

GOA: Gulf of Alaska survey data processing summary

fishglob, Aurore A. Maureaud, Juliano Palacios Abrantes, Zoë Kitchel, Dan Forrest, & Michelle Stuart

January, 2023

Contents

General info	1
Data cleaning in R	1
1. Overview of the survey data table	10
2. Summary of sampling intensity	11
3. Summary of sampling variables from the survey	12
4. Summary of biological variables	13
5. Extreme values	14
6. Summary of variables against swept area	15
7. Abundance or Weight trends of the six most abundant species	16
8. Distribution mapping	17
9. Taxonomic flagging	18
10. Spatio-temporal standardization	19
a. Standardization method 1	19
b. Standardization method 2	22
c. Standardization summary	22

General info

This document presents the summary of the Gulf of Alaska bottom trawl survey provided by Stan Kotwicki and Jim Thorson. It contains data from 1984-1999 (triennial) and 2001-2019 (biennial).

Data cleaning in R

```
#####
#### R code to clean trawl survey Gulf of Alaska
#### Public data Ocean Adapt
#### Contacts: Stan Kotwicki stan.kotwicki@noaa.gov Program Manager
####           Groundfish Assessment Program, NOAA AFSC
####           Jim Thorson james.thorson@noaa.gov Program Leader
####           Habitat and Ecological Processes Research, NOAA AFSC
#### Coding: Michelle Stuart, Dan Forrest, Zoë Kitchel September 2021
#####
#Alaska Fisheries Science Center - NOAA
#https://www.afsc.noaa.gov/RACE/groundfish/survey\_data/metadata\_template.php
#?fname=RACEweb.xml
#This NOAA center provides data for the Aleutian Islands, Eastern Bering Sea,
#and Gulf of Alaska.
#Files provided by the Alaska Fisheries Science Center
```

#NOTES

```

##From data providers: - Any species name values that contain the word
##"Lepidopsetta" are changed to "Lepidopsetta sp." because more than one genus/spp
##combo was used to describe the same organism over time. This also holds true
##for Myoxocephalus sp. excluding scorpius and Bathyraja sp. excluding panthera.
##Therefore, the final step in this code is to sum biomass within hauls

#-----#
#### LOAD LIBRARIES AND FUNCTIONS #####
#-----#


library(tidyverse)
library(lubridate)
library(googledrive)
library(taxize) # for getting correct species names
library(magrittr) # for names wrangling

source("functions/clean_taxa.R")
source("functions/write_clean_data.R")
source("functions/apply_trimming_method1.R")
source("functions/apply_trimming_method2.R")
source("functions/flag_spp.R")
fishglob_data_columns <- read_excel("standard_formats/fishglob_data_columns.xlsx")

#Data for the Gulf of Alaska can be accessed using the public Pinsky
#Lab OceanAdapt Git Hub Repository.
#Contact malin.pinsky@rutgers.edu for questions or help accessing

#-----#
#### PULL IN AND EDIT RAW DATA FILES #####
#-----#


#make list of csv files from OceanAdapt GitHub
files <- list(
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/goa1984_1987.csv",
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/goa1990_1999.csv",
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/goa2001_2005.csv",
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/goa2007_2013.csv",
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/goa2015_2019.csv")

# combine all of the data files into one table
goa_data <- files %>%
  # read in all of the csvs in the files list
  map_dfr(read_csv) %>%
  # remove any data rows that have headers as data rows
  filter(LATITUDE != "LATITUDE", !is.na(LATITUDE)) %>%
  mutate(stratum = as.integer(STRATUM))

# import the strata data
file <-
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/goa_strata.csv"

```

```

goa_strata <- file %>%
  # read in all of the csv's in the files list
  map_dfr(read_csv) %>%
  #select(StratumCode, Areakm2) %>%
  distinct() %>%
  rename(stratum = StratumCode)

#-----#
##### REFORMAT AND MERGE DATA FILES #####
#-----#


goa <- left_join(goa_data, goa_strata, by = "stratum")
#link strata DF and data DF by stratum column

# are there any strata in the data that are not in the strata file?
stopifnot(nrow(filter(goa, is.na(Areakm2))) == 0) #clear

#edit columns
goa <- goa %>%
  mutate(
    # Create a unique haul_id
    haul_id = paste(formatC(VESSEL, width=3, flag=0), CRUISE,
                    formatC(HAUL, width=3, flag=0), LATITUDE, LATITUDE, sep=''),
    #get rid of any use of -9999 as a no data marker
    numcpue = ifelse(NUMCPUE < -9000, NA, NUMCPUE),
    sbt = ifelse(BOT_TEMP < -9000, NA, BOT_TEMP),
    sst = ifelse(SURF_TEMP < -9000, NA, SURF_TEMP)) %>%
  rename(year = YEAR,
         latitude = LATITUDE,
         longitude = LONGITUDE,
         depth = BOT_DEPTH,
         spp = SCIENTIFIC,
         station = STATION,
         num_cpue.raw = numcpue, #units = number/hectare
         wgt_cpue.raw = WTCPUE #units = kg/hectare (1 hectare = 0.01 km^2)
     ) %>%
  mutate(
    #convert date to month and day columns
    datetime = mdy_hm(DATETIME),
    month = month(datetime),
    day = day(datetime),
    quarter = case_when(month %in% c(1,2,3) ~ 1,
                         month %in% c(4,5,6) ~ 2,
                         month %in% c(7,8,9) ~ 3,
                         month %in% c(10,11,12) ~ 4),
    season = 'NA',
    #convert cpue which is currently per hectare to per km^2 by multiplying by 100
    wgt_cpue = 100*wgt_cpue.raw,
    num_cpue = 100*num_cpue.raw
  ) %>%
  # remove non-fish
  filter(
    spp != '' &

```

```

!grepl("egg", spp)) %>%
# adjust spp names
mutate(
  #Manual taxa cleaning (happens later in other get.x.R scripts) *see notes at top for
  #further explanation for next 5 lines

  spp = ifelse(grepl("Lepidopsetta", spp), "Lepidopsetta sp.", spp),
  spp = ifelse(grepl("Myoxocephalus", spp) & !grepl("scorpius", spp),
               "Myoxocephalus sp.", spp),
  spp = ifelse(grepl("Bathyraja", spp) & !grepl("panthera", spp),
               'Bathyraja sp.', spp)
) %>%
#finalize columns
mutate(survey = "GOA",
       source = "NOAA",
       timestamp = mdy("03/01/2021"),
       country = "United States",
       sub_area = NA,
       continent = "n_america",
       stat_rec = NA,
       verbatim_name = spp,
       haul_dur = NA,
       gear = NA,
       num = NA,
       num_h = NA,
       wgt = NA,
       wgt_h = NA,
       area_swept = NA
) %>%
select(survey, haul_id, source, timestamp, country, sub_area, continent, stat_rec, station,
       stratum, year, month, day, quarter, season, latitude, longitude, haul_dur,
       area_swept, gear, depth, sbt, sst,
       num, num_h, num_cpue, wgt, wgt_h, wgt_cpue, verbatim_name)

#sum duplicates
goa <- goa %>%
  group_by(survey,
            source,timestamp,
            haul_id, country, sub_area, continent, stat_rec, station, stratum,
            year, month, day, quarter, season, latitude, longitude, haul_dur, area_swept,
            gear, depth, sbt, sst, verbatim_name) %>%
  summarise(num = sum(num, na.rm = T),
            num_h = sum(num_h, na.rm = T),
            num_cpue = sum(num_cpue, na.rm = T),
            wgt = sum(wgt, na.rm = T),
            wgt_h = sum(wgt_h, na.rm = T),
            wgt_cpue = sum(wgt_cpue, na.rm = T)) %>% ungroup()

#check for duplicates, should not be any with more than 1 obs
#check for duplicates
count_goa <- goa %>%
  group_by(haul_id, verbatim_name) %>%
  mutate(count = n())

