

# EBS: Eastern Bering Sea US survey data processing summary

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

November, 2023

## Contents

|  |    |
|--|----|
| General info . . . . .   | 1  |
| Data cleaning in R . . . . .   | 1  |
| 1. Overview of the survey data table . . . . .                           | 9  |
| 2. Summary of sampling intensity . . . . .                               | 10 |
| 3. Summary of sampling variables from the survey . . . . .               | 11 |
| 4. Summary of biological variables . . . . .                             | 12 |
| 5. Extreme values . . . . .  | 13 |
| 6. Summary of variables against swept area . . . . .                     | 14 |
| 7. Abundance or Weight trends of the six most abundant species . . . . . | 15 |
| 8. Distribution mapping . . . . .  | 16 |
| 9. Taxonomic flagging . . . . .  | 17 |
| 10. Spatio-temporal standardization . . . . .                            | 18 |
| a. Standardization method 1 . . . . .                                    | 18 |
| b. Standardization method 2 . . . . .                                    | 21 |
| c. Standardization summary . . . . .                                     | 21 |

## General info

This document presents the summary of the Eastern Bering Sea bottom trawl survey provided by Stan Kotwicki and Jim Thorson. It contains annual data from 1982-2019.

## Data cleaning in R

```
#####
#### R code to clean trawl survey Eastern Bering Sea
#### 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 November 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
#
#-----#
```

```

##### LOAD LIBRARIES AND FUNCTIONS #####
#-----#
library(tidyverse)
library(lubridate)
library(googledrive)
library(taxize) # for getting correct species names
library(magrittr) # for names wrangling
library(readxl)

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/ebs1982_1984.csv",
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/ebs1985_1989.csv",
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/ebs1990_1994.csv",
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/ebs1995_1999.csv",
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/ebs2000_2004.csv",
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/ebs2005_2008.csv",
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/ebs2009_2012.csv",
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/ebs2013_2016.csv",
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/ebs2017_2019.csv")

# combine all of the data files into one table
ebs_data <- files %>%
  # read in all of the csv's 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)) %>%
  # remove any extra white space from around spp and common names
  mutate(COMMON = str_trim(COMMON),
        SCIENTIFIC = str_trim(SCIENTIFIC))

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

```

```

ebs_strata <- read_csv(ebsstrat, col_types = cols(
  SubareaDescription = col_character(),
  StratumCode = col_integer(),
  Areakm2 = col_integer()
)) %>%
  rename(stratum = StratumCode)

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

ebs <- left_join(ebs_data, ebs_strata, by = "stratum")

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

ebs <- ebs %>%
  mutate(
    # Create a unique haul_id
    haul_id = paste(formatC(VESSEL, width=3, flag=0), CRUISE,
                    formatC(HAUL, width=3, flag=0), LONGITUDE, 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)
  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 = "EBS",
       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, 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)

#check to make sure all looks right
#str(ebs)

#-----#
#### 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
ebs_survey_code <- "EBS"

ebs_taxa <- ebs %>%
  select(verbatim_name) %>%
  mutate(
    taxa = str_squish(verbatim_name),
    taxa = str_remove_all(taxa, " spp.| sp.| spp| sp|NO "),
    taxa = str_to_sentence(str_to_lower(taxa))
) %>%
pull(taxa) %>%
unique()

```

```

# Get clean taxa
clean_auto <- clean_taxa(ebs_taxa, input_survey = ebs_survey_code, save = F,
                           output=NA, fishbase=T) # takes 4.1 mins!

#check those with no match from clean_taxa()
#Beringius beringii                                no match
#Crangon communis                                 no match
#Crangon abyssorum                               no match
#Cheiraster dawsoni                            no match

#####clear all invertebrates

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


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

clean_ebs <- left_join(ebs, clean_taxa, by=c("verbatim_name"="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,
        source = "NOAA",
        timestamp = my("03/2021"),
        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_ebs <- clean_ebs %>% count(haul_id, accepted_name)

#no duplicates

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


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

```

```

# -----#
##### 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",
                      "tidyR",
                      "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_ebs$survey))

#run flag_spp function in a loop
for (r in regions) {
  flag_spp(clean_ebs, 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_ebs, 7)

#apply trimming for hex size 8
dat_new_method1_hex8 <- apply_trimming_per_survey_unit_method1(clean_ebs, 8)

##### Apply trimming per survey_unit method 2
dat_new_method2 <- apply_trimming_per_survey_unit_method2(clean_ebs)

