPUE by temperature regime and depth bins

CPUE data for the EBSshelf and EBSslope, categorized by depth bins. Data are available only for the years with slope data: 2002, 2004, 2008, 2010, 2012, and 2016.

```

#load file observed dataframe
load('./obs_df.RData') #obs_df

#survey as factors and rename
obs_df$survey_name<-as.factor(obs_df$survey_name)
levels(obs_df$survey_name)<-c('EBSshelf','EBSslope','NBS')

```

```

# create data bins
obs_df$depth_bin<-ifelse(obs_df$depth_m<50,
                           '0-50',
                           ifelse(obs_df$depth_m<100,
                                  '50-100',
                                  ifelse(obs_df$depth_m<150,
                                         '100-150',
                                         ifelse(obs_df$depth_m<200,
                                                '150-200','>200'))))

# yrs with slope observations
slp_yrs<-c(2002,2004,2008,2010,2012,2016)

# EBSshelf and slope and only years with slope data
obs_df1<-subset(obs_df,
                 survey_name %in% c('EBSslope','EBSshelf') &
                   year %in% slp_yrs)

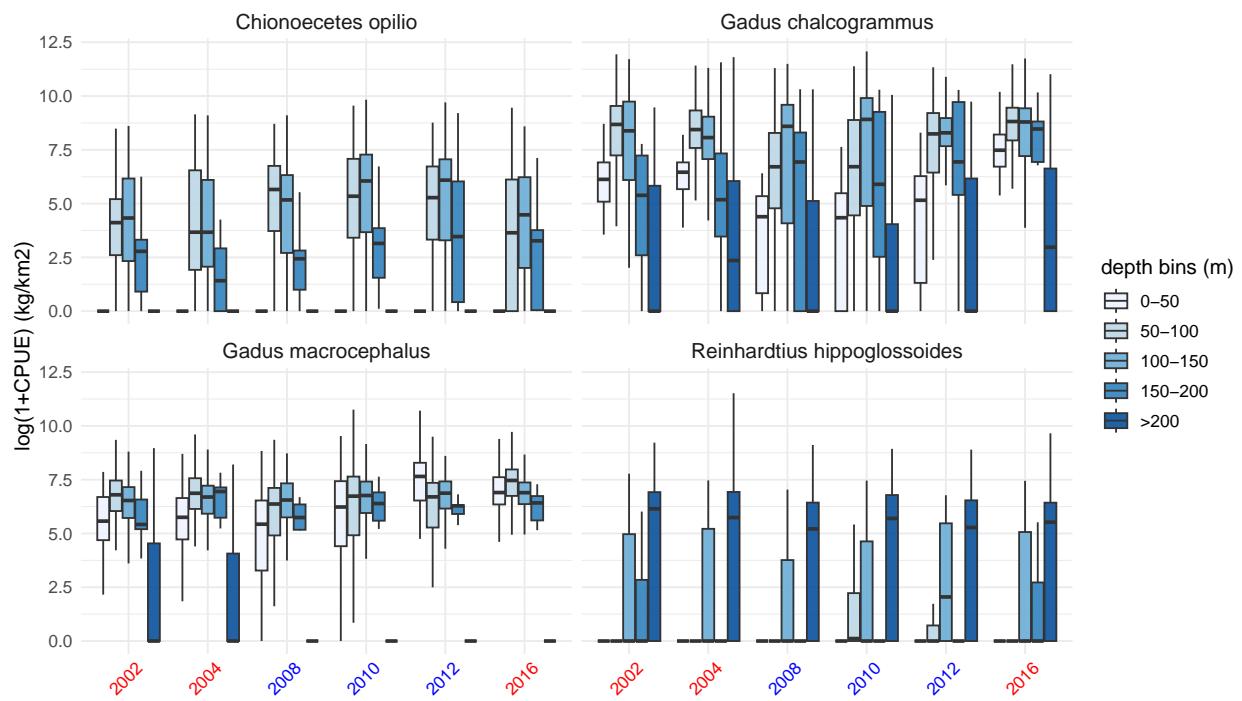
# year type
obs_df1$year_type<-ifelse(obs_df1$year %in% c(2002,2004,2016),
                           "warm",'cold')

# factors
obs_df1$year_type<-as.factor(obs_df1$year_type)
obs_df1$depth_bin<-factor(obs_df1$depth_bin,
                           levels = c("0-50", "50-100", '100-150',"150-200",">200"))
obs_df1$year<-as.factor(obs_df1$year)

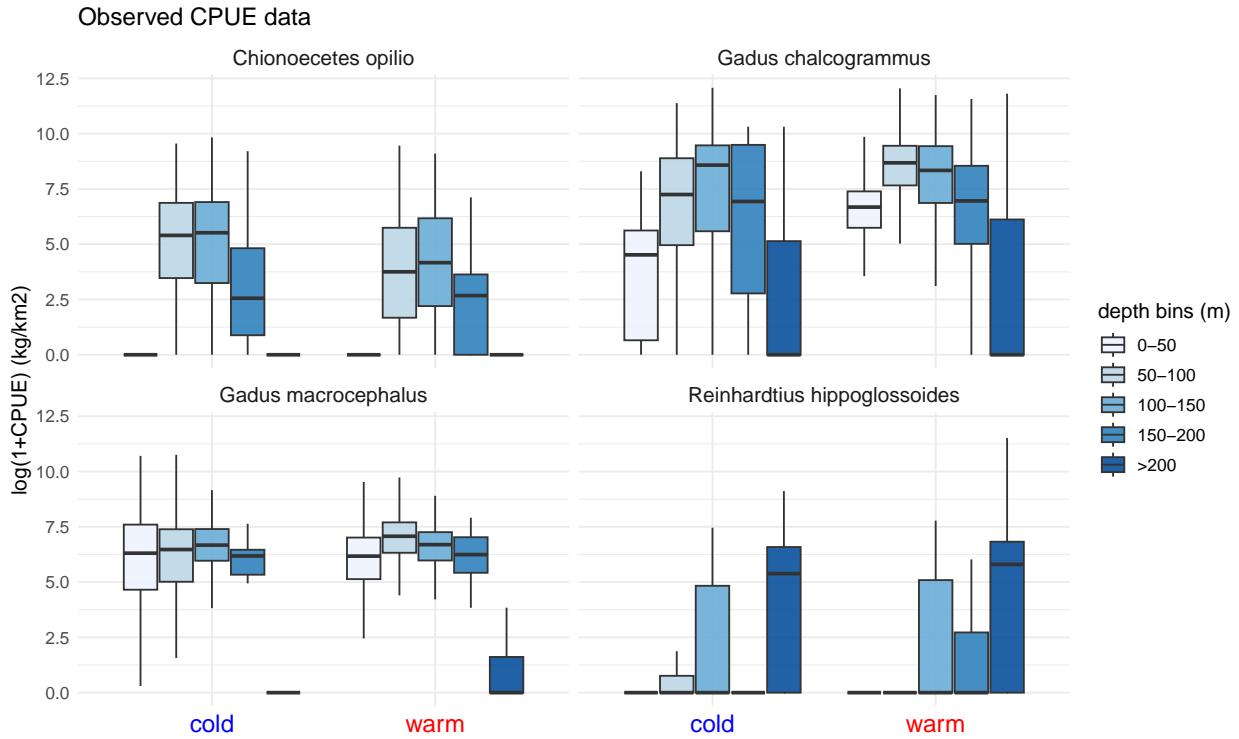
# plot by years
suppressWarnings(print(
  ggplot(data=subset(obs_df1,species %in% sel_sp))+ 
    geom_boxplot(aes(x=year,y=log(1+cpue_kgkm2),
                     fill=depth_bin,group=interaction(year,depth_bin)),
                  position=position_dodge(width=0.8,preserve = "single"),
                  outliers = FALSE,alpha=0.9)+ #outliers = FALSE,
    scale_fill_brewer(name = 'depth bins (m)')+ 
    theme_minimal()+
    theme(strip.text = element_text(size=12),
          axis.title.x = element_blank(),
          axis.text.x = element_text(color = ifelse(levels(obs_df1$year) %in%
                                                       c(2002,2004,2016),
                                                       "red",
                                                       ifelse(levels(obs_df1$year) %in%
                                                               c(2008,2010,2012),
                                                               "blue",'black'))),
          size=10,angle = 45,hjust=0.7,vjust=0.8),
    text = element_text(size = 12))+ 
    labs(y='log(1+CPUE) (kg/km2)',title = 'Observed CPUE data')+ 
    facet_wrap(~species)
))

```

Observed CPUE data



```
#plot by regime
suppressWarnings(print(
ggplot(data=subset(obs_df1,species %in% sel_sp))+ 
  geom_boxplot(aes(x=year_type,y=log(1+cpue_kgkm2),
  fill=depth_bin,group=interaction(year_type,depth_bin)),
  position=position_dodge(width=0.8,preserve = "single"),
  outliers = FALSE,alpha=0.9)+ 
  scale_fill_brewer(name = 'depth bins (m)')+ 
  theme_minimal()+
  labs(y='log(1+CPUE) (kg/km²)',title = 'Observed CPUE data')+ 
  theme(strip.text = element_text(size=12),
  axis.title.x = element_blank(),
  axis.text.x = element_text(color = ifelse(levels(obs_df1$year_type) == 'warm',
  "red",'blue'),size=14),
  text = element_text(size = 12))+ 
  facet_wrap(~species)
))
```



Fraction of total abundance by region

Fraction of total abundance in the Bering shelf area (EBSshelf + NBS) over the years, with colored rectangles representing sea bottom temperature (SBT) regimes: warm and cold.

```
# Define the file ID from the shareable link
file_id <- "11kFQjEqQLPiQ5SoFXnTXoSHP6D_J0k6"

# Download abundance index estimate file
suppressMessages(
  suppressWarnings(
    drive_download(as_id(file_id), path = "./ind_sel.RData", overwrite = TRUE)))

#load file index of selected species
load('./ind_sel.RData') #ind_all

#reshape index
ind_all1<-melt(ind_all,id.vars=c('year','species'))

#keep NBS and EBSshelf
ind_all2<-subset(ind_all1,variable %in% c('NBS','EBSshelf'))

#get fraction
ind_all3<-ind_all2 %>%
  group_by(species, year) %>%
  mutate(frac = value / sum(value))

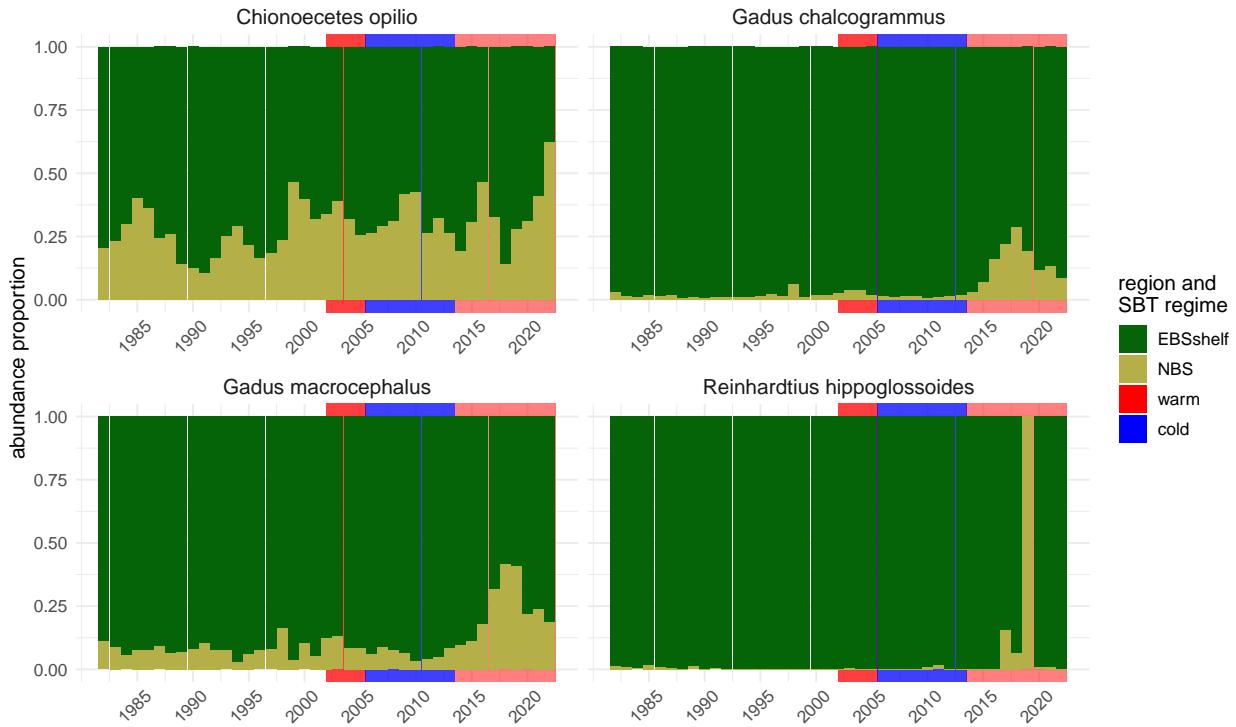
# Create a dummy data frame for legend rectangles
legend_rects <- data.frame(
```

```

ymin = -Inf,
ymax = Inf,
xmin = c(2002, 2005.5),
xmax = c(2005.5, 2013.5),
fill = c('red', 'blue'),
label = c('Period 1', 'Period 2')
)

#plot
ggplot() +
  geom_rect(data = legend_rects, aes(xmin = xmin, xmax = xmax,
                                      ymin = ymin, ymax = ymax, fill = label),
             alpha = 0.5, show.legend = TRUE) +
  annotate("rect", xmin = 2002, xmax = 2005.5,
           ymin = -Inf, ymax = Inf, alpha = 0.5, fill = 'red') +
  annotate("rect", xmin = 2013.5, xmax = 2022.5,
           ymin = -Inf, ymax = Inf, alpha = 0.5, fill = 'red') +
  annotate("rect", xmin = 2005.5, xmax = 2013.5,
           ymin = -Inf, ymax = Inf, alpha = 0.5, fill = 'blue') +
  geom_bar(data = ind_all3, aes(x = year, y = frac, fill = variable),
            stat = 'identity') +
  scale_fill_manual(values = c('EBSshelf' = '#046407',
                               'NBS' = '#B4AF46',
                               'Period 1' = 'red',
                               'Period 2' = 'blue'),
                     breaks = c('EBSshelf', 'NBS', 'Period 1', 'Period 2'),
                     labels = c('EBSshelf', 'NBS', 'warm', 'cold'),
                     name = 'region and\nSBT regime') +
  scale_x_continuous(breaks = seq(from = 1985, to = 2020, by = 5)) +
  labs(y = 'abundance proportion') +
  theme_minimal() +
  theme(strip.text = element_text(size = 12),
        text = element_text(size = 12),
        axis.text.x = element_text(angle = 45, hjust = 0.7, vjust = 0.8),
        axis.title.x = element_blank()) +
  facet_wrap(~ species, scales = 'free_x', nrow = 2) +
  guides(fill = guide_legend(override.aes = list(alpha = 1)))