```

```

#none!

#which ones are duplicated?
unique_name_match <- count_goa %>%
  group_by(verbatim_name) %>%
  filter(count>1) %>%
  distinct(verbatim_name)

unique_name_match
#empty

#duplicated (will just sum abundance and biomass)
#1 Bathyraja sp.
#2 Myoxocephalus sp.
#3 Lepidopsetta sp.
#4 Aphrocallystes vastus (only 2 times in total, all in 1996)

#-----#
##### INTEGRATE CLEAN TAXA FROM TAXA ANALYSIS #####
#-----#


# Get WoRM's id for sourcing
wrn <- gnr_datasources() %>%
  filter(title == "World Register of Marine Species") %>%
  pull(id)

### Automatic cleaning
# Set Survey code
goa_survey_code <- "GOA"

goa <- goa %>%
  mutate(
    taxa2 = str_squish(verbatim_name),
    taxa2 = str_remove_all(taxa2, " spp.| sp.| spp| sp|NO "),
    taxa2 = str_to_sentence(str_to_lower(taxa2)))

# Get clean taxa
clean_auto <- clean_taxa(unique(goа$taxa2), input_survey = goa_survey_code,
                           save = F, output=NA, fishbase=T) # takes 7.5 min

#Check those with no match from clean_taxa()
#Cheiraster dawsoni                                no match
#Crangon communis                                 no match
#Scalpellum cornutum                               no match
#Cancer gracilis                                  no match
#Nearchester pedicellaris                         no match
#Bathybuccinum clarki                            no match
#Nearchester variabilis                           no match
#Cancer branneri                                 no match
#Pandalopsis                                     no match
#Nearchester aciculosus                           no match
#Beringius beringii                                no match
#Crangon abyssorum                                no match

```

```

#####clear, all invertebrates

#-----#
##### INTEGRATE CLEAN TAXA in SURVEY survey data #####
#-----#


clean_taxa <- clean_auto %>%
  select(-survey)

clean_goa <- left_join(goa, clean_taxa, by=c("taxa2"="query")) %>%
  filter(!is.na(taxa)) %>% # query does not indicate taxa entry that were
#removed in the cleaning procedure
# so all NA taxa have to be removed from the surveys because: non-existing,
#non marine or non fish
  rename(accepted_name = taxa,
         aphia_id = worms_id) %>%
  mutate(verbatim_aphia_id = NA,
        num_cpua = num_cpue,
        num_cpue = num_h,
        wgt_cpua = wgt_cpue,
        wgt_cpue = wgt_h,
        survey_unit = ifelse(survey %in% c("BITS","NS-IBTS","SWC-IBTS"),
                             paste0(survey, "-", quarter), survey),
        survey_unit = ifelse(survey %in% c("NEUS","SEUS","SCS","GMEX"),
                             paste0(survey, "-", season), survey_unit)) %>%
  select(fishglob_data_columns$`Column name fishglob`)

#check for duplicates
count_clean_goa <- clean_goa %>%
  group_by(haul_id, accepted_name) %>%
  mutate(count = n())

#none!

#which ones are duplicated?
unique_name_match <- count_clean_goa %>%
  group_by(verbatim_name, accepted_name) %>%
  filter(count>1) %>%
  distinct(verbatim_name, accepted_name)

unique_name_match
#not empty

#there is one duplicate that we will keep. up to user whether or not they want to merge.
# Groups: verbatim_name, accepted_name [2]
# verbatim_name                                         accepted_name
#Platichthys stellatus                               Platichthys stellatus
#Platichthys stellatus X Pleuronectes quadrifasciatus hybrid Platichthys stellatus

# -----#
##### SAVE DATABASE IN GOOGLE DRIVE #####

```

```

# -----#
# Just run this routine should be good for all
write_clean_data(data = clean_goa, survey = "GOA", overwrite = T, type = F)

# -----#
##### FAGS #####
# -----#
#install required packages that are not already installed
required_packages <- c("data.table",
                      "devtools",
                      "dgridR",
                      "dplyr",
                      "fields",
                      "forcats",
                      "ggplot2",
                      "here",
                      "magrittr",
                      "maps",
                      "maptools",
                      "raster",
                      "rcompendium",
                      "readr",
                      "remotes",
                      "rrtools",
                      "sf",
                      "sp",
                      "tidyverse",
                      "usethis")

not_installed <- required_packages[!(required_packages %in% installed.packages()[, "Package"])]
if(length(not_installed)) install.packages(not_installed)

#load pipe operator
library(magrittr)

##### Apply taxonomic flagging per region
#get vector of regions (here the survey column)
regions <- levels(as.factor(clean_goa$survey))

#run flag_spp function in a loop
for (r in regions) {
  flag_spp(clean_goa, r)
}

##### Apply trimming per survey_unit method 1
#apply trimming for hex size 7
dat_new_method1_hex7 <- apply_trimming_per_survey_unit_method1(clean_goa, 7)

#apply trimming for hex size 8

```

```

dat_new_method1_hex8 <- apply_trimming_per_survey_unit_method1(clean_goa, 8)

##### Apply trimming per survey unit method 2
dat_new_method2 <- apply_trimming_per_survey_unit_method2(clean_goa)

#-----#
#### ADD STANDARDIZATION FLAGS #####
#-----#
surveys <- sort(unique(clean_goa$survey))
survey_units <- sort(unique(clean_goa$survey_unit))
survey_std <- clean_goa %>%
  mutate(flag_taxa = NA_character_,
        flag_trimming_hex7_0 = NA_character_,
        flag_trimming_hex7_2 = NA_character_,
        flag_trimming_hex8_0 = NA_character_,
        flag_trimming_hex8_2 = NA_character_,
        flag_trimming_2 = NA_character_)

# integrate taxonomic flags
for(i in 1:length(surveys)){
  if(!surveys[i] %in% c("FALK", "GSL-N", "MRT", "NZ-CHAT", "SCS", "SWC-IBTS")){
    xx <- data.frame(read_delim(paste0("outputs/Flags/taxonomic_flagging/",
                                         surveys[i], "_flagspp.txt"),
                                 delim=";", escape_double = FALSE, col_names = FALSE,
                                 trim_ws = TRUE))
    xx <- as.vector(unlist(xx[1,]))

    survey_std <- survey_std %>%
      mutate(flag_taxa = ifelse(survey == surveys[i] & accepted_name %in% xx,
                                "TRUE", flag_taxa))

    rm(xx)
  }
}

# integrate spatio-temporal flags
for(i in 1:length(survey_units)){
  if(!survey_units[i] %in% c("DFO-SOG", "IS-TAU", "SCS-FALL", "WBLS")){
    hex_res7_0 <- read.csv(paste0("outputs/Flags/trimming_method1/hex_res7/",
                                   survey_units[i], "_hex_res_7_trimming_0_hauls_removed.csv"),
                            sep = ";")
    hex_res7_0 <- as.vector(hex_res7_0[,1])

    hex_res7_2 <- read.csv(paste0("outputs/Flags/trimming_method1/hex_res7/",
                                   survey_units[i], "_hex_res_7_trimming_02_hauls_removed.csv"),
                            sep = ";")
    hex_res7_2 <- as.vector(hex_res7_2[,1])

    hex_res8_0 <- read.csv(paste0("outputs/Flags/trimming_method1/hex_res8",
                                   survey_units[i], "_hex_res_8_trimming_0_hauls_removed.csv"),
                            sep = ";")
  }
}