```

```

#-----#
##### ADD STANDARDIZATION FLAGS #####
#-----#
surveys <- sort(unique(clean_ebs$survey))
survey_units <- sort(unique(clean_ebs$survey_unit))
survey_std <- clean_ebs %>%
  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], "_flagsp.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= ";")
    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 = "EBS_std",
                 overwrite = T, rdata=TRUE)

```

## 1. Overview of the survey data table

| survey | source | timestamp  | haul_id                      | country       | sub_area |
|--------|--------|------------|------------------------------|---------------|----------|
| EBS    | NOAA   | 2021-03-01 | 037198401019-162.7143358.353 | United States | NA       |
| EBS    | NOAA   | 2021-03-01 | 037198401019-162.7143358.353 | United States | NA       |
| EBS    | NOAA   | 2021-03-01 | 037198401019-162.7143358.353 | United States | NA       |
| EBS    | NOAA   | 2021-03-01 | 037198401019-162.7143358.353 | United States | NA       |
| EBS    | NOAA   | 2021-03-01 | 037198401019-162.7143358.353 | United States | NA       |

| continent | stat_rec | station | stratum | year | month | day | quarter | season |
|-----------|----------|---------|---------|------|-------|-----|---------|--------|
| n_america | NA       | K-09    | 10      | 1984 | 6     | 14  | 2       | NA     |
| n_america | NA       | K-09    | 10      | 1984 | 6     | 14  | 2       | NA     |
| n_america | NA       | K-09    | 10      | 1984 | 6     | 14  | 2       | NA     |
| n_america | NA       | K-09    | 10      | 1984 | 6     | 14  | 2       | NA     |
| n_america | NA       | K-09    | 10      | 1984 | 6     | 14  | 2       | NA     |

| latitude | longitude | haul_dur | area_swept | gear | depth | sbt | sst |
|----------|-----------|----------|------------|------|-------|-----|-----|
| 58.353   | -162.7143 | NA       | NA         | NA   | 31    | 4.2 | 3.6 |
| 58.353   | -162.7143 | NA       | NA         | NA   | 31    | 4.2 | 3.6 |
| 58.353   | -162.7143 | NA       | NA         | NA   | 31    | 4.2 | 3.6 |
| 58.353   | -162.7143 | NA       | NA         | NA   | 31    | 4.2 | 3.6 |
| 58.353   | -162.7143 | NA       | NA         | NA   | 31    | 4.2 | 3.6 |