```



Predicted CPUE by temperature regime and region

Predicted catch per unit effort (CPUE) data were analyzed by year and region, including the NBS, EBSshelf, and EBSslope by latitude or depth. The Northern EBSslope is defined as latitudes higher than 57.5° , while the Southern EBS slope is defined as latitudes lower than 57.5° . Additionally, the upper EBS slope is classified as depths less than 400m, and the deep EBS slope as depths greater than 400m.

```
# Define the file ID from the shareable link
file_id <- "1lNPrNm7V-gaUEnh7j5IxossWrEkDDPbl"

# Download density file
suppressMessages(
  drive_download(as_id(file_id), path = "./dens_sel.RData",
                 overwrite = TRUE))

#load file observed dataframe input
load('./dens_sel.RData') #dens_all

#classify region depending on EBSslope
dens_all$region1<-ifelse(dens_all$region=='NBS',
                           'NBS',
                           ifelse(dens_all$region=='EBSshelf',
                                  'EBSshelf',
                                  ifelse(dens_all$region==
                                         'EBSslope' &
                                         dens_all$Lat>57.5,
                                         'northern EBSslope','southern EBSslope')))

dens_all$region2<-ifelse(dens_all$region=='NBS',
                           'NBS',
                           ifelse(dens_all$region=='EBSshelf',
                                  'EBSshelf',
                                  ifelse(dens_all$region==
                                         'EBSslope' &
                                         dens_all$Lat>57.5,
                                         'northern EBSslope','southern EBSslope')))
```

```

        'NBS',
      ifelse(dens_all$region=='EBSshelf',
              'EBSshelf',
              ifelse(dens_all$region=='EBSslope' &
                     dens_all$depth_m<=400,
                     'upper EBSslope','deep EBSslope')))

#subset by year that include EBSslope data
dens_all1<-subset(dens_all,Year %in% c(2002:2016))

#sort factors
dens_all1$region<-factor(dens_all1$region,
                           levels = c('NBS' , 'EBSshelf' , 'EBSslope'))

# Create a dummy data frame for legend rectangles
legend_rects <- data.frame(
  ymin = -Inf,
  ymax = Inf,
  xmin = c(2002, 2005.5),
  xmax = c(2005.5, 2013.5),
  fill = c('red', 'blue'),
  label = c('Period 1', 'Period 2')
)

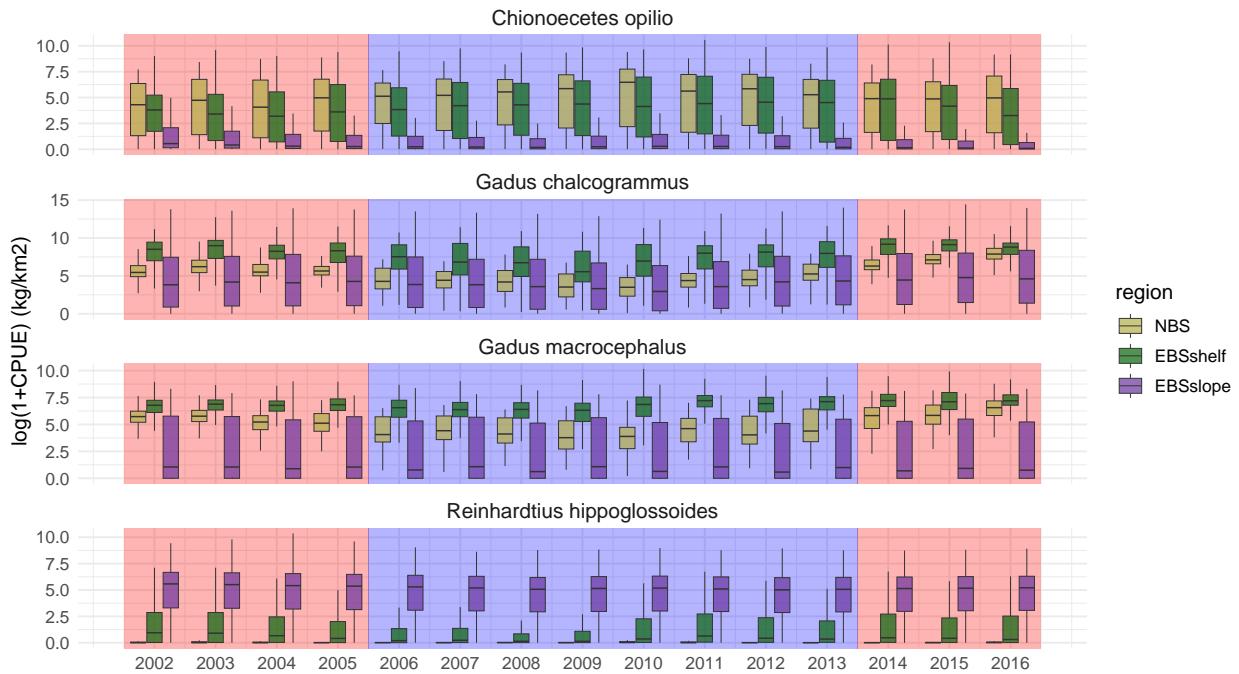
#plot pred cpue region1
print(
ggplot(dens_all1, aes(x = Year, y=log(1+dens),group=interaction(Year,region),
                      fill=region)) +
  annotate("rect", xmin = 2001.5, xmax = 2005.5, ymin = -Inf,
          ymax = Inf, alpha = 0.3, fill = 'red') +
  annotate("rect", xmin = 2013.5, xmax = 2016.5, ymin = -Inf,
          ymax = Inf, alpha = 0.3, fill = 'red') +
  annotate("rect", xmin = 2005.5, xmax = 2013.5, ymin = -Inf,
          ymax = Inf, alpha = 0.3, fill = 'blue') +
  geom_boxplot(alpha = 0.7,outliers = FALSE,
               position=position_dodge(width=0.8,preserve = "single"),
               size=0.2) +
  theme_minimal()+
  labs(y='log(1+CPUE) (kg/km2)',title = 'Predicted CPUE data',x='')+
  scale_fill_manual(values = c('EBSshelf' = '#046407',
                             'NBS' = '#B4AF46',
                             'EBSslope'='#6a329f'),
                    breaks = c('NBS','EBSshelf','EBSslope'),
                    labels = c('NBS','EBSshelf','EBSslope'),name='region') +
  scale_x_continuous(breaks=c(2002:2016))+

  theme(strip.text = element_text(size=12),text = element_text(size = 12))+

  facet_wrap(~species,scales='free_y',ncol = 1)
)

```

Predicted CPUE data



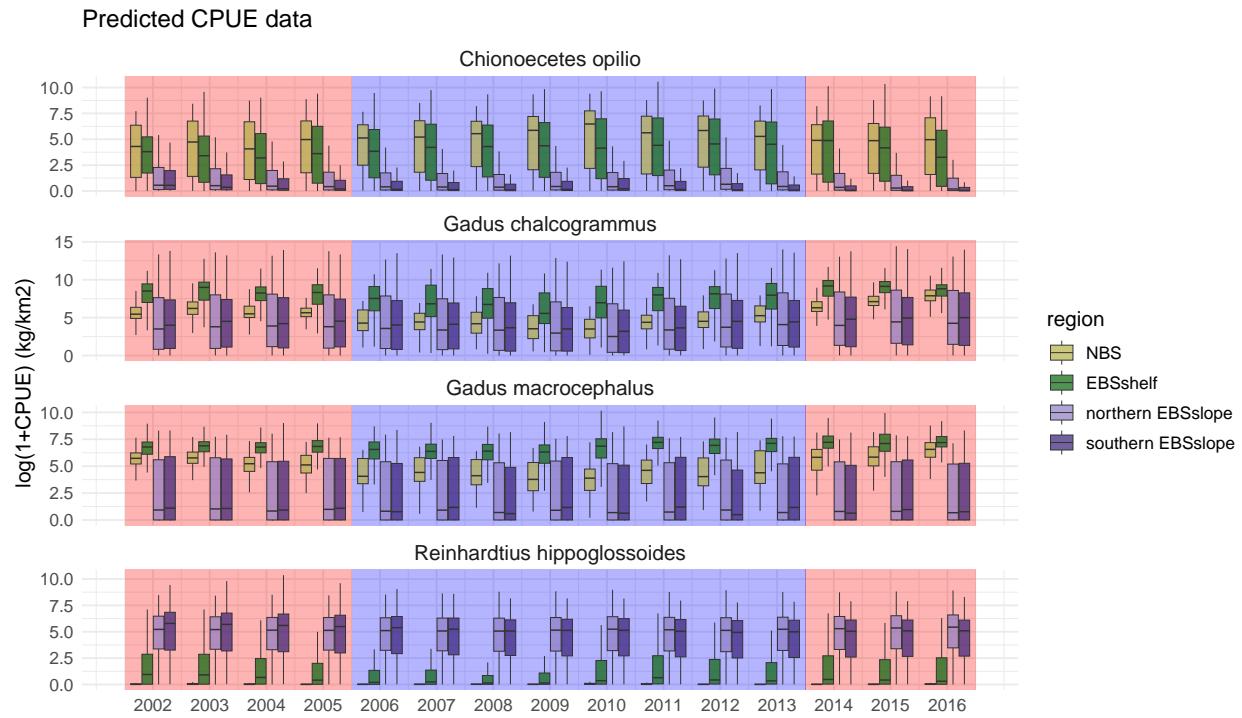
```
#sort factors
dens_all1$region1<-factor(dens_all1$region1,
                            levels = c('NBS' ,
                                       'EBSshelf' ,
                                       'northern EBSslope',
                                       'southern EBSslope'))

#plot pred cpue region1
print(
ggplot(dens_all1, aes(x = Year, y=log(1+dens),group=interaction(Year,region1),
                      fill=region1)) +
  annotate("rect", xmin = 2001.5, xmax = 2005.5, ymin = -Inf,
          ymax = Inf, alpha = 0.3, fill='red') +
  annotate("rect", xmin = 2013.5, xmax = 2016.5, ymin = -Inf,
          ymax = Inf, alpha = 0.3, fill = 'red') +
  annotate("rect", xmin = 2005.5, xmax = 2013.5, ymin = -Inf,
          ymax = Inf, alpha = 0.3, fill = 'blue') +
  geom_boxplot(alpha = 0.7,outliers = FALSE,
               position=position_dodge(width=0.8,preserve = "single"),
               size=0.2) +
  theme_minimal()+
  labs(y='log(1+CPUE) (kg/km2)',title = 'Predicted CPUE data')+ 
  scale_fill_manual(values = c('NBS' = '#B4AF46',
                               'EBSshelf' = '#046407',
                               'northern EBSslope'='#8e7cc3',
                               'southern EBSslope'='#351c75'),
                    breaks = c('NBS','EBSshelf',
                              'northern EBSslope','southern EBSslope'),
                    labels = c('NBS','EBSshelf',
```

```

        'northern EBSslope','southern EBSslope'),
  name='region') +
  labs(x='')+
  scale_x_continuous(breaks=c(2002:2016))+
  theme(strip.text = element_text(size=12),
    text = element_text(size = 12))+ 
  facet_wrap(~species,scales='free_y',ncol = 1)
)

```



```

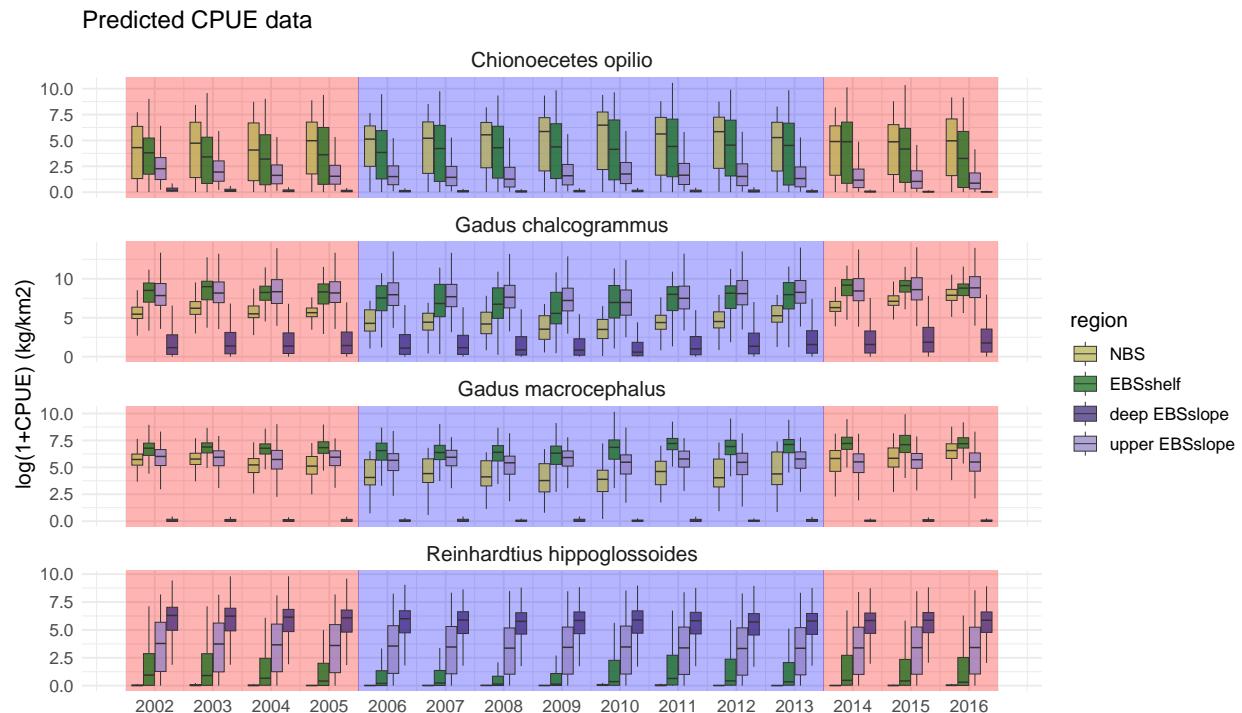
#sort factors
dens_all1$region2<-factor(dens_all1$region2,
                            levels = c('NBS' , 'EBSshelf' ,
                                      'upper EBSslope','deep EBSslope'))

#plot pred cpue region2
print(
ggplot(dens_all1, aes(x = Year, y=log(1+dens),group=interaction(Year,region2),
                     fill=region2)) +
  annotate("rect", xmin = 2001.5, xmax = 2005.5, ymin = -Inf,
          ymax = Inf, alpha = 0.3, fill = 'red') +
  annotate("rect", xmin = 2013.5, xmax = 2016.5, ymin = -Inf,
          ymax = Inf, alpha = 0.3, fill = 'red') +
  annotate("rect", xmin = 2005.5, xmax = 2013.5, ymin = -Inf,
          ymax = Inf, alpha = 0.3, fill = 'blue') +
  geom_boxplot(alpha = 0.7,outliers = FALSE,
               position=position_dodge(width=0.8,preserve = "single"),
               size=0.2) +
  theme_minimal()+
  labs(y='log(1+CPUE) (kg/km2)',title = 'Predicted CPUE data')+
```

```

scale_fill_manual(values = c('EBSshelf' = '#046407',
                           'NBS' = '#B4AF46',
                           'deep EBSslope'='#351c75',
                           'upper EBSslope'='#8e7cc3'),
  breaks = c('NBS','EBSshelf',
            'deep EBSslope','upper EBSslope'),
  labels = c('NBS','EBSshelf',
            'deep EBSslope','upper EBSslope'),
  name='region') +
  labs(x='')+ 
  scale_x_continuous(breaks=c(2002:2016))+ 
  theme(strip.text = element_text(size=12),
        text = element_text(size = 12))+ 
  facet_wrap(~species,scales='free_y',ncol = 1)
)