```

```

            sep= ";")
hex_res8_0 <- as.vector(hex_res8_0[,1])

hex_res8_2 <- read.csv(paste0("outputs/Flags/trimming_method1/hex_res8/",
                               survey_units[i], "_hex_res_8_trimming_02_hauls_removed.csv"),
                        sep = ";")
hex_res8_2 <- as.vector(hex_res8_2[,1])

trim_2 <- read.csv(paste0("outputs/Flags/trimming_method2/",
                           survey_units[i],"_hauls_removed.csv"))
trim_2 <- as.vector(trim_2[,1])

survey_std <- survey_std %>%
  mutate(flag_trimming_hex7_0 = ifelse(survey_unit == survey_units[i] & haul_id %in% hex_res7_0,
                                         "TRUE",flag_trimming_hex7_0),
         flag_trimming_hex7_2 = ifelse(survey_unit == survey_units[i] & haul_id %in% hex_res7_2,
                                         "TRUE",flag_trimming_hex7_2),
         flag_trimming_hex8_0 = ifelse(survey_unit == survey_units[i] & haul_id %in% hex_res8_0,
                                         "TRUE",flag_trimming_hex8_0),
         flag_trimming_hex8_2 = ifelse(survey_unit == survey_units[i] & haul_id %in% hex_res8_2,
                                         "TRUE",flag_trimming_hex8_2),
         flag_trimming_2 = ifelse(survey_unit == survey_units[i] & haul_id %in% trim_2,
                                         "TRUE", flag_trimming_2)
      )
rm(hex_res7_0, hex_res7_2, hex_res8_0, hex_res8_2, trim_2)
}

# Just run this routine should be good for all
write_clean_data(data = survey_std, survey = "GOA_std",
                 overwrite = T, rdata=TRUE)
}

```

1. Overview of the survey data table

survey	source	timestamp	haul_id		country	sub_area
GOA	NOAA	2021-03-01	019199001003-166.2406753.48167	United States	NA	
GOA	NOAA	2021-03-01	019199001003-166.2406753.48167	United States	NA	
GOA	NOAA	2021-03-01	019199001003-166.2406753.48167	United States	NA	
GOA	NOAA	2021-03-01	019199001003-166.2406753.48167	United States	NA	
GOA	NOAA	2021-03-01	019199001003-166.2406753.48167	United States	NA	

continent	stat_rec	station	stratum	year	month	day	quarter	season
n_america	NA	45-29	111	1990	6	4	2	NA
n_america	NA	45-29	111	1990	6	4	2	NA
n_america	NA	45-29	111	1990	6	4	2	NA
n_america	NA	45-29	111	1990	6	4	2	NA
n_america	NA	45-29	111	1990	6	4	2	NA

latitude	longitude	haul_dur	area_swept	gear	depth	sbt	sst
53.48167	-166.2407	NA	NA	NA	134	NA	7
53.48167	-166.2407	NA	NA	NA	134	NA	7
53.48167	-166.2407	NA	NA	NA	134	NA	7
53.48167	-166.2407	NA	NA	NA	134	NA	7
53.48167	-166.2407	NA	NA	NA	134	NA	7

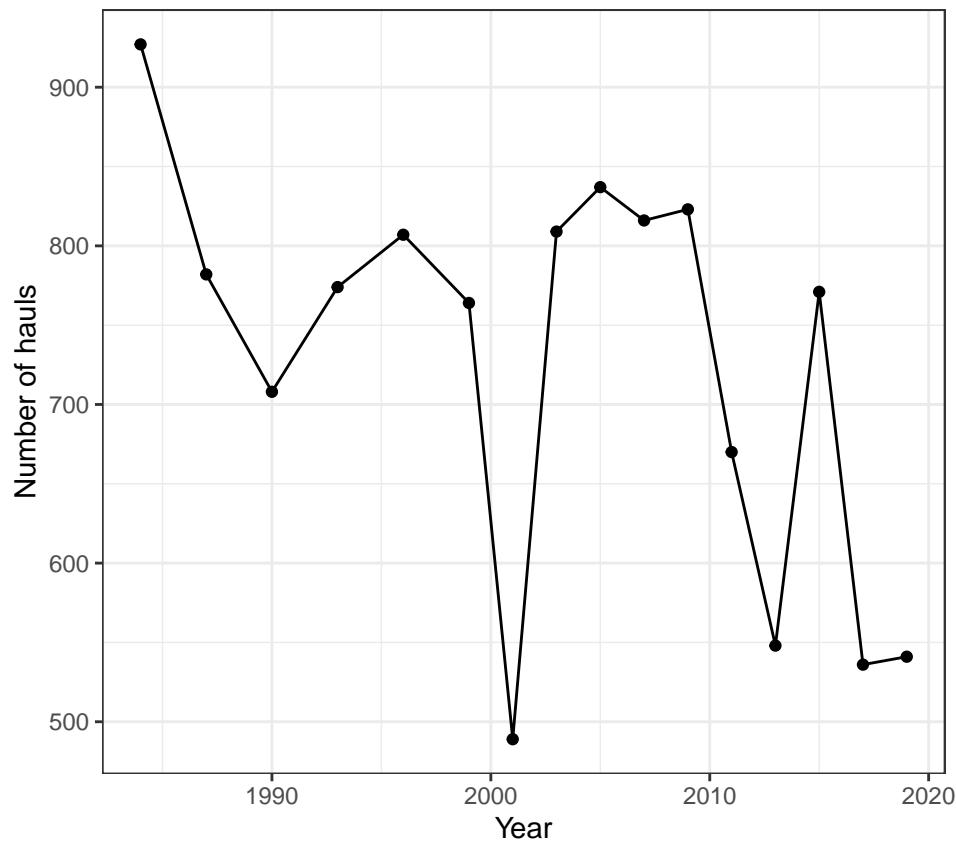
num	num_cpue	num_cputa	wgt	wgt_cpue	wgt_cputa	verbatim_name
0	0	20396.79	0	0	10940.13	Atheresthes stomias
0	0	29.78	0	0	405.20	Beringraja binoculata
0	0	59.55	0	0	40.53	Gadus chalcogrammus
0	0	595.53	0	0	1404.67	Gadus macrocephalus
0	0	655.08	0	0	405.20	Glyptocephalus zachirus

verbatim_aphia_id	accepted_name	aphia_id	SpecCode	kingdom
NA	Atheresthes stomias	279792	517	Animalia
NA	Beringraja binoculata	1021330	2556	Animalia
NA	Gadus chalcogrammus	300735	318	Animalia
NA	Gadus macrocephalus	254538	308	Animalia
NA	Glyptocephalus zachirus	274287	4238	Animalia

phylum	class	order	family	genus	rank	survey_unit
Chordata	Teleostei	Pleuronectiformes	Pleuronectidae	Atheresthes	Species	GOA
Chordata	Elasmobranchii	Rajiformes	Rajidae	Beringraja	Species	GOA
Chordata	Teleostei	Gadiformes	Gadidae	Gadus	Species	GOA
Chordata	Teleostei	Gadiformes	Gadidae	Gadus	Species	GOA
Chordata	Teleostei	Pleuronectiformes	Pleuronectidae	Glyptocephalus	Species	GOA