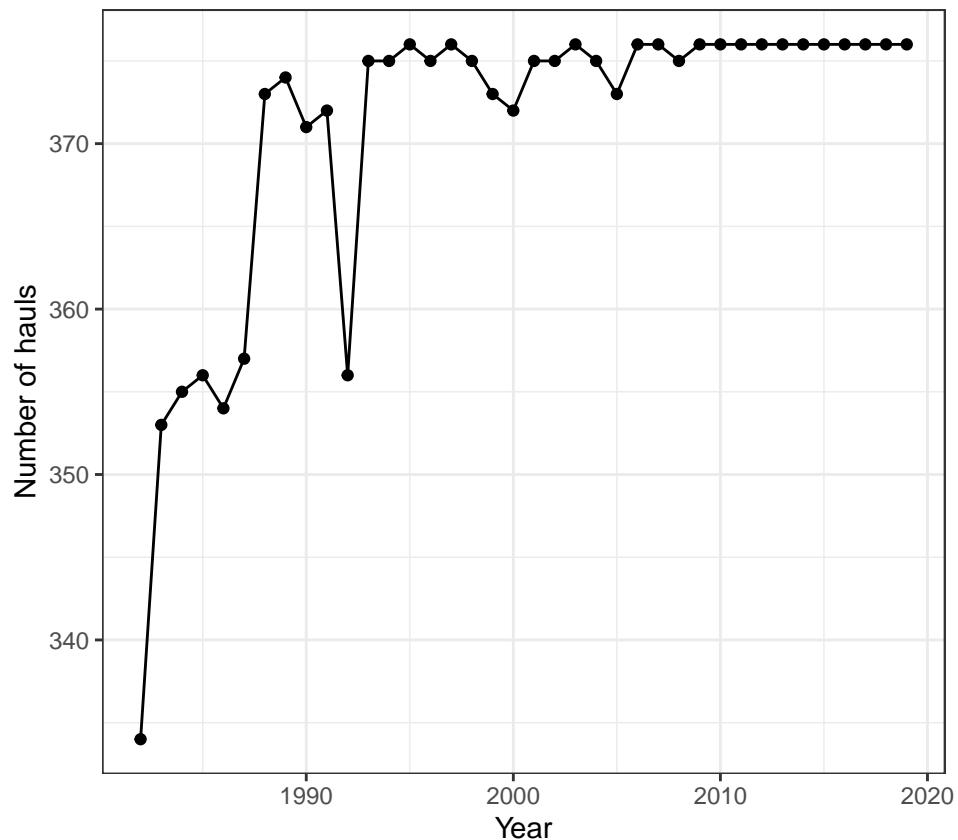
| num | num_cpue | num_ccpuia | wgt | wgt_cpue | wgt_ccpuia | verbatim_name             |
|-----|----------|------------|-----|----------|------------|---------------------------|
| NA  | NA       | 30.73      | NA  | NA       | 139.38     | Rajidae                   |
| NA  | NA       | 399.46     | NA  | NA       | 11.15      | Mallotus villosus         |
| NA  | NA       | 61.46      | NA  | NA       | 111.51     | Gadus chalcogrammus       |
| NA  | NA       | 1136.92    | NA  | NA       | 3094.20    | Gadus macrocephalus       |
| NA  | NA       | 61.46      | NA  | NA       | 5.56       | Podothecus accipenserinus |

| verbatim_aphia_id | accepted_name             | aphia_id | SpecCode | kingdom  |
|-------------------|---------------------------|----------|----------|----------|
| NA                | Rajidae                   | 105711   | NA       | Animalia |
| NA                | Mallotus villosus         | 126735   | 252      | Animalia |
| NA                | Gadus chalcogrammus       | 300735   | 318      | Animalia |
| NA                | Gadus macrocephalus       | 254538   | 308      | Animalia |
| NA                | Podothecus accipenserinus | 254501   | 4153     | Animalia |

| phylum   | class          | order        | family    | genus      | rank    | survey_unit |
|----------|----------------|--------------|-----------|------------|---------|-------------|
| Chordata | Elasmobranchii | Rajiformes   | Rajidae   | NA         | Family  | EBS         |
| Chordata | Teleostei      | Osmeriformes | Osmeridae | Mallotus   | Species | EBS         |
| Chordata | Teleostei      | Gadiformes   | Gadidae   | Gadus      | Species | EBS         |
| Chordata | Teleostei      | Gadiformes   | Gadidae   | Gadus      | Species | EBS         |
| Chordata | Teleostei      | Perciformes  | Agonidae  | Podothecus | Species | EBS         |