```



```

# Compute the mean values
mean_values <- dens_all1 %>%
  group_by(Year, region2, species) %>%
  summarise(mean_log_dens = mean(log(1 + dens),
                                   na.rm = TRUE), .groups = 'drop')

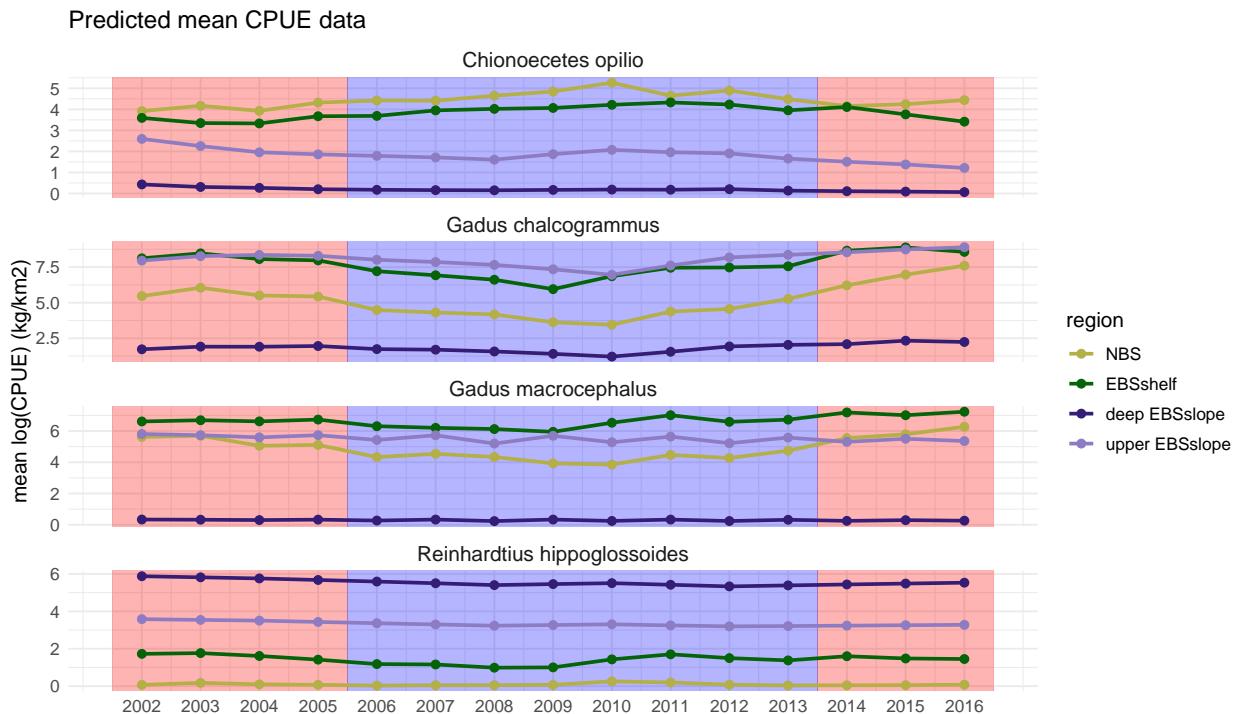
# Plot the time series of mean values
print(
  ggplot(mean_values, aes(x = Year, y = mean_log_dens, group = region2, color = region2)) +
  annotate("rect", xmin = 2001.5, xmax = 2005.5, ymin = -Inf,
          ymax = Inf, alpha = 0.3, fill = 'red') +
  annotate("rect", xmin = 2013.5, xmax = 2016.5, ymin = -Inf,
          ymax = Inf, alpha = 0.3, fill = 'red')
)

```

```

annotate("rect", xmin = 2005.5, xmax = 2013.5, ymin = -Inf,
        ymax = Inf, alpha = 0.3, fill = 'blue') +
  geom_line(lineWidth = 1) +
  geom_point(size = 2) +
  theme_minimal() +
  labs(y = 'mean log(CPUE) (kg/km2)', title = 'Predicted mean CPUE data') +
  scale_color_manual(values = c('EBSshelf' = '#046407',
                               'NBS' = '#B4AF46',
                               'deep EBSslope' = '#351c75',
                               'upper EBSslope' = '#8e7cc3'),
                     breaks = c('NBS', 'EBSshelf',
                               'deep EBSslope', 'upper EBSslope'),
                     labels = c('NBS', 'EBSshelf',
                               'deep EBSslope', 'upper EBSslope'),
                     name = 'region') +
  labs(x = '') +
  scale_x_continuous(breaks = c(2002:2016)) +
  theme(strip.text = element_text(size = 12), text = element_text(size = 12)) +
  facet_wrap(~ species, scales = 'free_y', ncol = 1)
)

```



Latitudinal and bathymetrical COG relative to temperature and biomass and effective area occupied

Effective area occupied, as well as the latitudinal and bathymetric center of gravity (COG), were analyzed relative to sea bottom temperature (SBT) and biomass from 2002 to 2016. This analysis used predicted data from the EBSshelf, NBS and EBSslope

```

# Define the files ID from the shareable link
file_id1 <- "1TEAsIAqWPjyVoOM_eljU3D4DZC5IhZvk"
file_id2 <- "1y345z3DbErtCtz3nA_Xu91LStejLVntU"

# Download density and grid data
suppressMessages(
  drive_download(as_id(file_id1), path = "./dens_shelfslope.RData",
                 overwrite = TRUE))
suppressMessages(
  drive_download(as_id(file_id2), path = "./grid_EBS_NBS_envs.RData",
                 overwrite = TRUE))

#load files density slope and shelf data
load('./dens_shelfslope.RData') #dens_all
load('./grid_EBS_NBS_envs.RData') #xx

#df to store results
metrics_df<-data.frame(matrix(NA,nrow = 0,ncol=13))
names(metrics_df)<-c("Year","COG_lat","COG_lon","COG_depth","COG_temp",
                     "total_bio",'mean_dens','mean_temp','q10',"q20","q80",
                     "q90","species")

#shelf slope spp
shelfslope_spp<-unique(dens_all$species)

#calculate center of gravity, percentiles and effective area occupied
#loop over common spp in shelf and slope
for (s in shelfslope_spp) {

  #s<-shelfslope_spp[1];

  #cat(paste('##### ',s," - ",'\n'))

  #species dens
  dens<-subset(dens_all,species==s & Year %in% c(2002:2016))

  #grid with envs
  grid.ebs_year1<-subset(xx,Year %in% c(2002:2016))

  #merge dens and grid
  df<-merge(grid.ebs_year1,
            dens,by=c('Lat','Lon','Area_in_survey_km2','Stratum',
                      'region','DepthGEBCO','depth_m','Year','Temp'))

  #calculate quantiles using tapply
  quantiles <- aggregate(dens ~ Year, data = df,
                         FUN = quantile,c(0.10,0.20,0.80,0.90))
  quantiles<- data.frame(as.matrix(quantiles))
  names(quantiles)<-c('Year','q10','q20','q80','q90')

  #COG and total abundance
  centroids <- df %>%
    group_by(Year) %>%

```

```

    summarise(COG_lat=sum(dens*Lat,na.rm=TRUE)/sum(dens,na.rm=TRUE),
              COG_lon=sum(dens*Lon,na.rm=TRUE)/sum(dens,na.rm=TRUE),
              COG_depth=sum(dens*DepthGEBCO,na.rm=TRUE)/sum(dens,na.rm=TRUE),
              COG_temp=sum(dens*Temp,na.rm = TRUE)/sum(dens,na.rm=TRUE),
              total_bio=sum(dens*Area_in_survey_km2,na.rm=TRUE),
              mean_dens=mean(dens,na.rm=TRUE),
              mean_temp=mean(Temp,na.rm=TRUE))

#cbind data
metrics<-cbind(centroids,quantiles[,c('q10','q20','q80','q90')],'species'=s)

#append
metrics_df<-rbind(metrics_df,metrics)
}

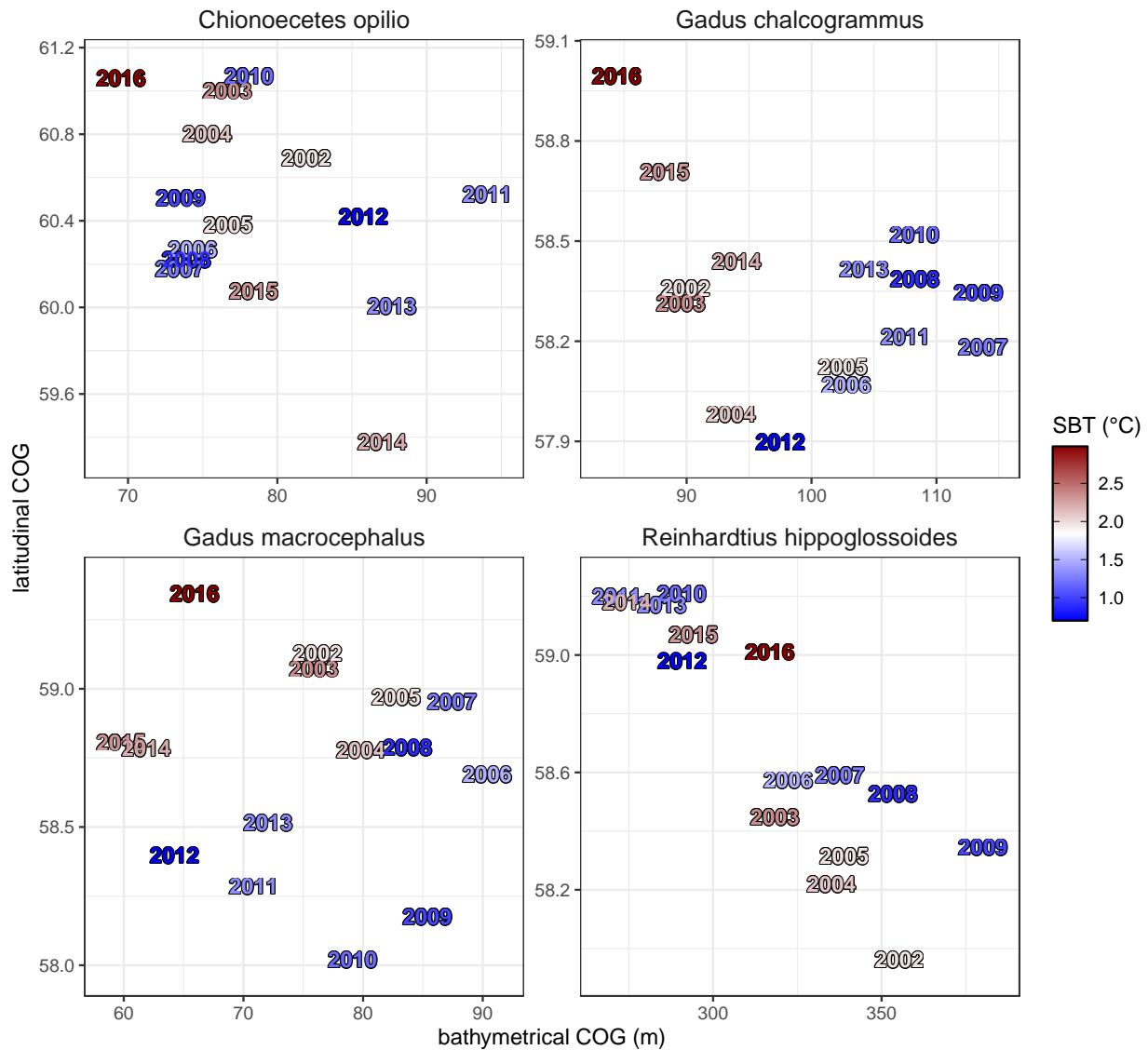
#save table
save(metrics_df,file='./metrics_df.RData')

# Define a custom color scale function
custom_colors <- colorRampPalette(c("blue", "white", "darkred"))

#selected species
sel_sp<-c("Gadus chalcogrammus", "#Alaskan pollock
          "Gadus macrocephalus", "#pacific cod
          "Reinhardtius hippoglossoides", "#Greenland turbot
          "Chionoecetes opilio") #snow crab

#plot using SBT color
ggplot(data = subset(metrics_df, species %in% sel_sp)) +
  geom_shadowtext(aes(x = COG_depth, y = COG_lat, color = mean_temp, label = Year),
                  fontface = 'bold', bg.r = 0.05) +
  theme_bw() +
  labs(x = 'bathymetrical COG (m)', y = 'latitudinal COG') +
  theme(aspect.ratio = 1,
        strip.background = element_blank(),
        strip.text = element_text(size = 12)) +
  scale_color_gradientn(colors = custom_colors(100), name = 'SBT (°C)',
                        guide = guide_colorbar(frame.colour = "black", ticks.colour = "black")) +
  scale_x_continuous(expand = expansion(mult = 0.1)) + # Add buffer on the x-axis
  scale_y_continuous(expand = expansion(mult = 0.1)) + # Add buffer on the y-axis
  facet_wrap(~species, scales = 'free')

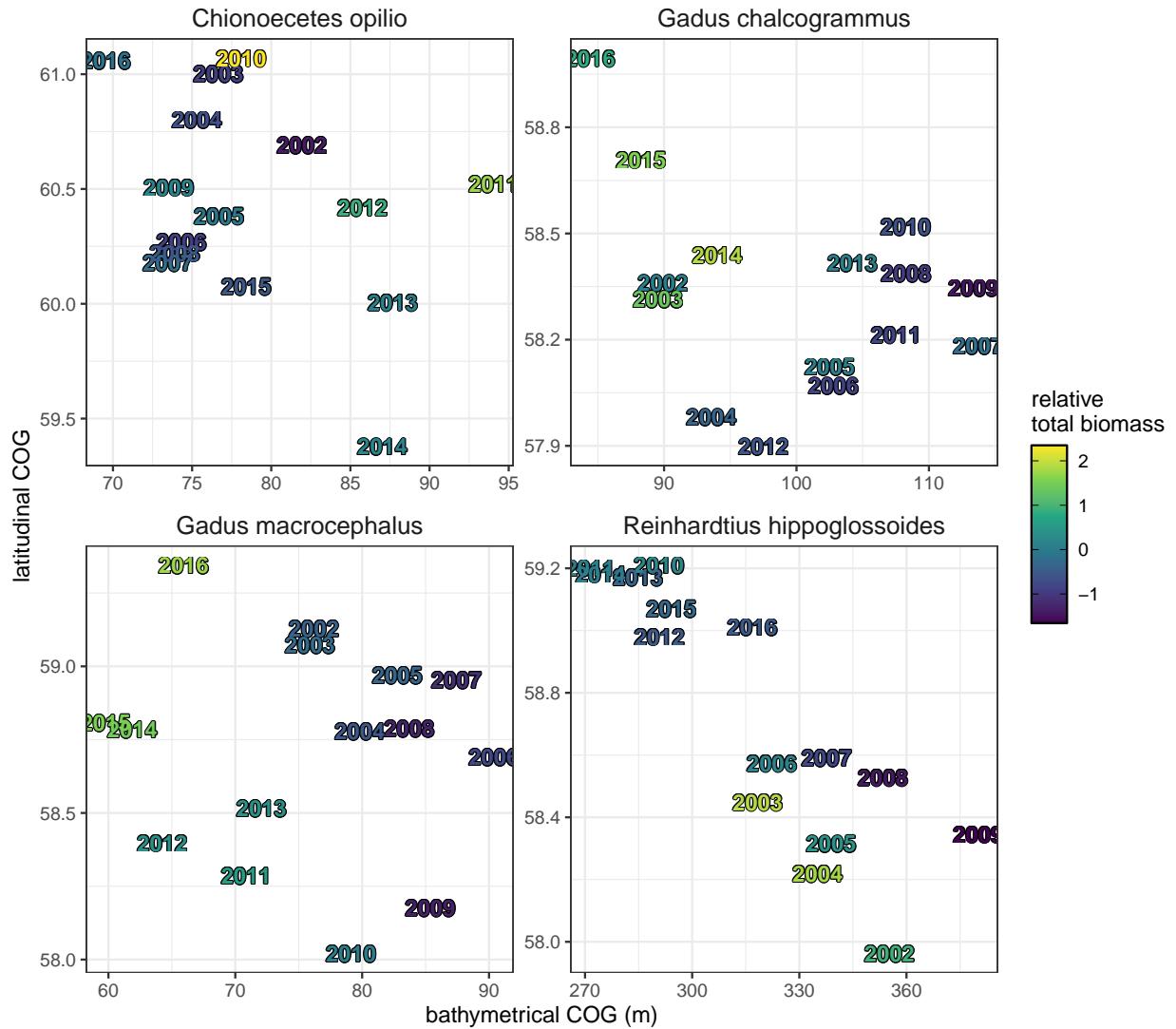
```



```
#scale to keep one legend
metrics_df1 <- metrics_df %>%
  group_by(species) %>%
  mutate(bio_scaled = scale(total_bio))

#plot using scaled bio color
ggplot(data=subset(metrics_df1,species %in% sel_sp)) +
  geom_shadowtext(aes(x=COG_depth,y=COG_lat,color=bio_scaled,label=Year),
                 fontface='bold',bg.r = 0.05) +
  theme_bw() +
  labs(x='bathymetrical COG (m)',y='latitudinal COG') +
  theme(aspect.ratio = 1,strip.background = element_blank(),strip.text = element_text(size=12)) +
  scale_color_viridis_c(name = 'relative\ntotal biomass',guide = guide_colorbar(frame.colour = "black",
  ticks.colour = "black")) +
```

```
facet_wrap(~species,scales='free')
```



```
#define a data frame for the rectangles
rect_data <- data.frame(xmin = c(2002, 2005.5, 2013.5),
                        xmax = c(2005.5, 2013.5, 2016),
                        fill = factor(c("warm", "cold", "warm")))

#plot effective area
ggplot(data=subset(metrics_df,species %in% sel_sp))+ 
  geom_rect(data = rect_data, aes(ymin = -Inf, ymax = Inf,
                                  xmin = xmin, xmax = xmax, fill = fill),
            alpha = 0.3) +
  geom_line(aes(x=Year,y=total_bio/mean_dens),color='black')+
```

```

  labs(x=' ',y='effective area occupied (km2)')+  

  theme_bw() +  

  scale_x_continuous(breaks=c(2002,2006,2010,2014,2016))+  

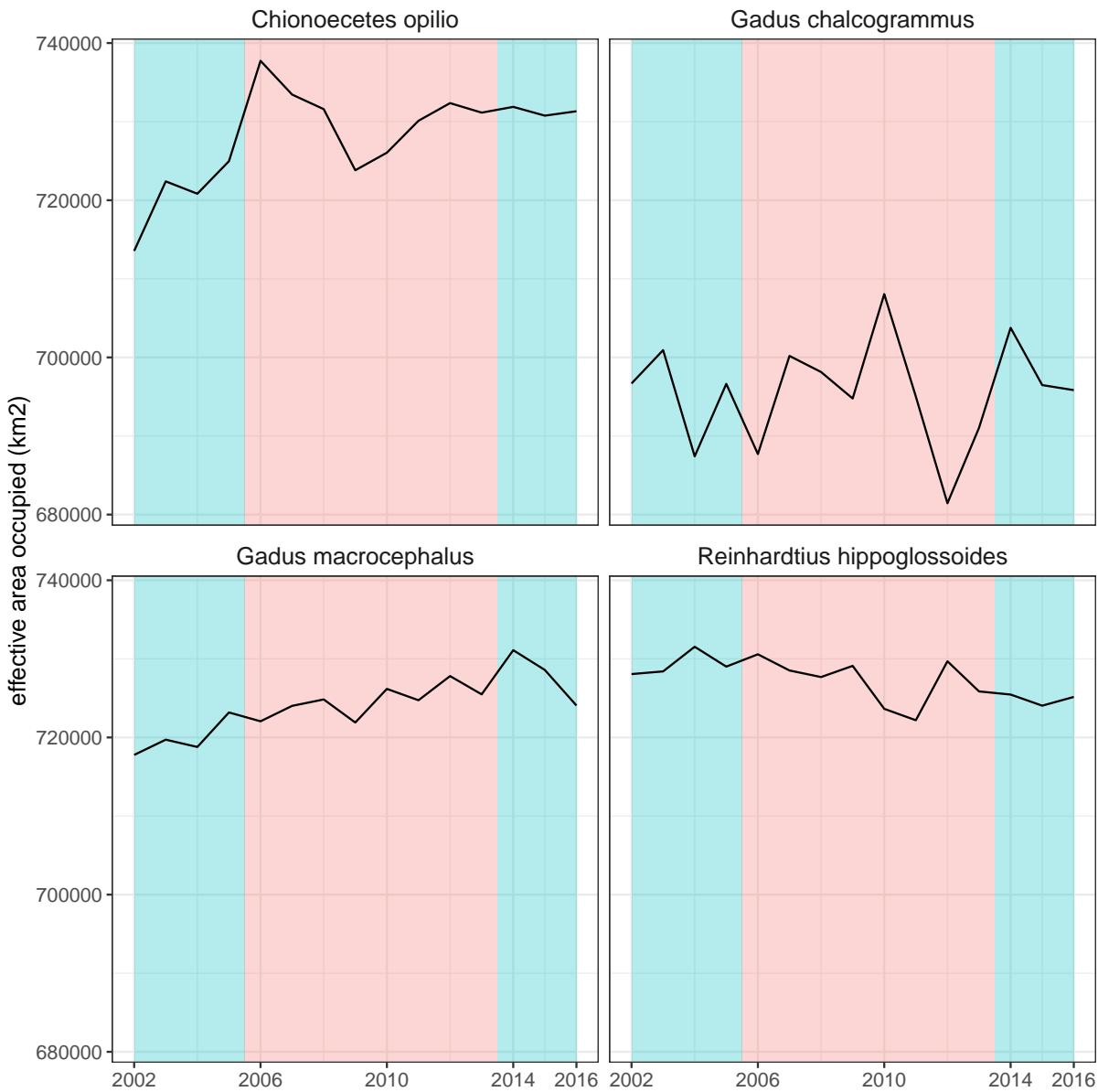
  theme(aspect.ratio = 1,strip.background = element_blank(),  

        strip.text = element_text(size=12),text = element_text(size = 12))+  

  facet_wrap(~species,nrow = 2)+  

  theme(legend.position = 'none',aspect.ratio = 1)