2. Summary of sampling intensity

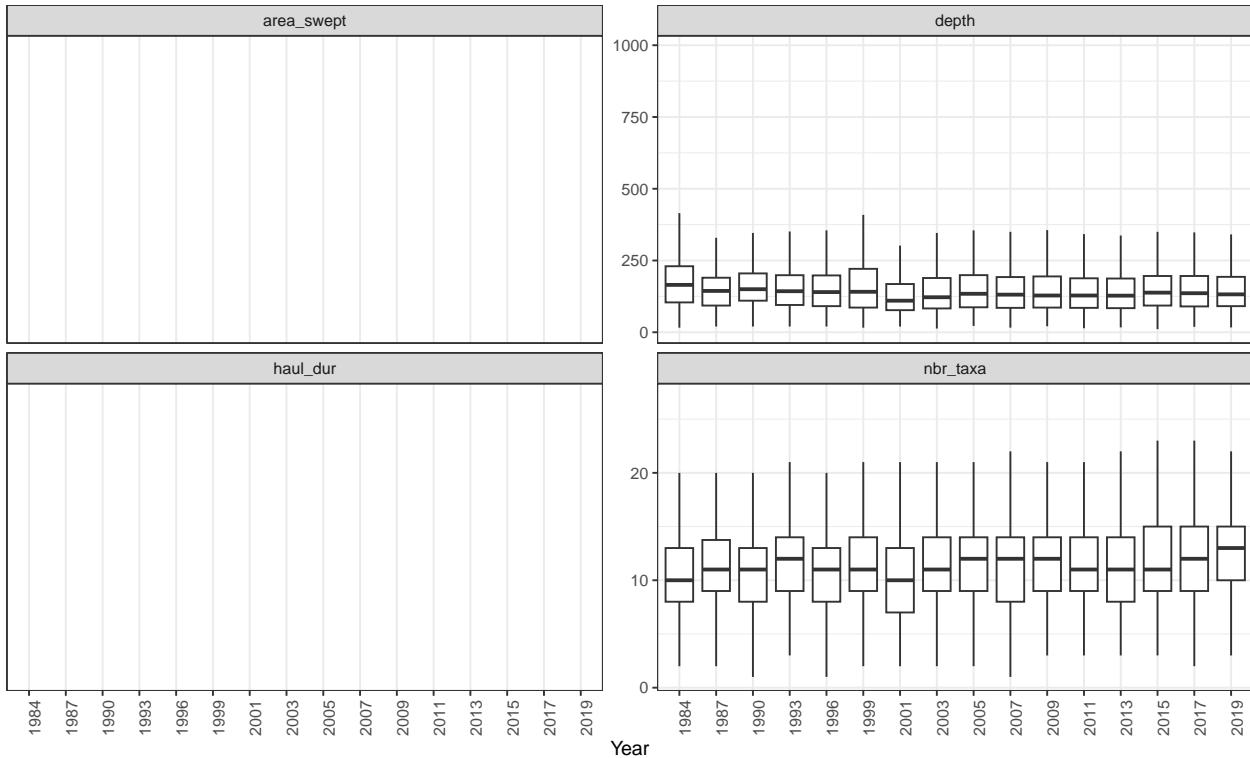
Number of hauls per year performed during the survey after data processing.



3. Summary of sampling variables from the survey

Here we show the yearly total and average of the following variables reported in the survey data:

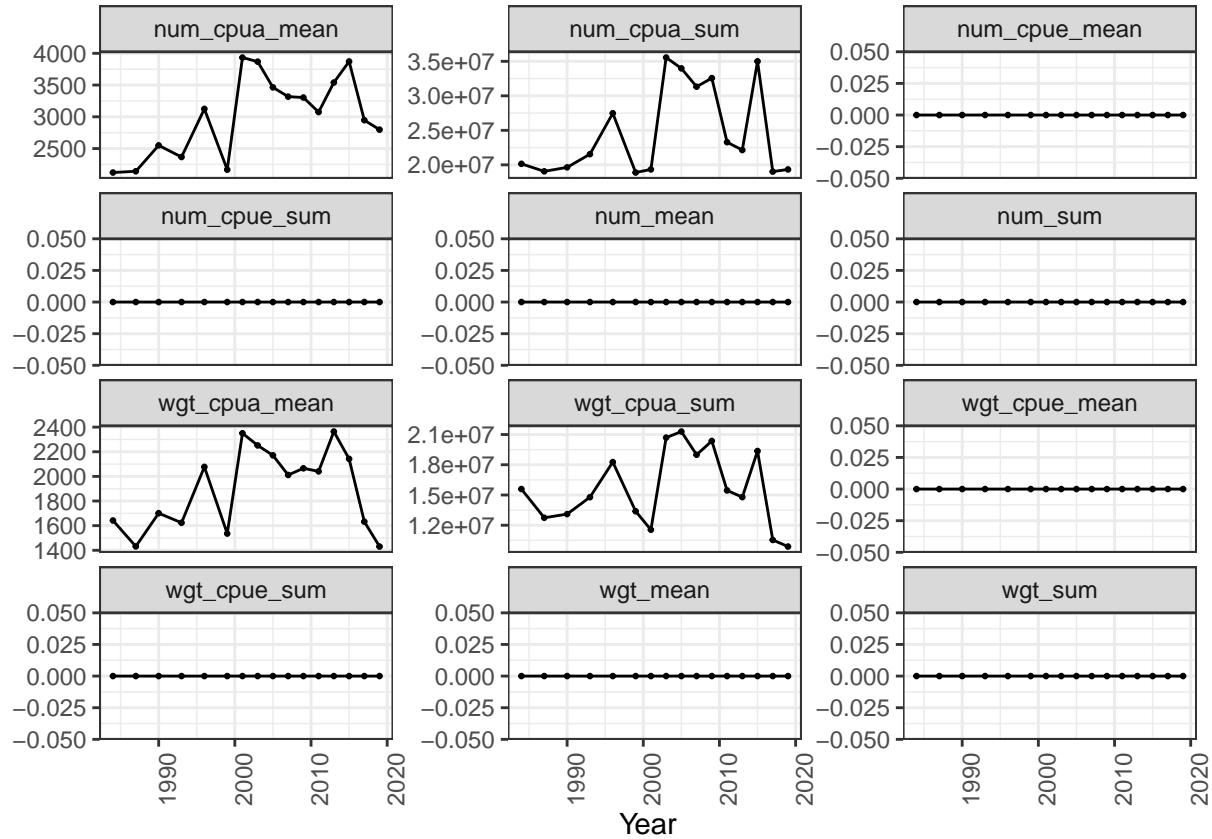
- *area_swept*, swept area by the bottom trawl gear km^2
- *depth*, sampling depth in m
- *haul_dur*, haul sampling duration *hour*
- *number of marine fish taxa*, taxa were cleaned following the last version of taxonomy from the World Register of Marine Species (<https://www.marinespecies.org/>, October 2021)