## 2. Summary of sampling intensity

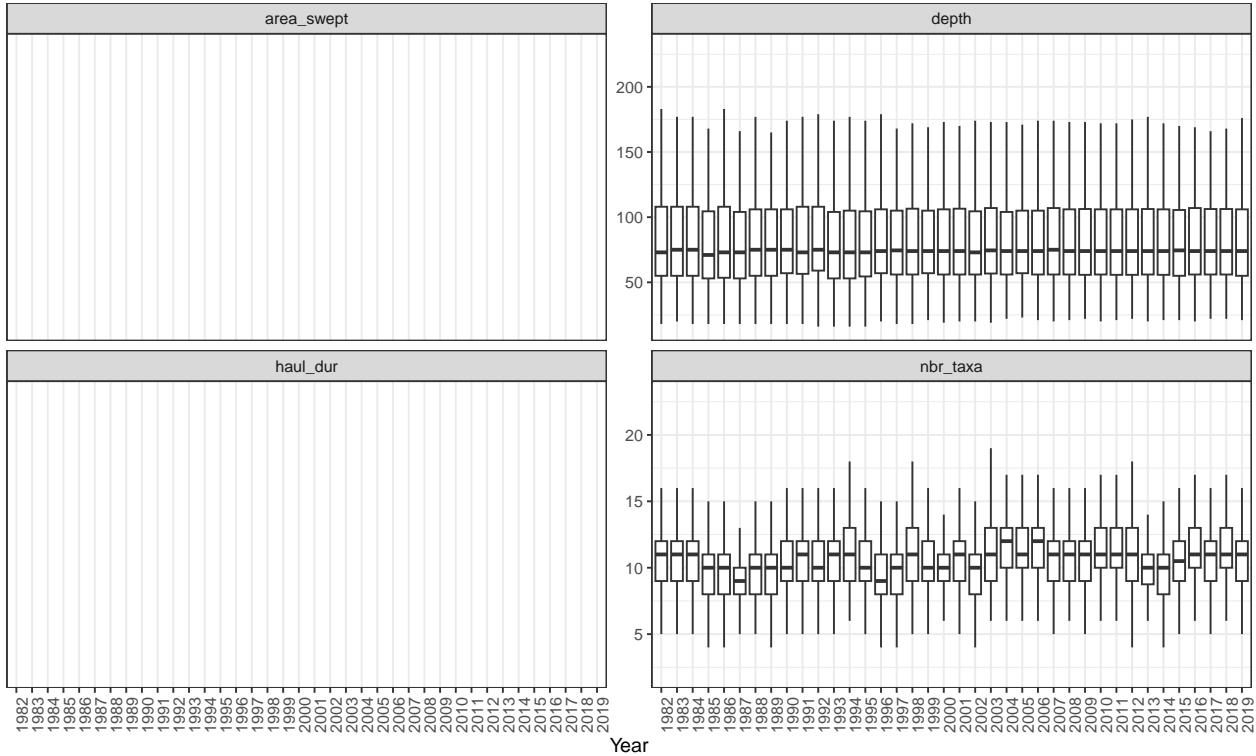
Number of hauls per year performed during the survey after data processing.



### 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:

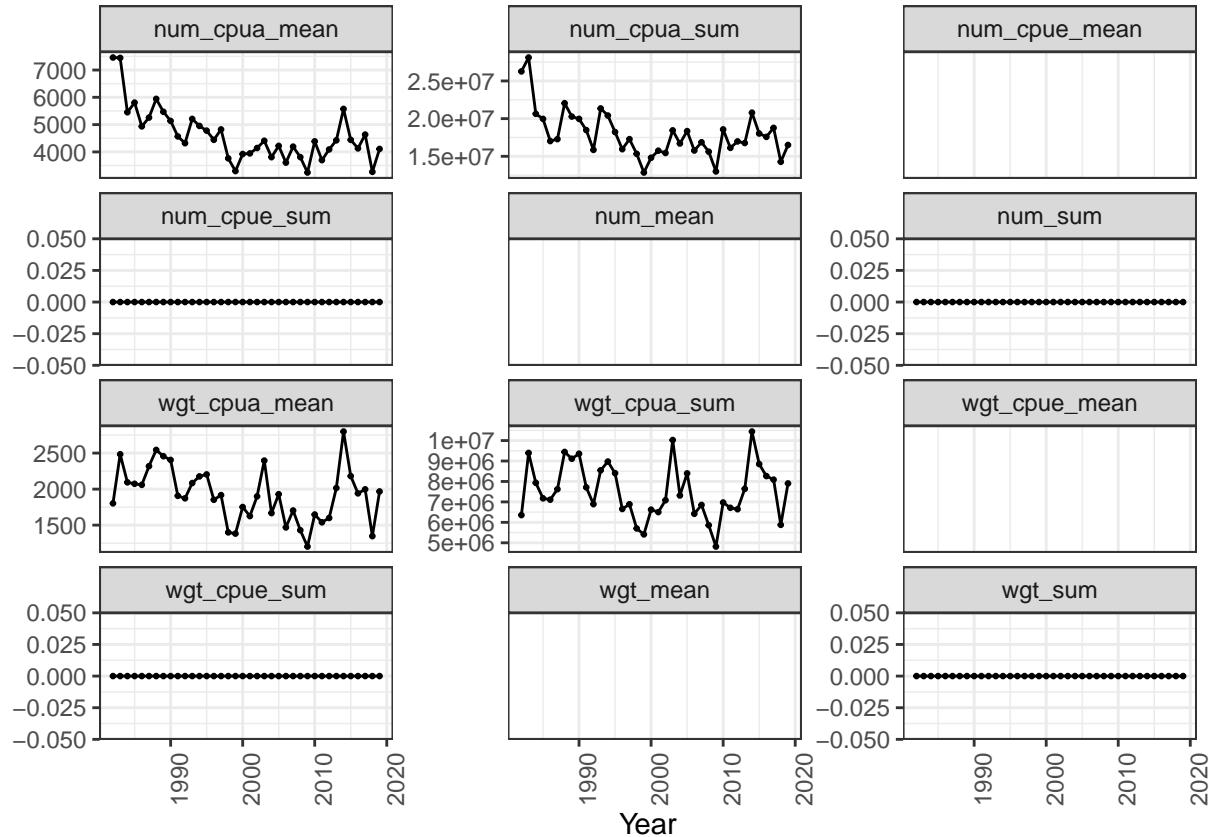
- *area\_swept*, swept area by the bottom trawl gear  $km^2$
- *depth*, sampling depth in  $m$
- *haul\_dur*, haul sampling duration *hour*
- *number of marine fish taxa*, taxa were cleaned following the last version of taxonomy from the World Register of Marine Species (<https://www.marinespecies.org/>,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