```



Range edges relative to cold pool area

Range edges of species relative to the cold pool were visualized and saved as PNG and GIF files. (*This process can take a few minutes, approximately 10 minutes per species, depending on the computer's performance.*)

Cumulative distribution of predicted biomass relative to depth

The cumulative distribution of predicted biomass was analyzed in relation to depth and depth ranges.

```
# Define the files ID from the shareable link
file_id2 <- "1y345z3DbErtCtz3nA_Xu91LStejLVntU"

suppressMessages(
  drive_download(as_id(file_id2), path = "./grid_EBS_NBS_envs.RData",
    overwrite = TRUE))

#load grid
load('./grid_EBS_NBS_envs.RData') #xx
xx <- as.data.table(xx)

#average SBT by region and year
temp_region<-aggregate(Temp ~ Year + region,xx,FUN=mean)
names(temp_region)<-c('year','region','temp')

#get average SBT for all region (EBSshelf+NBS+EBSSlope)
alltemp<-aggregate(temp ~ year, temp_region,FUN=mean)
alltemp$region<-'all'
alltemp<-alltemp[,c("year","region","temp")]

#rbind
temp_region<-rbind(temp_region,alltemp)

#shelf slope spp
shelfslope_spp<-unique(dens_all$species)

# Define the file ID from the shareable link
file_id2 <- "1y345z3DbErtCtz3nA_Xu91LStejLVntU"

# Download the file
suppressMessages(
  drive_download(as_id(file_id2), path = "./grid_EBS_NBS_envs.RData",
    overwrite = TRUE))

#df to store depth percentiles data to evaluate range
depth_percentiles<-data.frame(matrix(NA,ncol = 7,nrow = 0))
names(depth_percentiles)<-c('species','year','temp','depth_p10',
  'depth_p20','depth_p80','depth_p90')

#calculate center of gravity, edges and effective area occupied
#loop over common spp in shelf and slope
for (s in shelfslope_spp) { #[ -6] no anoplopoma fimbria

  #cat(paste('##### ',s," - ",'\n'))

  #species dens
  dens<-subset(dens_all,species==s & Year %in% c(2002:2016))

  #grid with envs
  grid.ebs_year1<-subset(xx,Year %in% c(2002:2016))
```

```

#merge dens and grid
df<-merge(grid.ebs_year1,dens,by=c('Lat','Lon','Area_in_survey_km2',
                                     'Stratum','region','DepthGEBCO',
                                     'depth_m','Year','Temp'))

#calculate quantiles using tapply
quantiles <- data.frame(as.matrix(aggregate(dens ~ Year, data = df,
                                              FUN = quantile,
                                              c(0.10,0.20,0.80,0.90)))) 

# Initialize a list to store plots
plot_list <- list()

#loop over years
for (y in 2002:2016) {

  #y<-'2002'

  #print progress
  #cat(paste('##### ',y," - ",'\n'))

  #temp region and year for label
  alltemp_region <- subset(temp_region, region == 'all' &
                           year %in% 2002:2016)
  temp_min <- min(alltemp_region$temp)
  temp_max <- max(alltemp_region$temp)
  itemp_df <- subset(alltemp_region, year == y)
  itemp <- itemp_df$temp

  #get quantiles
  p1 <- quantiles[which(quantiles$Year==y),'dens.10.']
  p2 <- quantiles[which(quantiles$Year==y),'dens.90.']

  #filter
  df$quantile <- ifelse(df$dens < p2 & df$dens > p1, 1, 0)
  all_bio4<-df[which(df$Year==y),]

  # Sort the dataframe by DepthGEBCO
  all_bio4 <- all_bio4[order(all_bio4$DepthGEBCO),]

  # Calculate cumulative biomass
  all_bio4$bio<-all_bio4$dens*all_bio4$Area_in_survey_km2
  all_bio4$cumulative_biomass <- cumsum(all_bio4$bio)

  # Normalize the cumulative biomass to get a cumulative distribution (0 to 1)
  all_bio4$cumulative_biomass_scale <- all_bio4$cumulative_biomass /
                                         max(all_bio4$cumulative_biomass)

  # Find the depths corresponding to the 10th and 90th percentiles
  ps <- c(0.10, 0.90)
  depth_10th <- all_bio4$DepthGEBCO[which.min(
                                         abs(all_bio4$cumulative_biomass_scale - ps[1]))]
  depth_90th <- all_bio4$DepthGEBCO[which.min(

```

```

                abs(all_bio4$cumulative_biomass_scale - ps[2]))]
y_10th <- ps[1]
y_90th <- ps[2]

# Find the depths corresponding to the 10th and 90th percentiles
ps2 <- c(0.20, 0.80)
depth_20th <- all_bio4$DepthGEBCO[which.min(
  abs(all_bio4$cumulative_biomass_scale - ps2[1]))]
depth_80th <- all_bio4$DepthGEBCO[which.min(
  abs(all_bio4$cumulative_biomass_scale - ps2[2]))]
y_20th <- ps2[1]
y_80th <- ps2[2]

#limits
lims<-ifelse(s=='Reinhardtius hippoglossoides',1200,500)

#function to get the color scale
get_temp_color <- function(itemp, temp_min, temp_max) {
  gradient <- colorRampPalette(c("blue", "white", "darkred"))(100)
  temp_scaled <- rescale(itemp, to = c(1, 100), from = c(temp_min, temp_max))
  return(gradient[round(temp_scaled)])
}

# Plot
p1<-
  ggplot()+
    geom_line(data=all_bio4, aes(x = DepthGEBCO, y = cumulative_biomass),
               na.rm=TRUE) +
    geom_vline(xintercept = c(depth_10th, depth_90th),
               linetype = "dashed", color = c('grey20'), na.rm=TRUE) +
    geom_ribbon(data = all_bio4, aes(x = DepthGEBCO, ymin = cumulative_biomass,
                                      ymax = 1),
                fill = "grey80", alpha = 0.5) +
    annotate("text", x = depth_10th, y = max(all_bio4$cumulative_biomass)/2,
             label = paste("10th Percentile Depth:", round(depth_10th, 2)),
             angle = 90, vjust = -0.5, color = "grey20",size=4) +
    annotate("text", x = depth_90th, y = max(all_bio4$cumulative_biomass)/2,
             label = paste("90th Percentile Depth:", round(depth_90th, 2)),
             angle = 90, vjust = 1.5, color = "grey20",size=4) +
    theme_bw()+
    labs(x='depth (m)',y='cumulative biomass')+
    scale_x_continuous(limits = c(0,lims),expand = c(0,0))+ 
    scale_y_continuous(expand = c(0,0))+ 
    annotate("shadowtext",x=350,y=max(all_bio4$cumulative_biomass)/2,
            bg.color = "black",
            color = get_temp_color(itemp, temp_min, temp_max),label=y,
            bg.r = 0.05,size=12)+ 
    annotate("shadowtext",x=350,y=max(all_bio4$cumulative_biomass)/4,
            bg.color = "black",
            color = 'black',
            label=paste("P10-90 range depth:", (depth_90th - depth_10th)),
            bg.r = 0.00,size=6)+ 
    theme(panel.grid.minor = element_line(linetype='dashed')),