4. Summary of biological variables

Here we display the yearly total and average across hauls of the following variables recorded in the data:

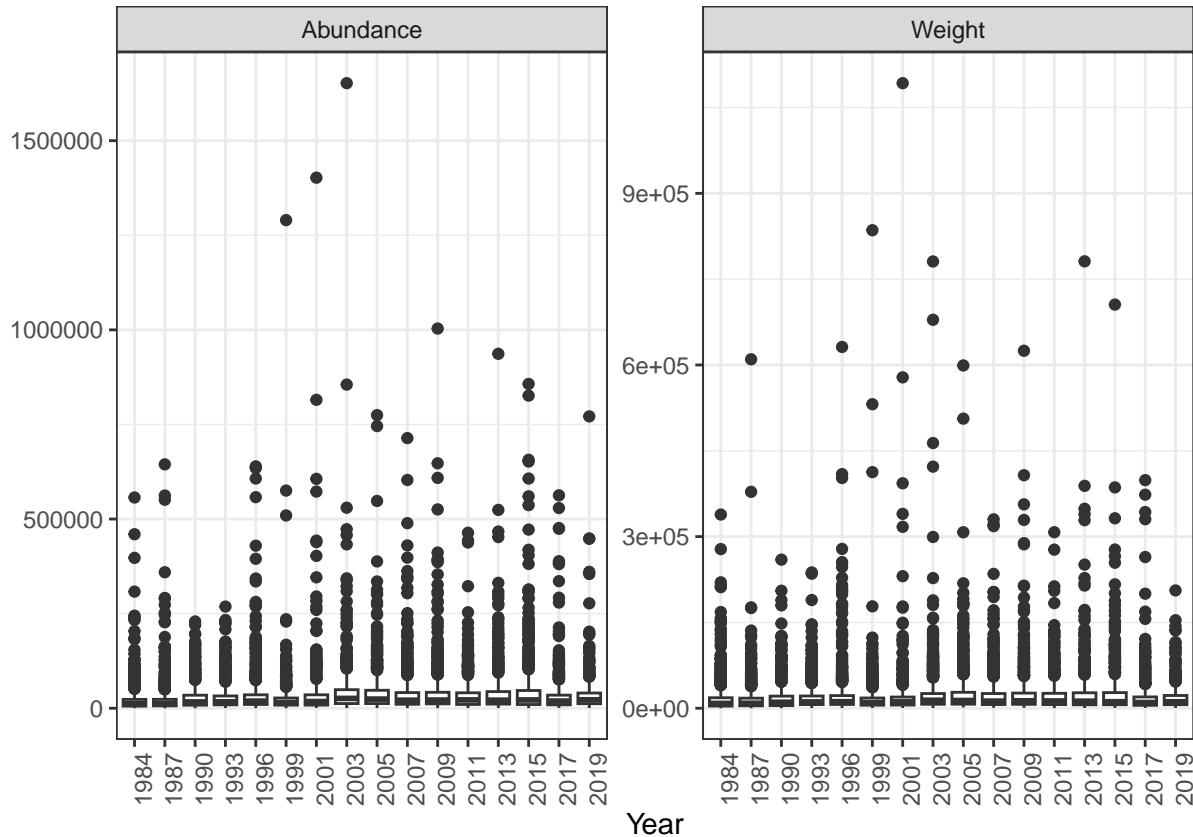
- num_cpua , number of individuals (abundance) in $\frac{individuals}{km^2}$
- num_cpue , number of individuals (abundance) in $\frac{individuals}{h}$
- num , number of individuals (abundance)
- wgt_cpua , weight in $\frac{kg}{km^2}$
- wgt_cpue , weight in $\frac{kg}{h}$
- wgt , weight in kg



5. Extreme values

Here we show a yearly total distribution of the biomass data to visualize outliers:

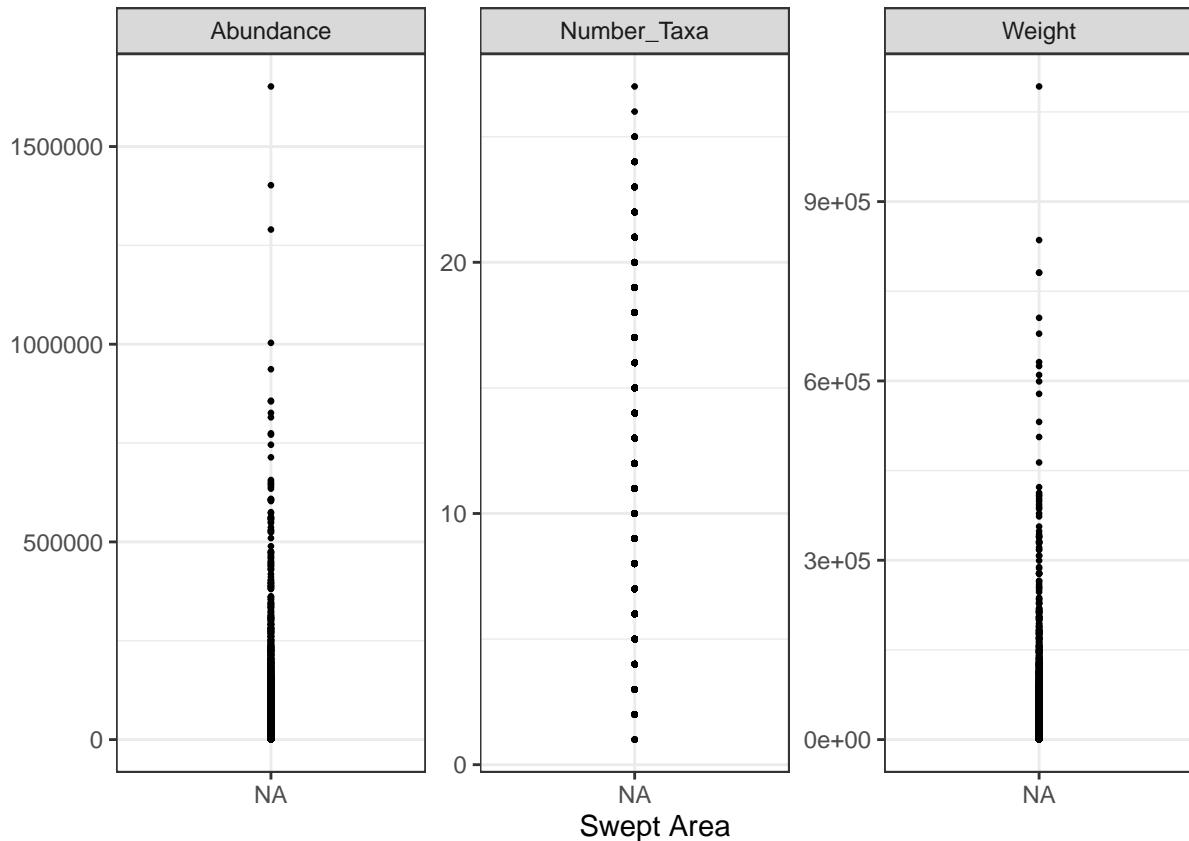
- num_cpue , number of individuals (abundance) in $\frac{individuals}{km^2}$
- wgt_cpue , weight in $\frac{kg}{km^2}$