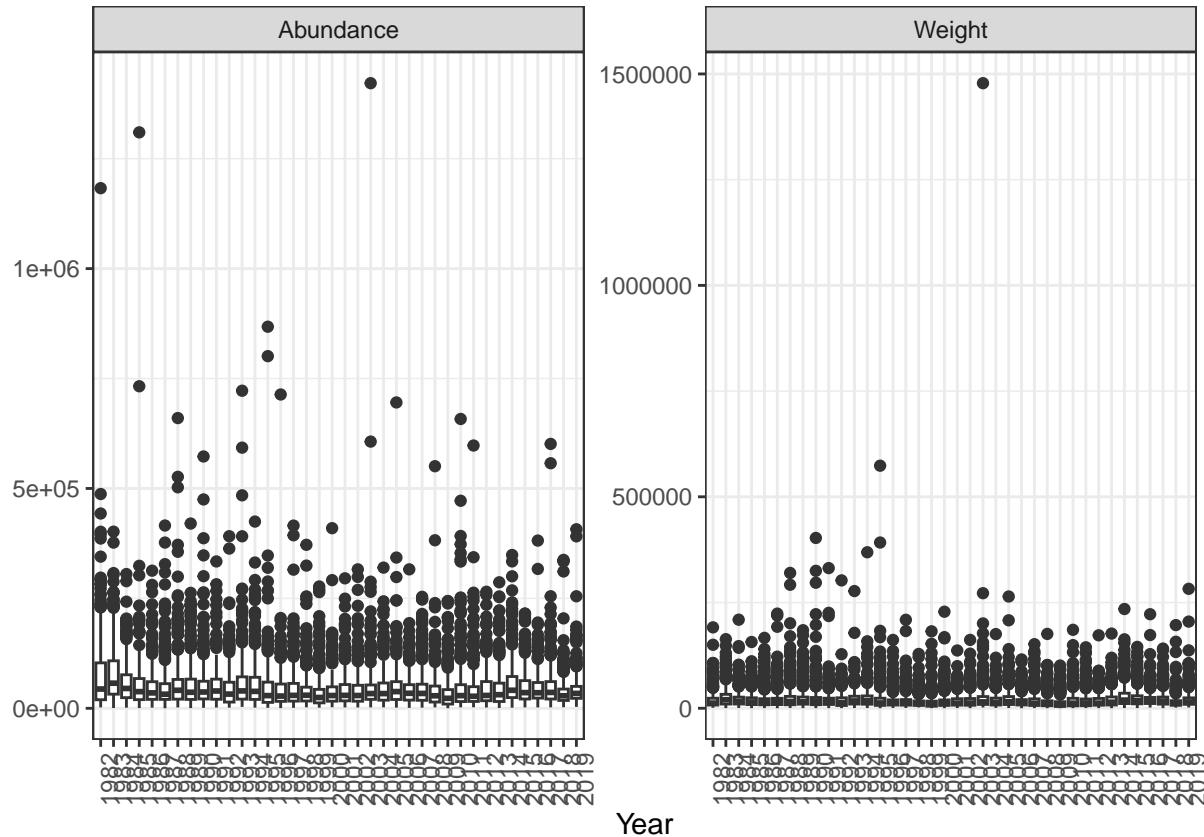
- $num\_cpua$ , number of individuals (abundance) in  $\frac{individuals}{km^2}$
- $num\_cpue$ , number of individuals (abundance) in  $\frac{individuals}{h}$
- $num$ , number of individuals (abundance)
- $wgt\_cpua$ , weight in  $\frac{kg}{km^2}$
- $wgt\_cpue$ , weight in  $\frac{kg}{h}$
- $wgt$ , weight in  $kg$