```

```

    axis.text.y = element_blank(),
    axis.text.x = element_text(vjust=5,margin=margin(-7,0,7,0,unit='points')),
    axis.ticks.length = unit(-5,'points'),axis.text = element_text(size=10))

#append plot
plot_list[[as.character(y)]]<-p1

#append results
depth_percentiles<-rbind(depth_percentiles,
                           data.frame(species=s,
                                       year=y,
                                       temp=itemp_df$temp,
                                       depth_p10=depth_10th,
                                       depth_p20=depth_20th,
                                       depth_p80=depth_80th,
                                       depth_p90=depth_90th))
}

#all plots
plots<-plot_grid(plotlist = plot_list,ncol = 3)

#title
title <- ggdraw() + draw_label(s, fontface='bold',size = 20)

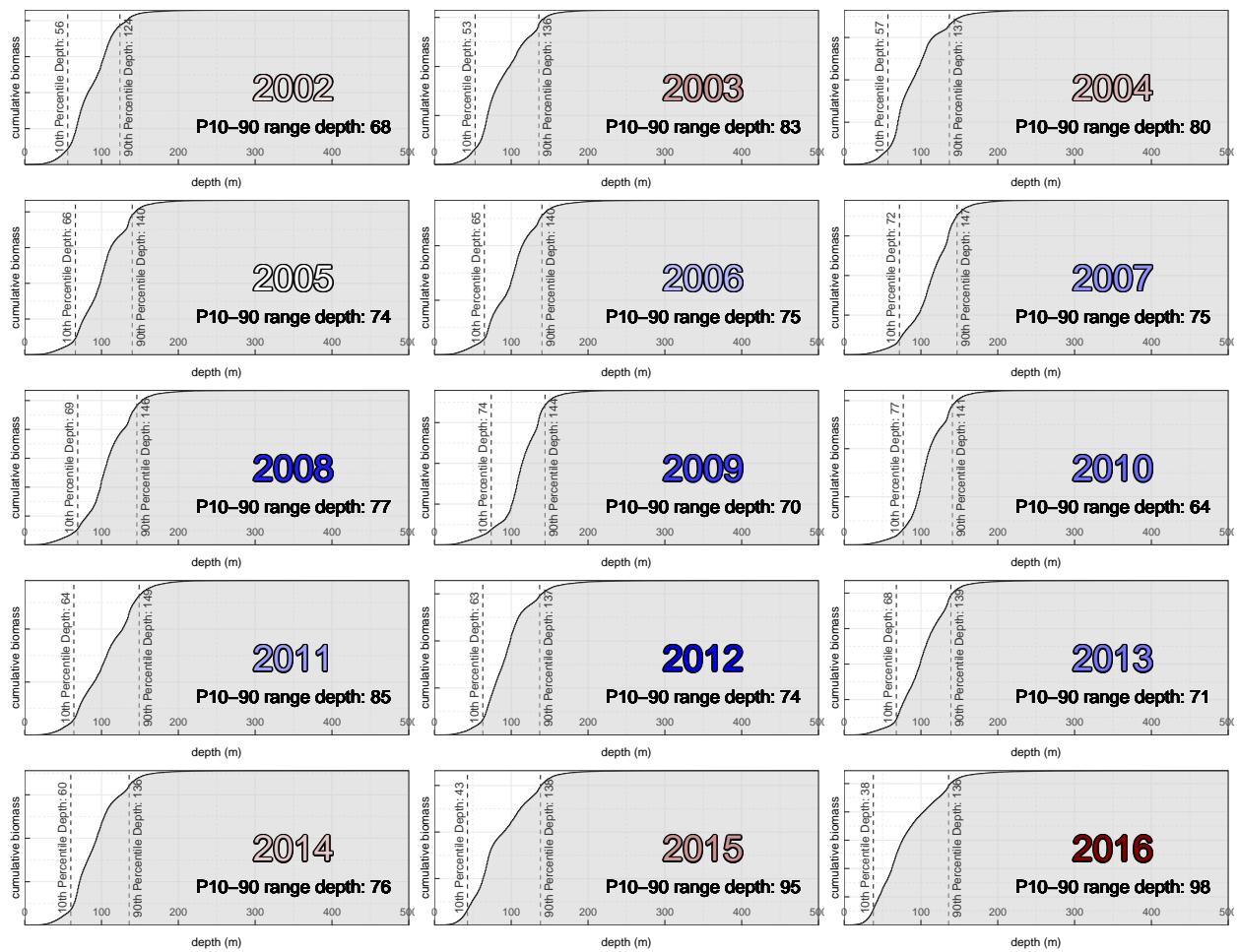
#grid plot
plot<-plot_grid(title,plots,ncol=1, rel_heights=c(0.05, 1))
print(plot)

#save plot
ggsave(filename = paste0('./cumdist_',s,'.png'),plot = plot,
       width = 14, height = 13, units = "in")

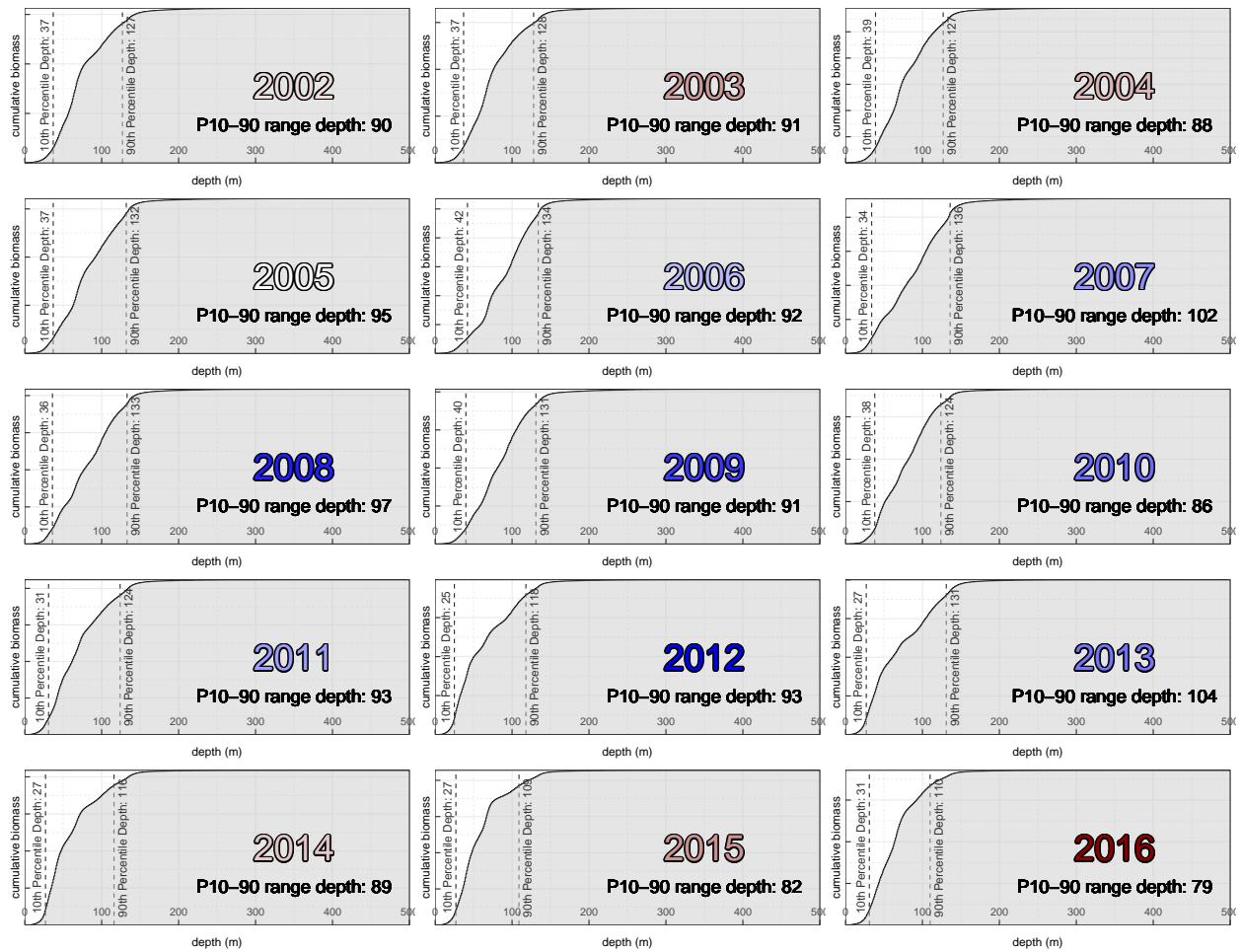
}