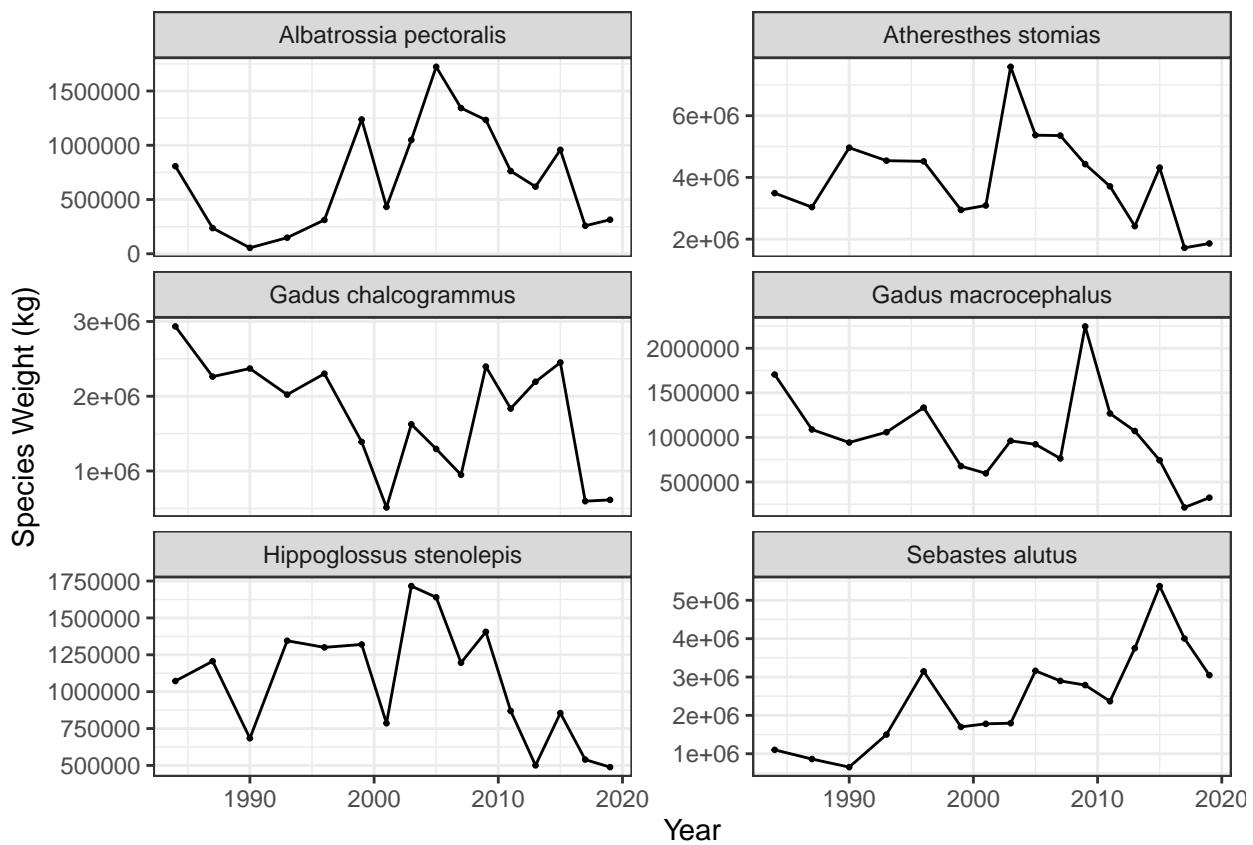
6. Summary of variables against swept area

Here we show the total abundance and number of taxa relationships with the area swept:

- nbr_taxa , number of marine fish taxa after taxonomic data cleaning
- num_cpua , number of individuals (abundance) in $\frac{individuals}{km^2}$
- wgt_cpua , weight in $\frac{kg}{km^2}$

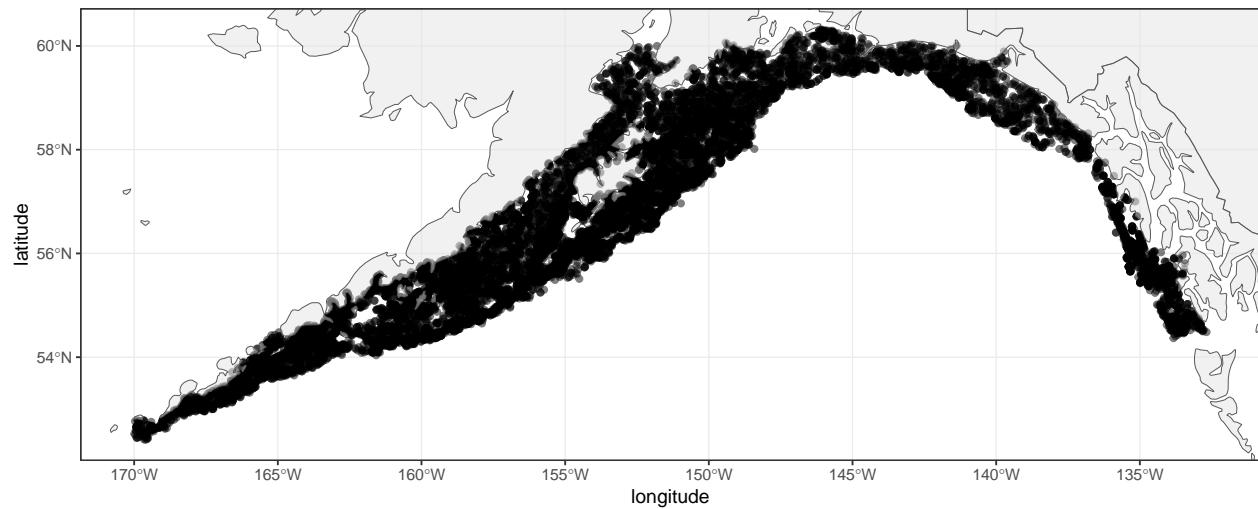


7. Abundance or Weight trends of the six most abundant species



8. Distribution mapping

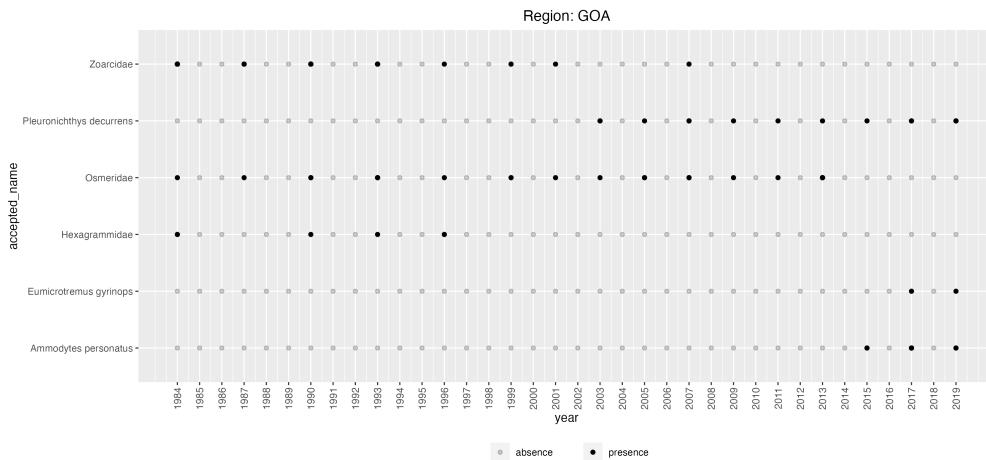
Map of the sampling distribution in space. Note that we only show one year per coordinate.



9. Taxonomic flagging

This species flagging method was adapted from <https://github.com/pinskylab/OceanAdapt/blob/master/R/add-spp-to-taxonomy.Rmd#L33>

Visualization of flagged taxa



Statistics related to the taxonomic flagging outputs

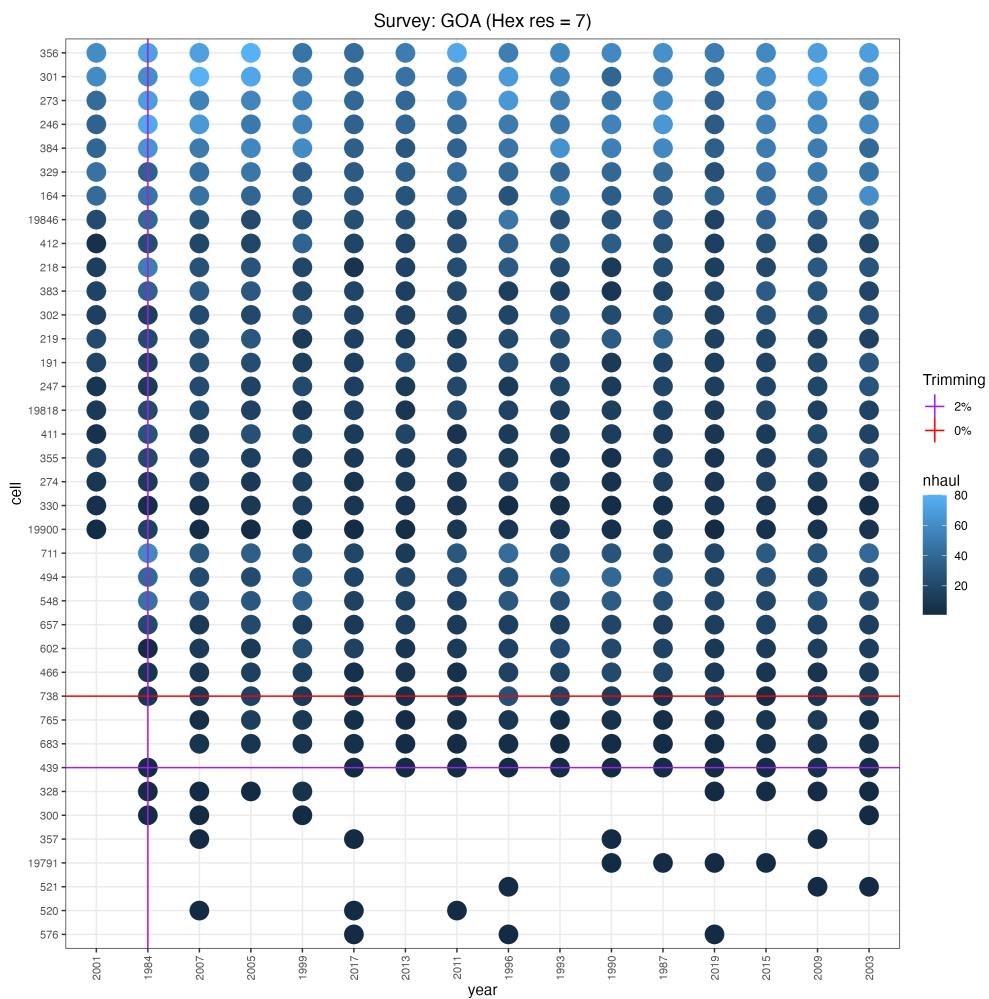
Total number of species	358.0
Percentage of species flagged	1.7