```

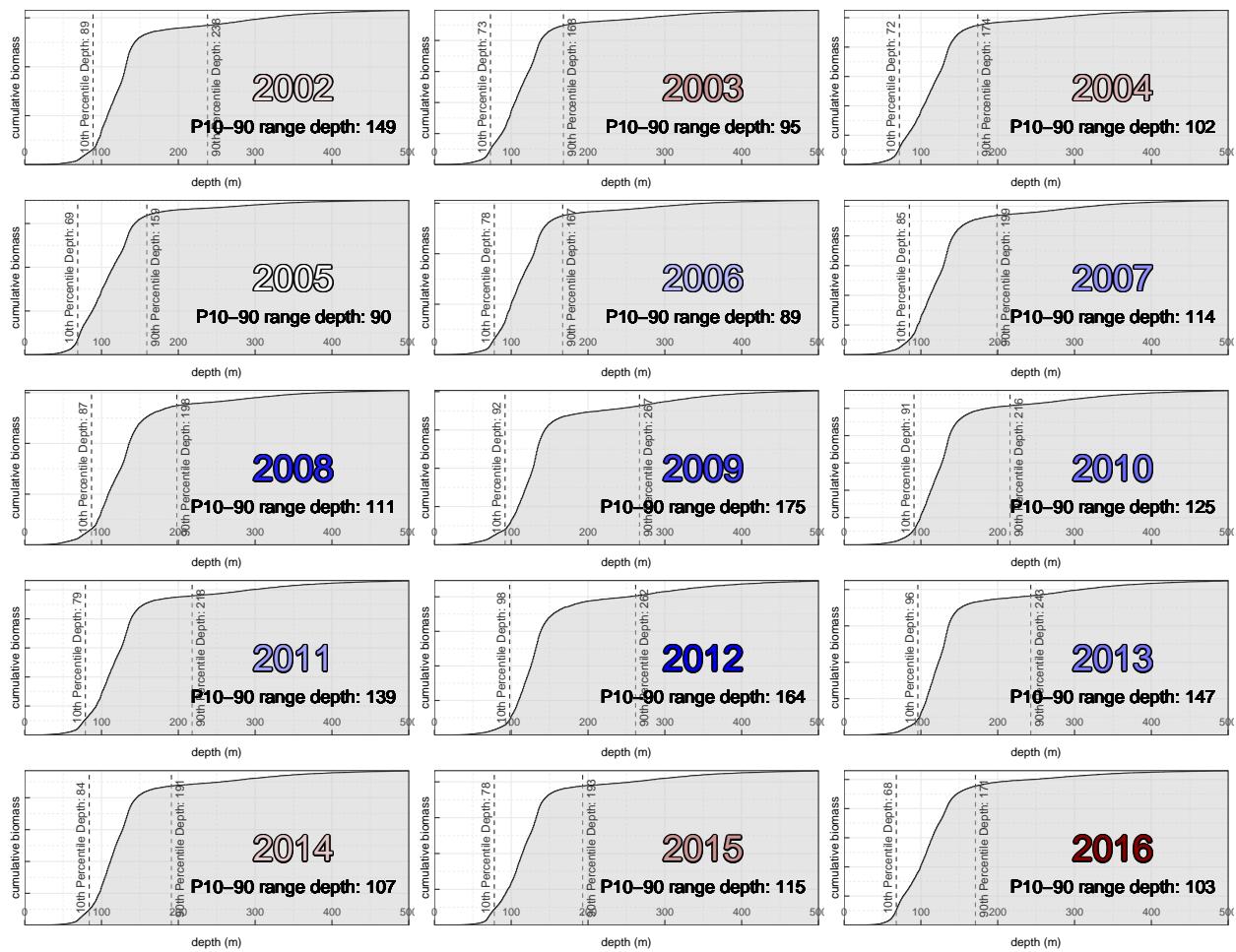
Gadus chalcogrammus



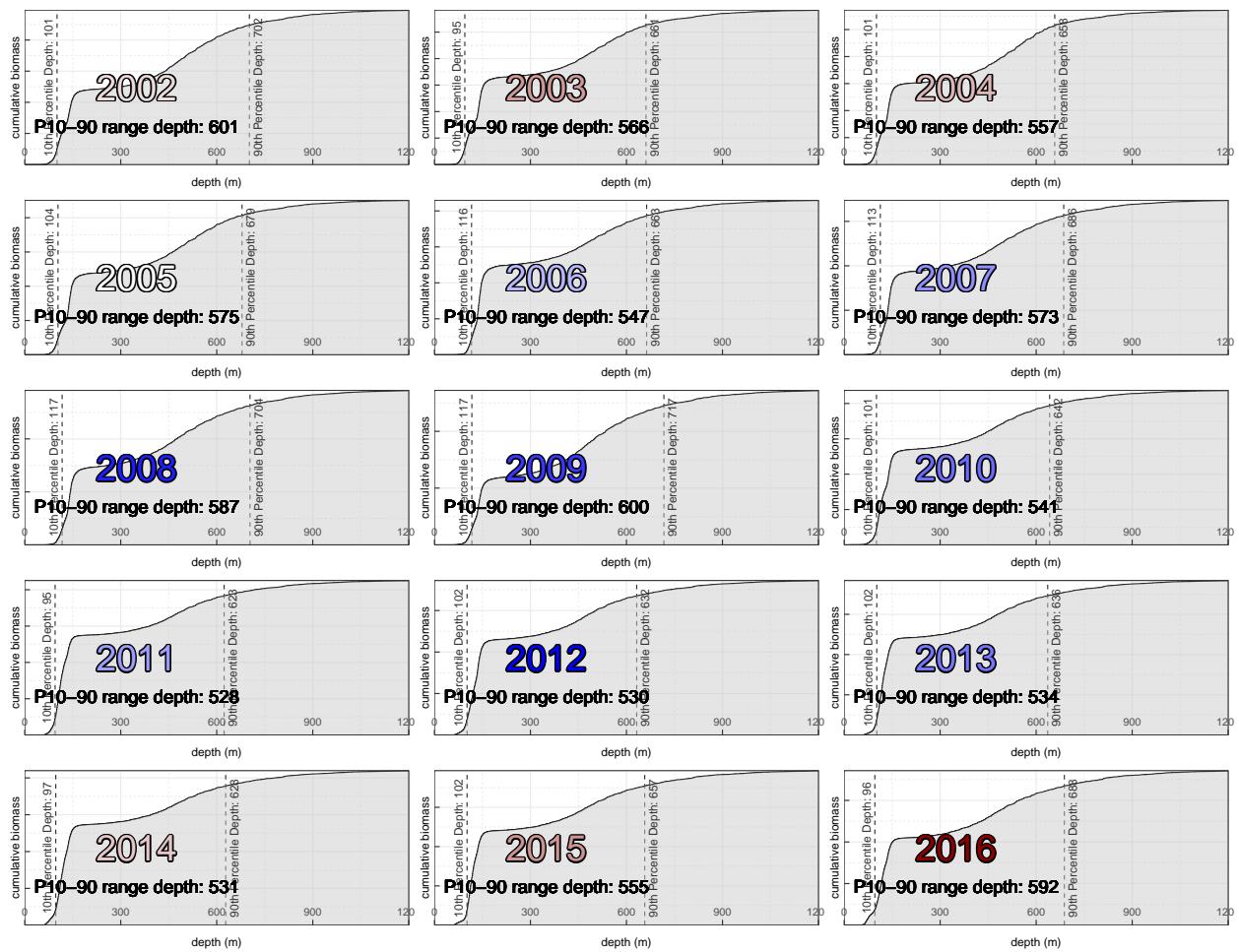
Gadus macrocephalus



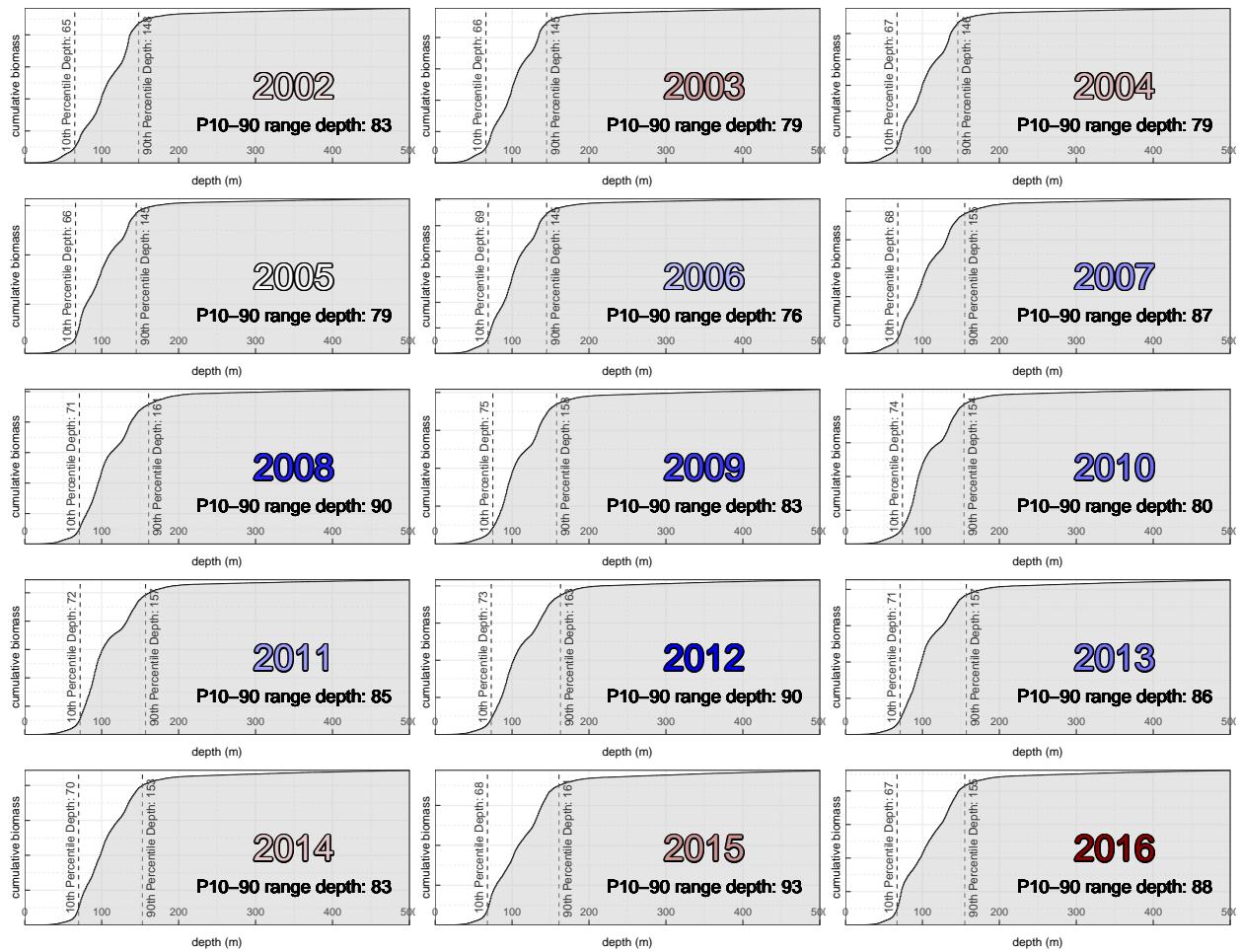
Atheresthes stomias



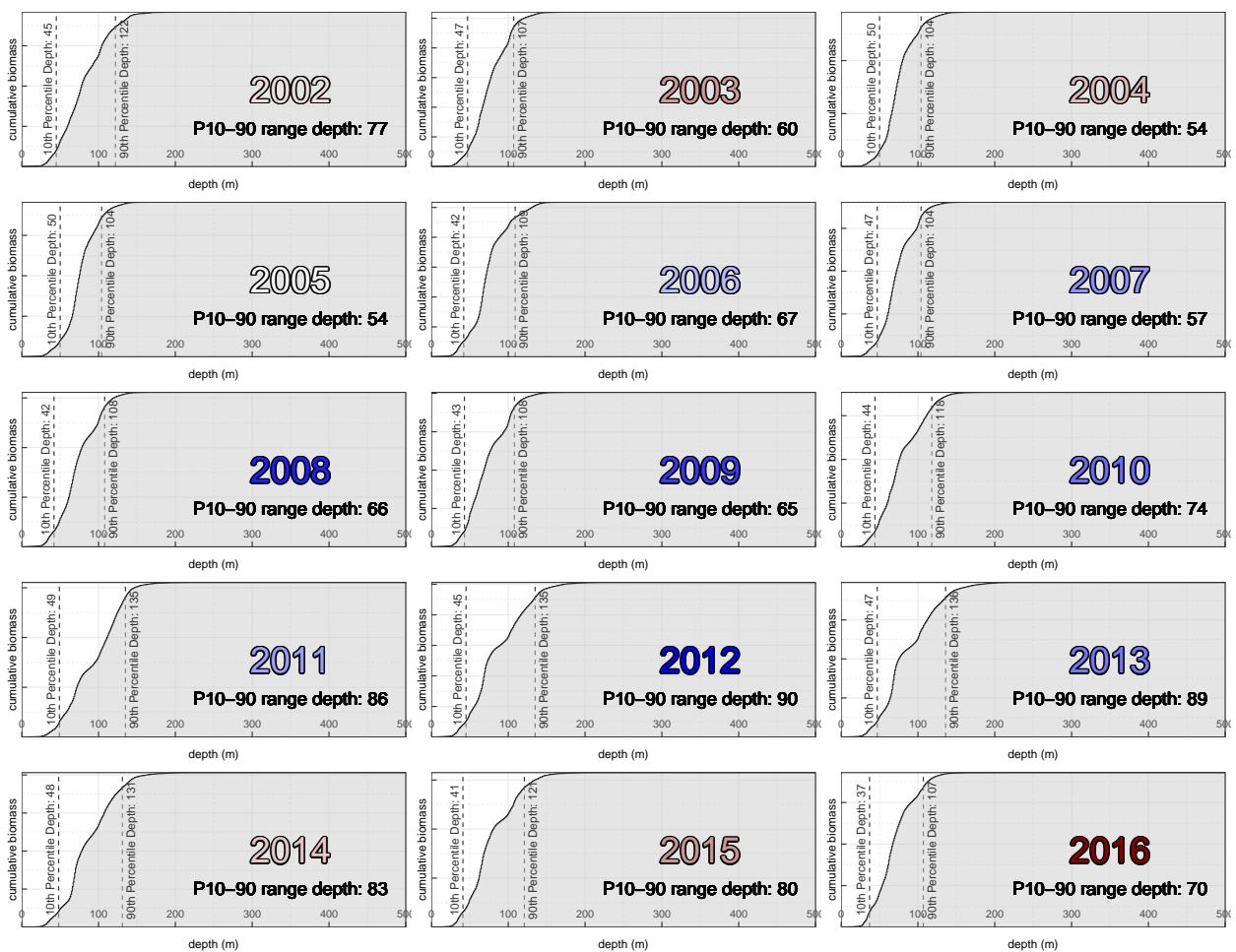
Reinhardtius hippoglossoides



Hippoglossoides elassodon



Chionoecetes opilio

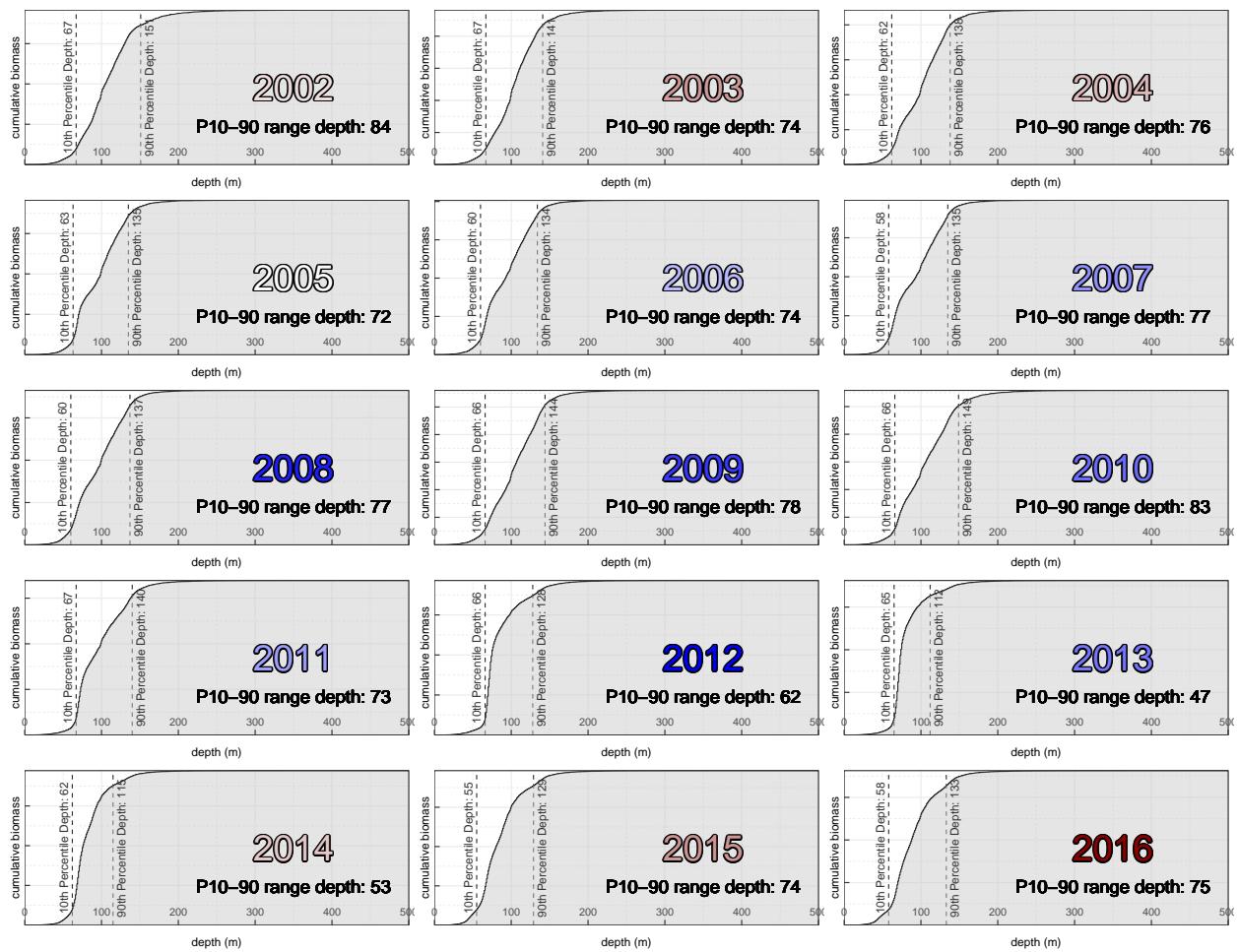


```

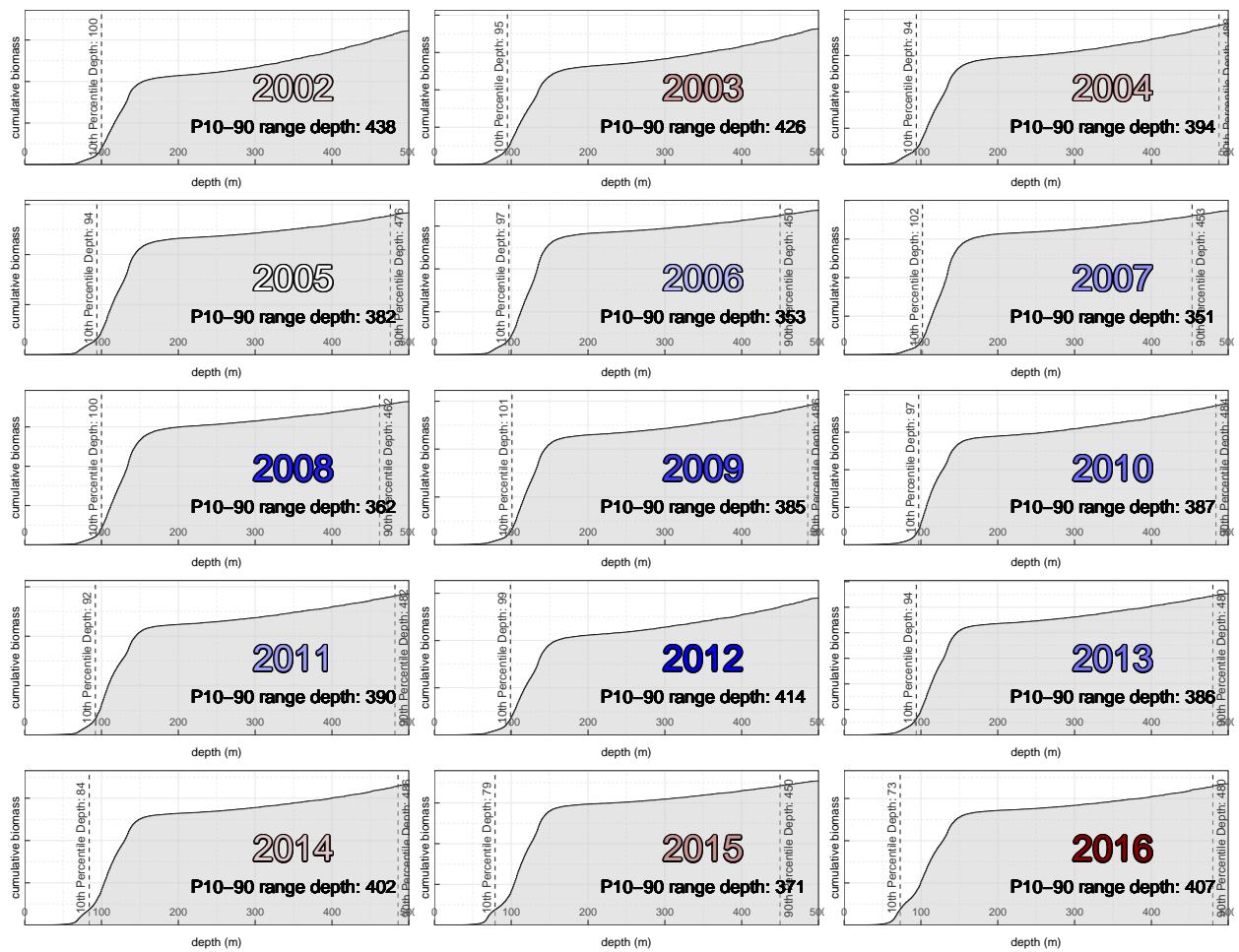
## Warning: Removed 1 row containing missing values or values outside the scale range
## ('geom_text()').
## Removed 1 row containing missing values or values outside the scale range
## ('geom_text()').
## Removed 1 row containing missing values or values outside the scale range
## ('geom_text()').