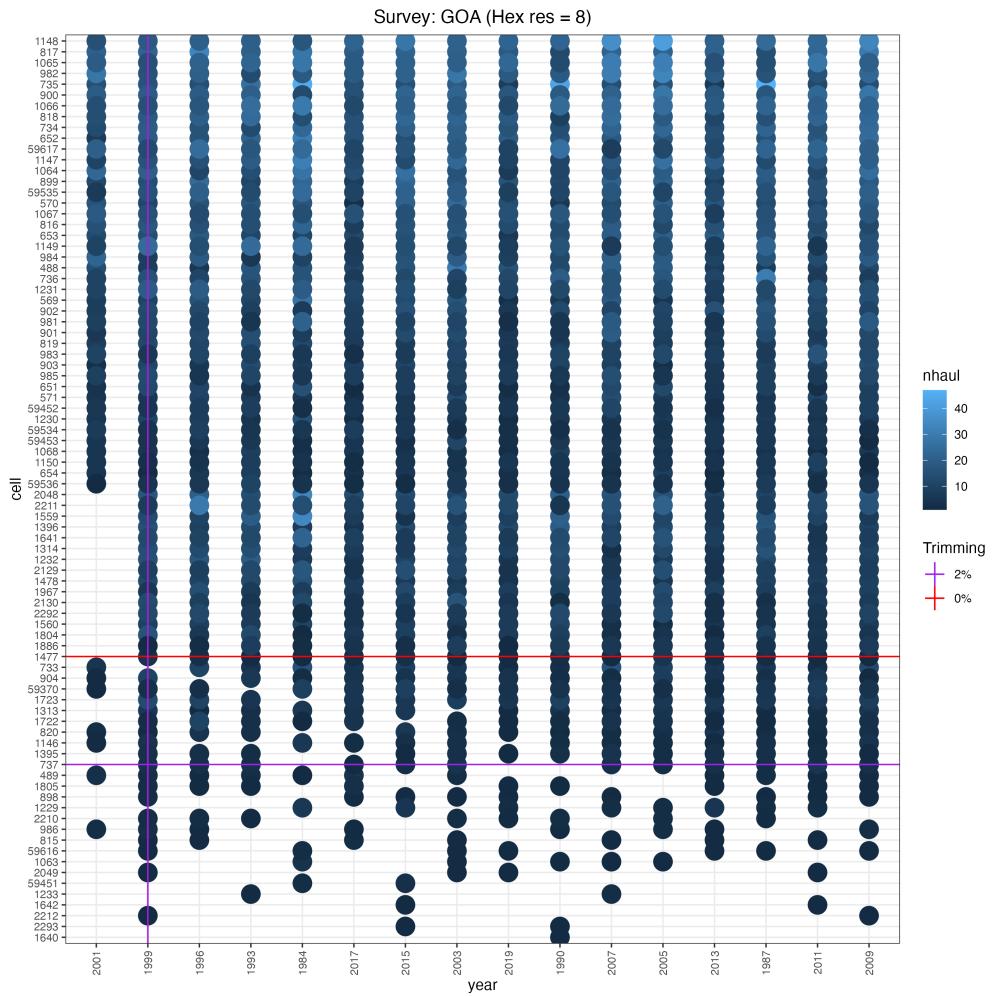
10. Spatio-temporal standardization

a. Standardization method 1

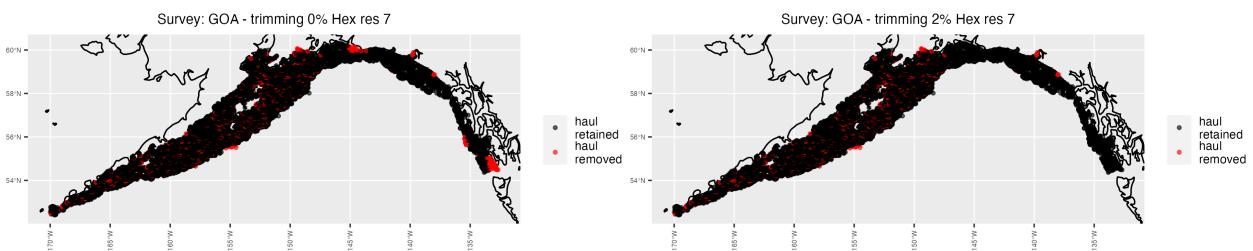
This standardization method was adapted from https://github.com/zookitchel/trawl_spatial_turnover/blob/master/data_prep_code/species/explore_NorthSea_trimming.Rmd
It was run for hex resolution 7 and 8.

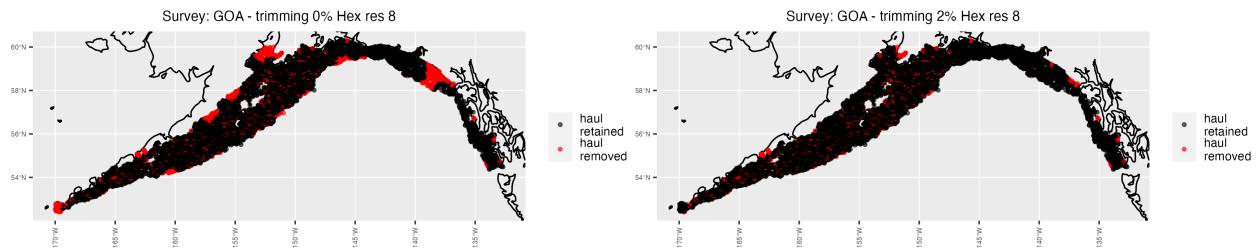
Plot of number of cells x years with overlaid flagging options



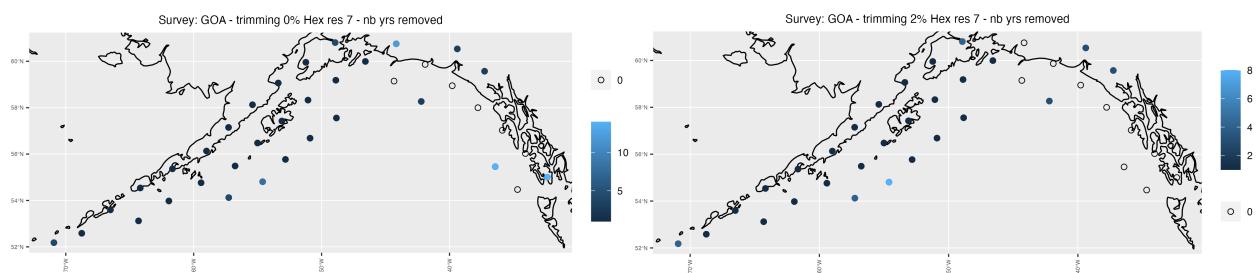


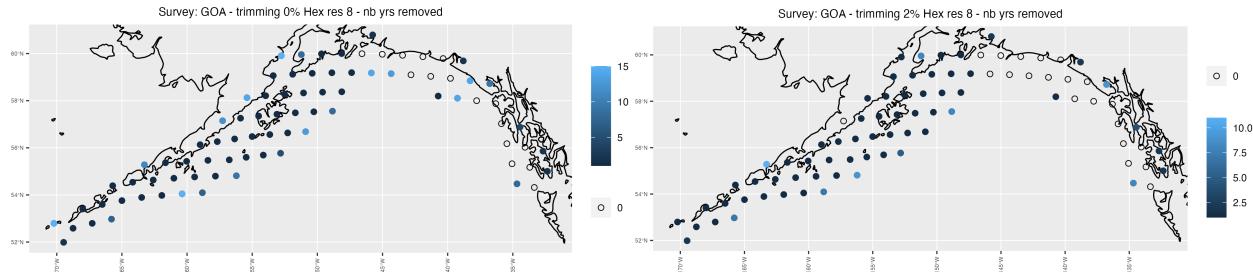
Map of hauls retained and removed per flagging method and threshold





Map of numbers of years removed per grid cell and flagging method/threshold

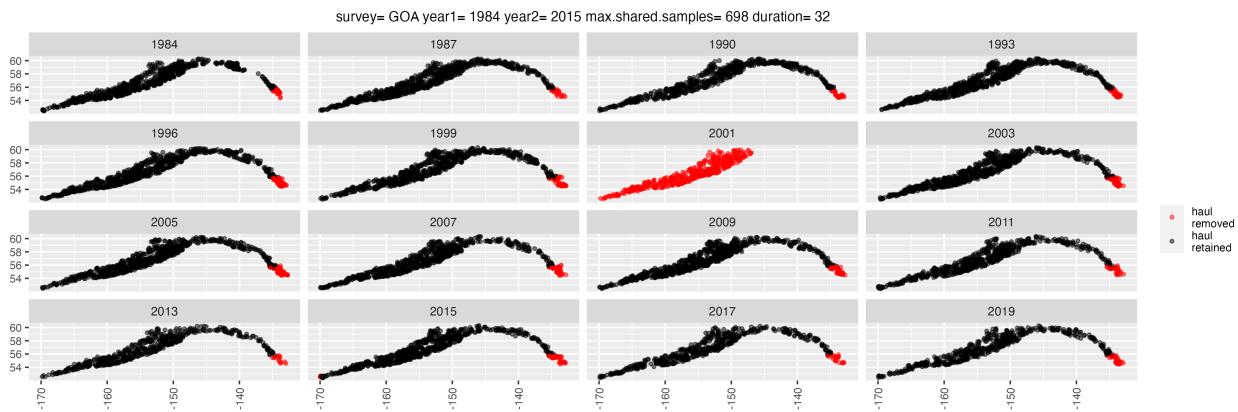




b. Standardization method 2

This standardization method was adapted from BioTIME code from https://github.com/Wubing-Xu/Range_size_winners_losers

Map of hauls retained and removed



c. Standardization summary

Statistics of hauls removed for each standardization method

summary	grid cell 7, 0% threshold	grid cell 7, 2% threshold	grid cell 8, 0% threshold	grid cell 8, 2% threshold	method 2 (biotime)
number of hauls removed	701	529.0	1178.0	621.0	12692.0
percentage of hauls removed	6	4.6	10.2	5.4	9.6