```

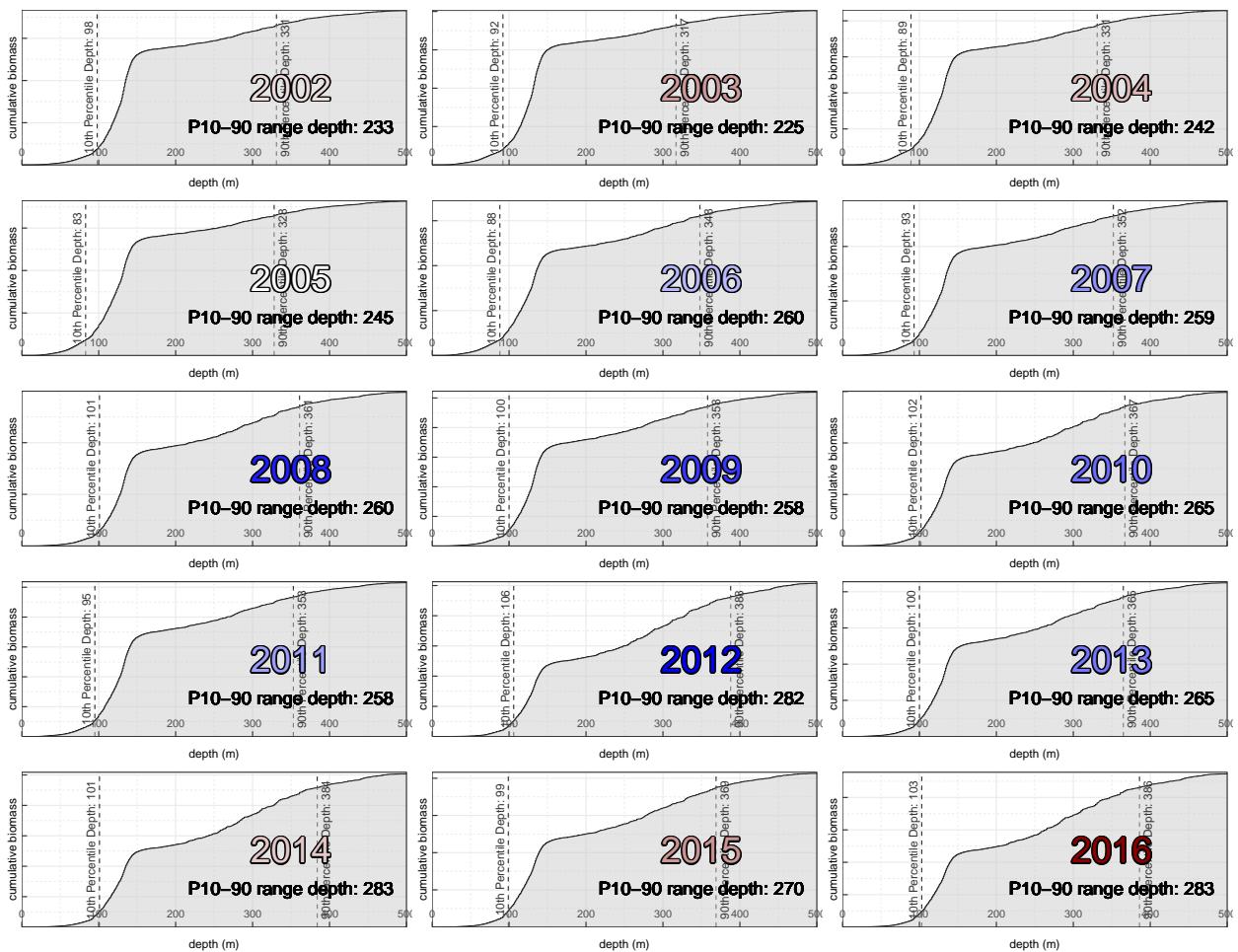
Chionoecetes bairdi



Atheresthes evermanni



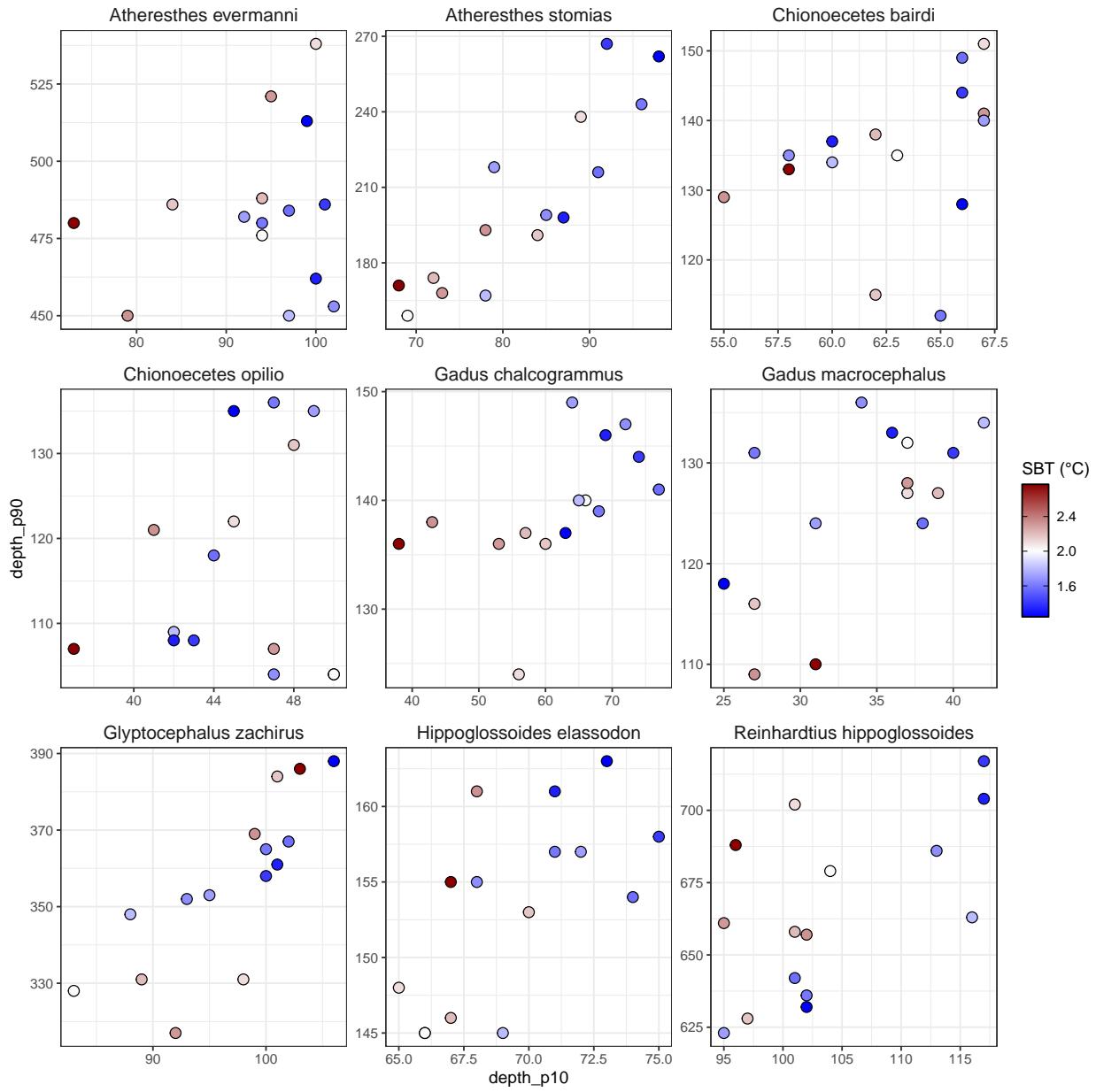
Glyptocephalus zachirus



Depth percentiles of biomass

Depth at the 10th and 90th percentiles of biomass and the range relative to SBT were analyzed. The interdecile depth range, defined as the difference between the depth at the 90th percentile and the depth at the 10th percentile, was calculated.

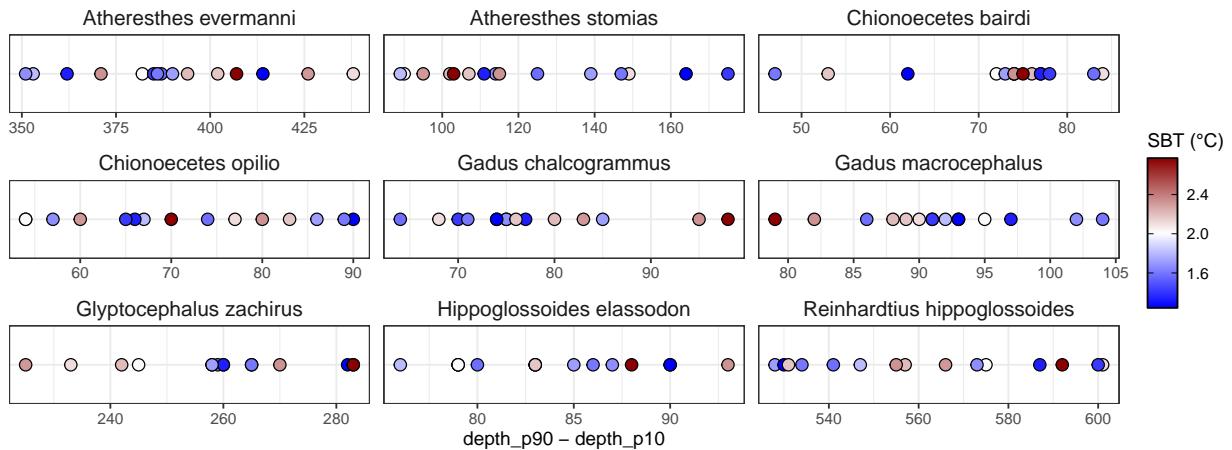
```
#plot depth range
ggplot()+
  geom_point(data=depth_percentiles,aes(x=depth_p10,y=depth_p90,fill=temp),
             shape=21,size=3)+
  facet_wrap(~species,scales = 'free')+
  scale_fill_gradientn(colors = custom_colors(100),name = 'SBT (°C)',
                       guide = guide_colorbar(frame.colour = "black",
                                              ticks.colour = "black"))+
  theme_bw()+
  theme(strip.background = element_blank(),strip.text = element_text(size=12))
```



```
#plot depth range
ggplot()+
  geom_point(data=depth_percentiles,aes(x=depth_p90-depth_p10,y=species,fill=temp),
             shape=21,size=3)+
  facet_wrap(~species,scales = 'free')+
  scale_fill_gradientn(colors = custom_colors(100),name = 'SBT (°C)',
                       guide = guide_colorbar(frame.colour = "black",
                                              ticks.colour = "black"))+
  theme_bw()+
  theme(
    axis.text.y = element_blank(),
    axis.ticks.y = element_blank(),
    axis.title.y = element_blank(),
    strip.background = element_blank(),strip.text = element_text(size=12))+
  
```

```
labs(title = 'depth at 90th percentile - depth at 10th percentile')
```

depth at 90th percentile – depth at 10th percentile



Bathymetrical COG relative to prey variables

We analyzed the correlation between prey variables (large zooplankton and large phytoplankton). We downloaded multiple ROMS variables to investigate alternative drivers of changes in the distribution of groundfish species. To do this, we used various zooplankton and phytoplankton abundance variables and calculated the mean annual value for all cells in the ROMS model of the Bering Sea, specifically focusing on abundance.

```
# Define the file ID from the shareable link
file_id <- "1y345z3DbErtCtz3nA_Xu91LStejLVntU"

# Download the grid file
suppressMessages(
  suppressWarnings(
    drive_download(as_id(file_id), path = "./grid_EBS_NBS_envs.RData",
                  overwrite = TRUE)))

#load file grid
load('./grid_EBS_NBS_envs.RData') #xx
data_preys<-aggregate(cbind(Cop,EupS,EupO,NCaS,NCaO,PhL) ~ Year,xx,
                      FUN='mean')

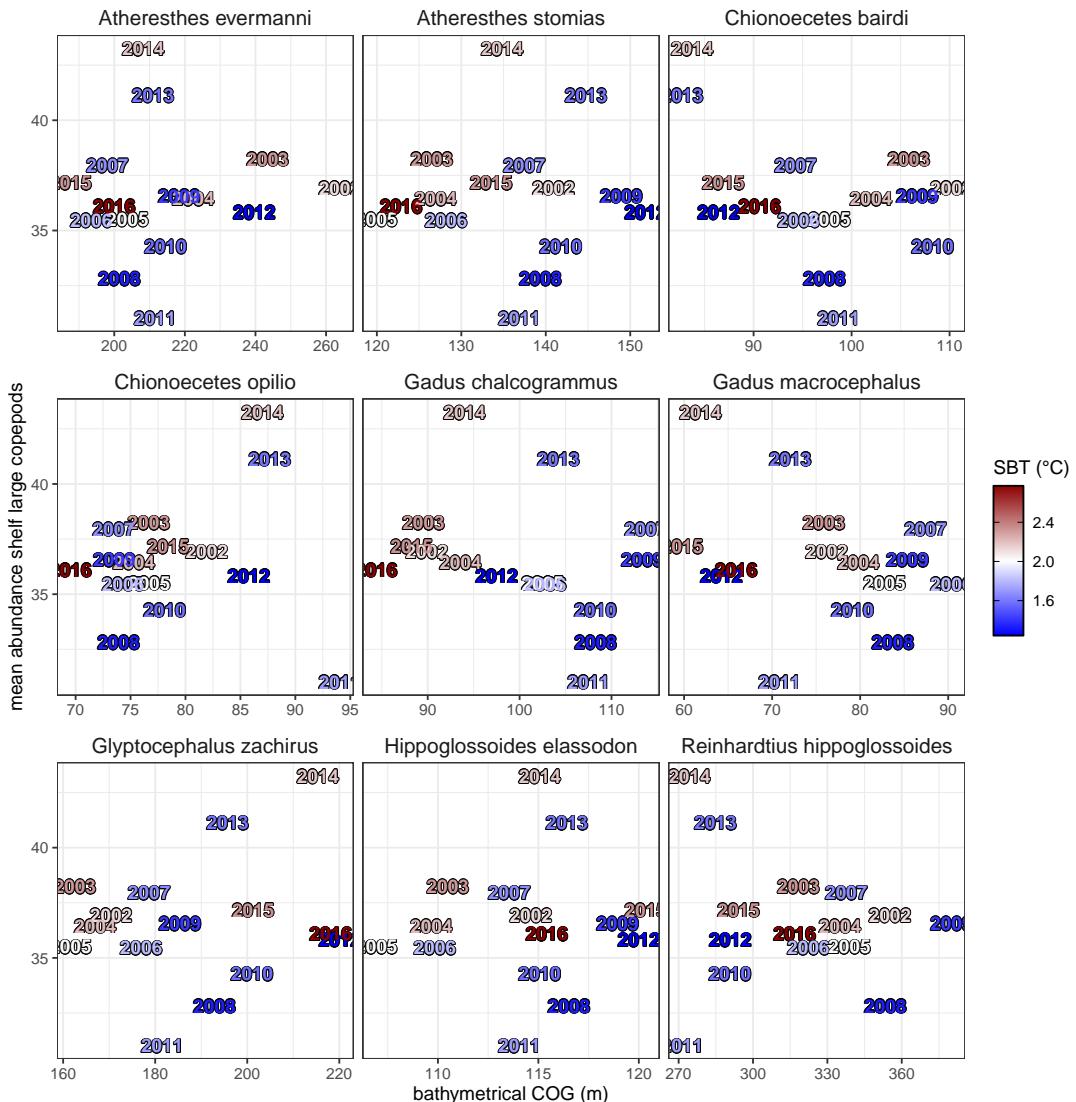
load('./metrics_df.RData') #metrics_df
cog_prey<-merge(metrics_df,data_preys,by='Year')
cog_prey1<-merge(cog_prey,subset(temp_region,region=='all'),
                 by.x=c('Year'),by.y=c('year'))

#plot using SBT color and zoopl
ggplot()+
  geom_shadowtext(data=cog_prey1,aes(x=COG_depth,y=NCaS,color=temp,label=Year),
                  fontface='bold',bg.r = 0.05)+
  theme_bw()+
  labs(y='mean abundance shelf large copepods',x='bathymetrical COG (m)')+
  theme(aspect.ratio = 1,strip.background = element_blank(),
        strip.text = element_text(size=12))+
```

```

scale_color_gradientn(colors = custom_colors(100),name = 'SBT (°C)',
                      guide = guide_colorbar(frame.colour = "black",
                                             ticks.colour = "black"))+
facet_wrap(~species,scales='free_x')

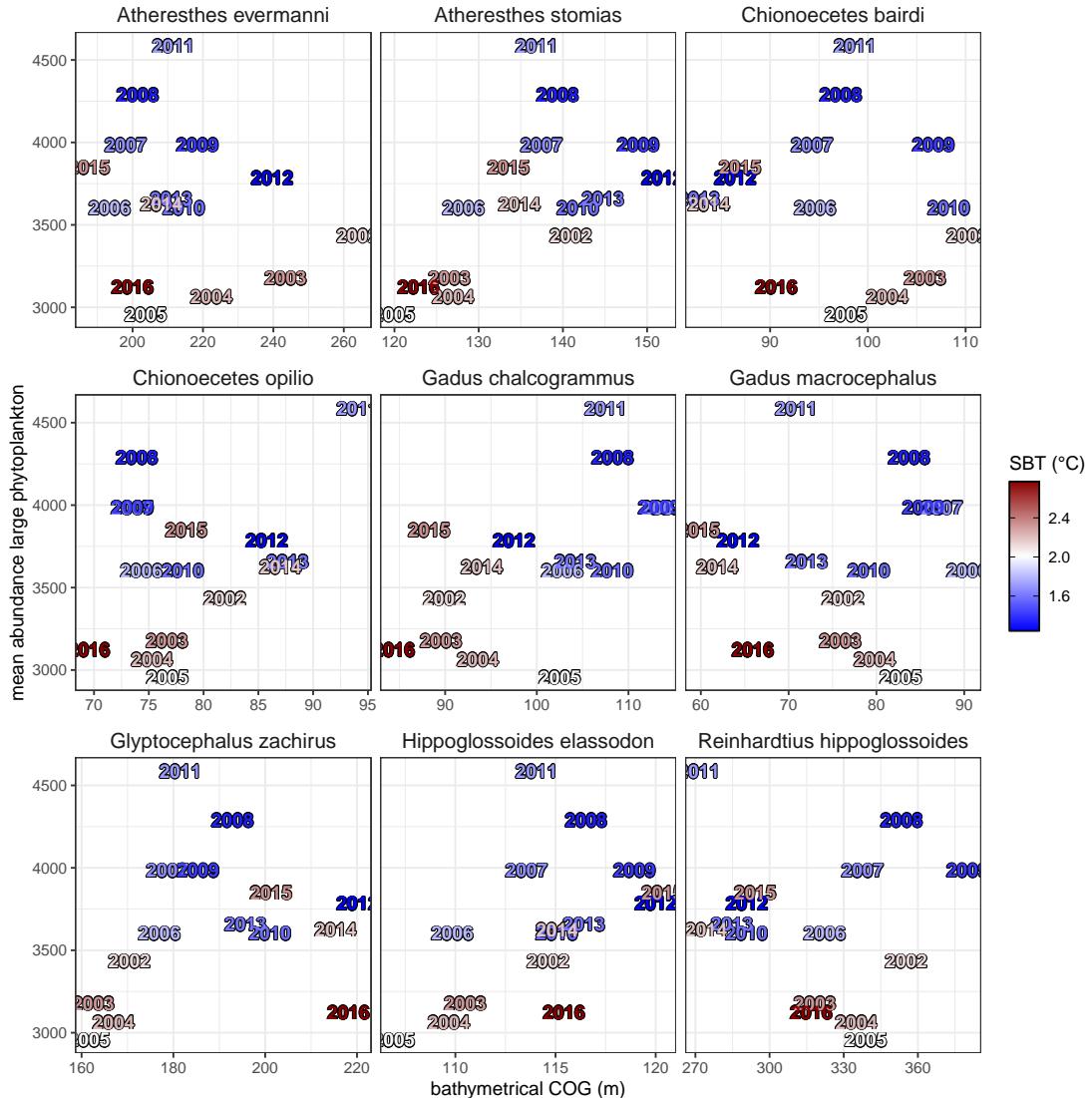
```



```

#plot using SBT color and phyto (for crabs)
ggplot()+
  geom_shadowtext(data=cog_prey1,aes(y=PhL,x=COG_depth,color=temp,label=Year),
                  fontface='bold',bg.r = 0.05)+
  theme_bw()+
  labs(y='mean abundance large phytoplankton',x='bathymetrical COG (m')+
  theme(aspect.ratio = 1,strip.background = element_blank(),
        strip.text = element_text(size=12))+
  scale_color_gradientn(colors = custom_colors(100),name = 'SBT (°C)',
                        guide = guide_colorbar(frame.colour = "black",
                                              ticks.colour = "black"))+
  facet_wrap(~species,scales='free_x')

```



Bathymetrical COG of adults

We used length data to determine the population proportion at different lengths for selected species. Different thresholds were applied for each species to distinguish between juveniles and adults. We then calculated the COG after filtering the data to include only adults.

```
# Define the file ID from the shareable link
file_id2 <- '11EqIEhkXZQUW_OP_hPgVUlwfqcyyTeQ2'

# Download the length file
suppressMessages(
  drive_download(as_id(file_id2), path = "./ak_bts_ebs_nbs_slope.rds",
                overwrite = TRUE))

#read data
data_length<-readRDS('./ak_bts_ebs_nbs_slope.rds') #data_length
load('./metrics_df.RData') #metrics_df
```

```

load('~/grid_EBS_NBS_envs.RData') #xx

#code species
spp_code<-unique(data_length$species[,c('SPECIES_CODE',"REPORT_NAME_SCIENTIFIC")])
names(spp_code)<-c('species_code',"scientific_name")
spp_code1<-spp_code[which(spp_code$scientific_name %in%
                           unique(metrics_df$species)),]

#average SBT by region and year
#get average SBT for all region (EBSshelf+NBS+EBSslope)
alltemp<-aggregate(Temp ~ Year, xx,FUN=mean)
names(alltemp)<-c('year','temp')

#data length frequency
data_size<-subset(data_length$size,SPECIES_CODE %in%
                   spp_code1$species_code)
data_catch<-subset(data_length$catch,SPECIES_CODE %in%
                    spp_code1$species_code)
data<-merge(data_size,data_length$haul,by=c('HAULJOIN','CRUISEJOIN'))

#convert time
time_axis <- as.POSIXct(data$START_TIME, origin = "1900-01-01", tz = "GMT")
data$year <- format(time_axis, "%Y")

#merge to get codes
data1<-merge(data,spp_code1,by.x='SPECIES_CODE',by.y='species_code')

# Create variable 'juv' based on species and length thresholds
data1$juv <- ifelse(data1$scientific_name %in% c("Atheresthes stomias",
                                                 "Reinhardtius hippoglossoides",
                                                 "Gadus macrocephalus") &
                           data1$LENGTH <= 300, TRUE,
                           ifelse(data1$scientific_name %in% c("Atheresthes evermanni",
                                                 "Hippoglossoides elassodon",
                                                 "Glyptocephalus zachirus") &
                                   data1$LENGTH <= 200, TRUE,
                                   ifelse(data1$scientific_name %in% c("Gadus chalcogrammus") &
                                         data1$LENGTH <= 150, TRUE, FALSE)))

#to percentage
data2 <- data1 %>%
  group_by(year,scientific_name) %>%
  mutate(PERCENTAGE = FREQUENCY / sum(FREQUENCY)) %>%
  ungroup() # remove grouping

#add \n
data2$scientific_name<-gsub(' ','\n',data2$scientific_name)

#population size structure
ggplot()+
  geom_bar(data=data2,aes(x=LENGTH,y=PERCENTAGE,fill=juv),stat='identity')+
  facet_grid(year ~ scientific_name,scales='free_y')+

```

```

theme_minimal()+
scale_fill_manual(values = c("FALSE" = "#5dc3b3", "TRUE" = "grey"),
                  labels = c("FALSE" = "Adult", "TRUE" = "Juvenile")) +
labs(y='proportion',x='length (mm)',title = 'Proportion at length')+
theme(axis.text.y = element_blank(),strip.text.y.right = element_text(angle=360),
      legend.title = element_blank())

```

Proportion at length



```

#subset adults
sum_len2<-subset(data1,juv==FALSE)

#to calculate COG
sum_len2$freq_lat<-sum_len2$FREQUENCY*sum_len2$START_LATITUDE
sum_len2$freq_depth<-sum_len2$FREQUENCY*sum_len2$BOTTOM_DEPTH

#sum by years
sum_len3<-aggregate(FREQUENCY ~ year + scientific_name,sum_len2,FUN=sum)
sum_len3d<-aggregate(freq_depth ~ year + scientific_name,sum_len2,FUN=sum)

```

```

sum_len3l<-aggregate(freq_lat ~ year + scientific_name,sum_len2,FUN=sum)

#calculate metrics
data_len<-merge(sum_len3,sum_len3d,by=c('year','scientific_name'))
data_len1<-merge(data_len,sum_len3l,by=c('year','scientific_name'))
data_len1$COG_depth<-data_len1$freq_depth/data_len1$FREQUENCY
data_len1$COG_lat<-data_len1$freq_lat/data_len1$FREQUENCY

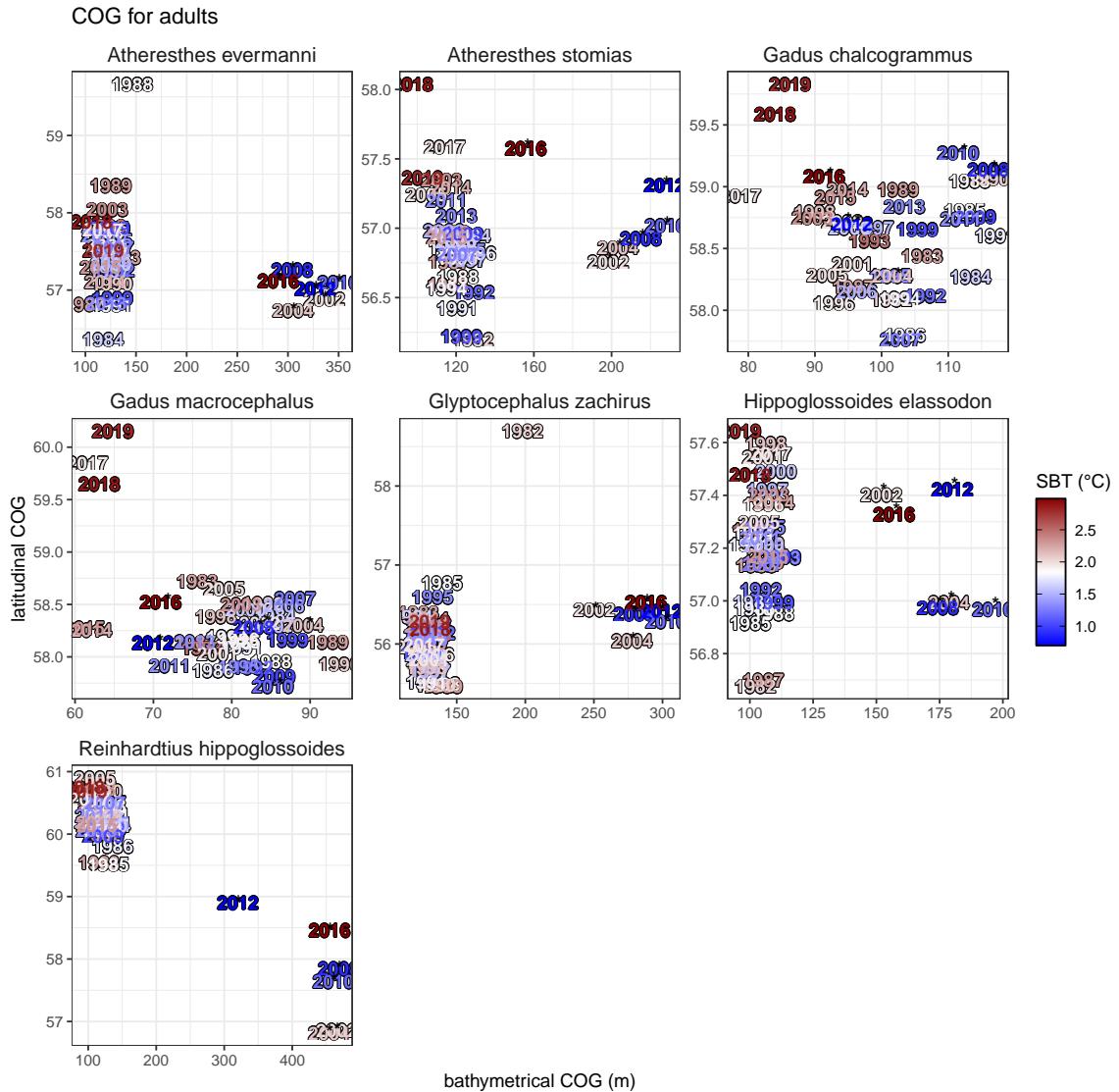
#merge with annual SBT
data_len2<-merge(data_len1,alltemp,by='year')

#Define a custom color scale function
custom_colors <- colorRampPalette(c("blue", "white", "darkred"))

#yrs with slope observations
slp_yrs<-c(2002,2004,2008,2010,2012,2016)

#plot using SBT color metrics by adults
ggplot()+
  geom_shadowtext(data=data_len2,aes(x=COG_depth,y=COG_lat,color=temp,label=year),
                  fontface='bold',bg.r = 0.05)+
  geom_text(data=subset(data_len2,year %in% slp_yrs),aes(x=COG_depth,y=COG_lat),
            label='*',color='black',fontface='bold',nudge_x = 0.95,nudge_y = 0.02)+ 
  theme_bw()+
  labs(y='latitudinal COG',x='bathymetrical COG (m)',title = 'COG for adults')+
  theme(aspect.ratio = 1,strip.background = element_blank(),
        strip.text = element_text(size=12))+
  scale_color_gradientn(colors = custom_colors(100),name = 'SBT (°C)',
                        guide = guide_colorbar(frame.colour = "black",
                                              ticks.colour = "black"))+
  facet_wrap(~scientific_name,scales='free')

```



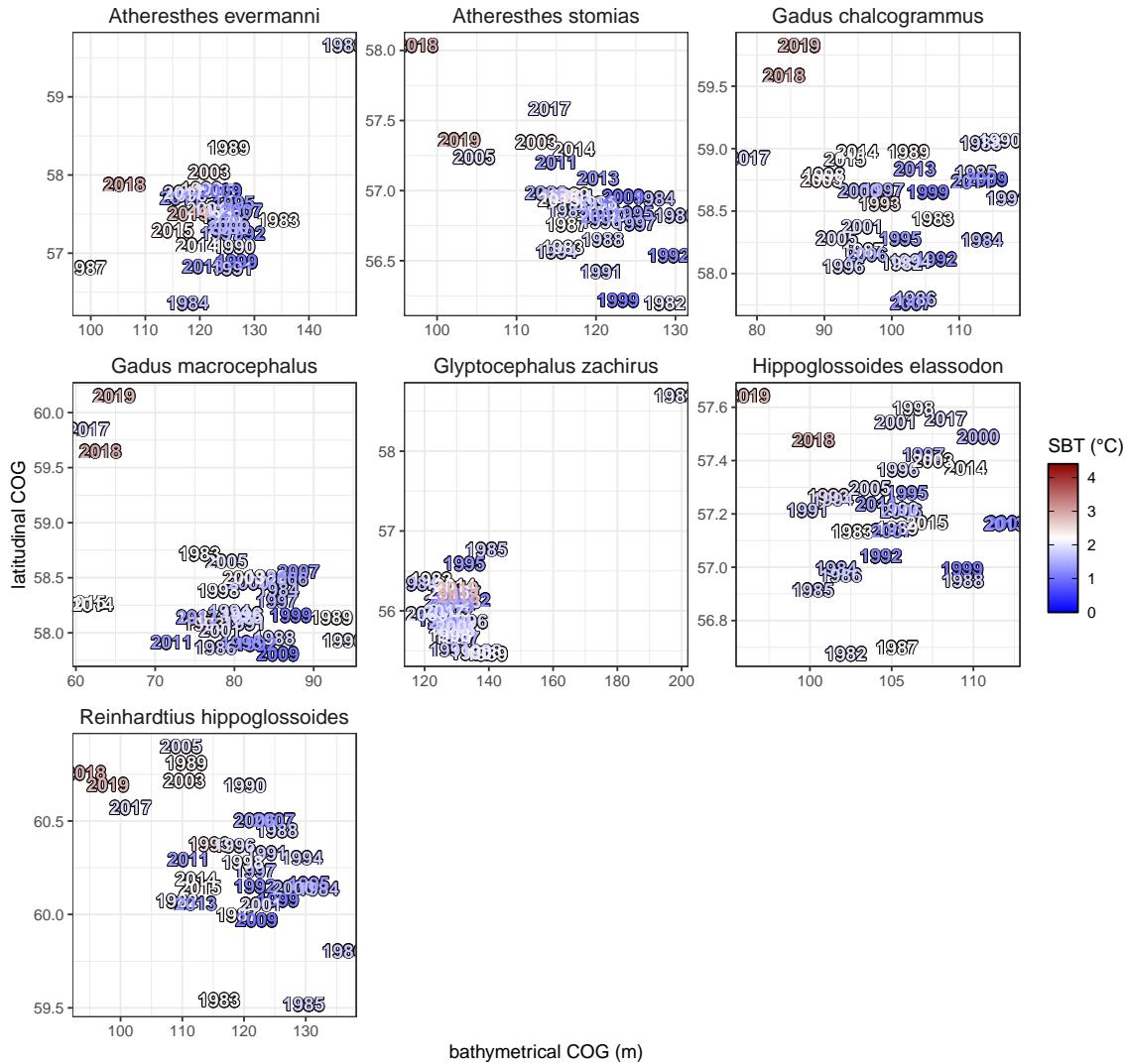
Bathymetrical COG of adults excluding slope data

Latitudinal COG relative to bathymetrical COG using adult data relative to SBT in the regions

```
#plot without slope years
ggplot()+
  geom_shadowtext(data=subset(data_len2,year %in% setdiff(1982:2022,slp_yrs)),
                  aes(x=COG_depth,y=COG_lat,color=temp,label=year),
                  fontface='bold',bg.r = 0.05)+
  theme_bw()+
  labs(y='latitudinal COG',x='bathymetrical COG (m)',
       title = 'COG for adults excluding years with slope data')+
  theme(aspect.ratio = 1,strip.background = element_blank(),
        strip.text = element_text(size=12))+
  scale_color_gradientn(colors = custom_colors(100),name = 'SBT (°C)',
                        guide = guide_colorbar(frame.colour = "black",
                                              ticks.colour = "black"),limits=c(0,4.4))+
```

```
facet_wrap(~scientific_name, scales='free')
```

COG for adults excluding years with slope data



Latitudinal COG relative to bathymetrical COG using adult data relative to SBT in the regions. Slope years were removed to avoid bias towards slope years for the bathymetrical COG.