## 5. Extreme values

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

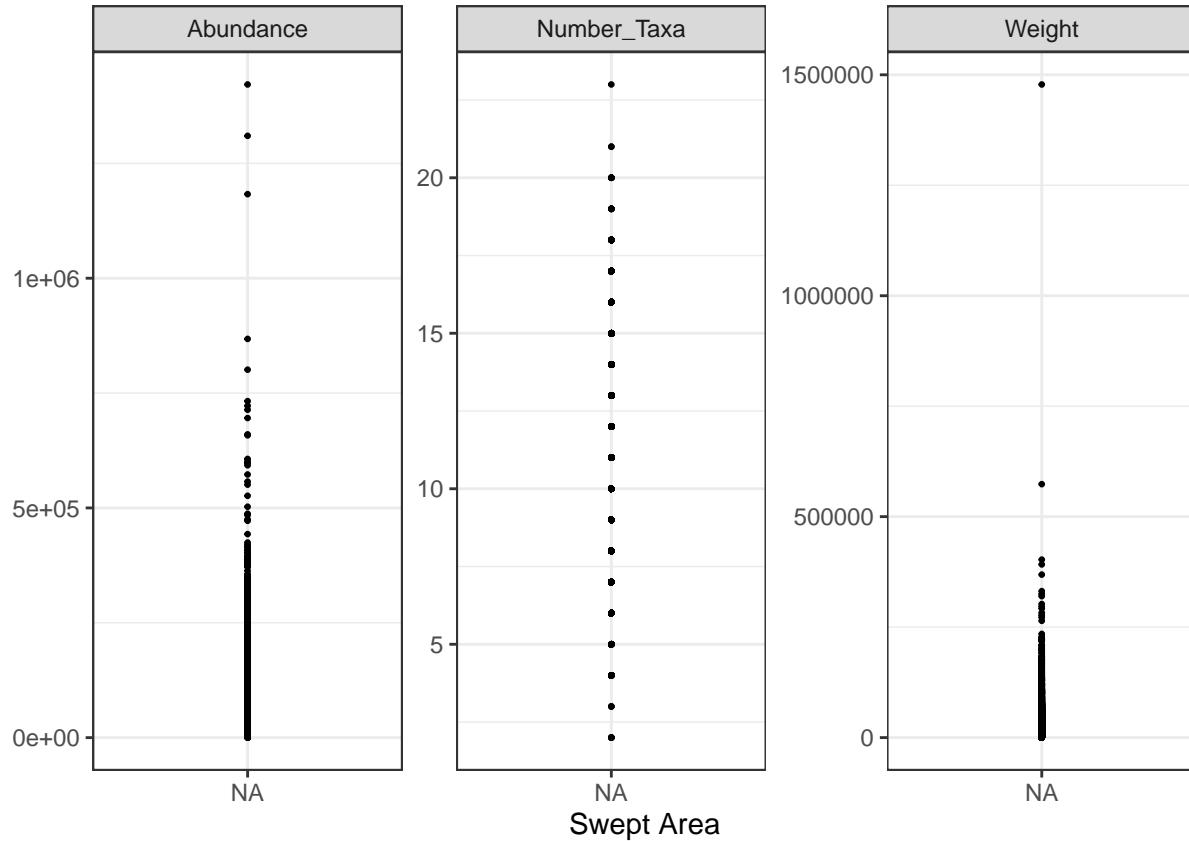
- $num\_cpue$ , number of individuals (abundance) in  $\frac{individuals}{km^2}$
- $wgt\_cpue$ , weight in  $\frac{kg}{km^2}$



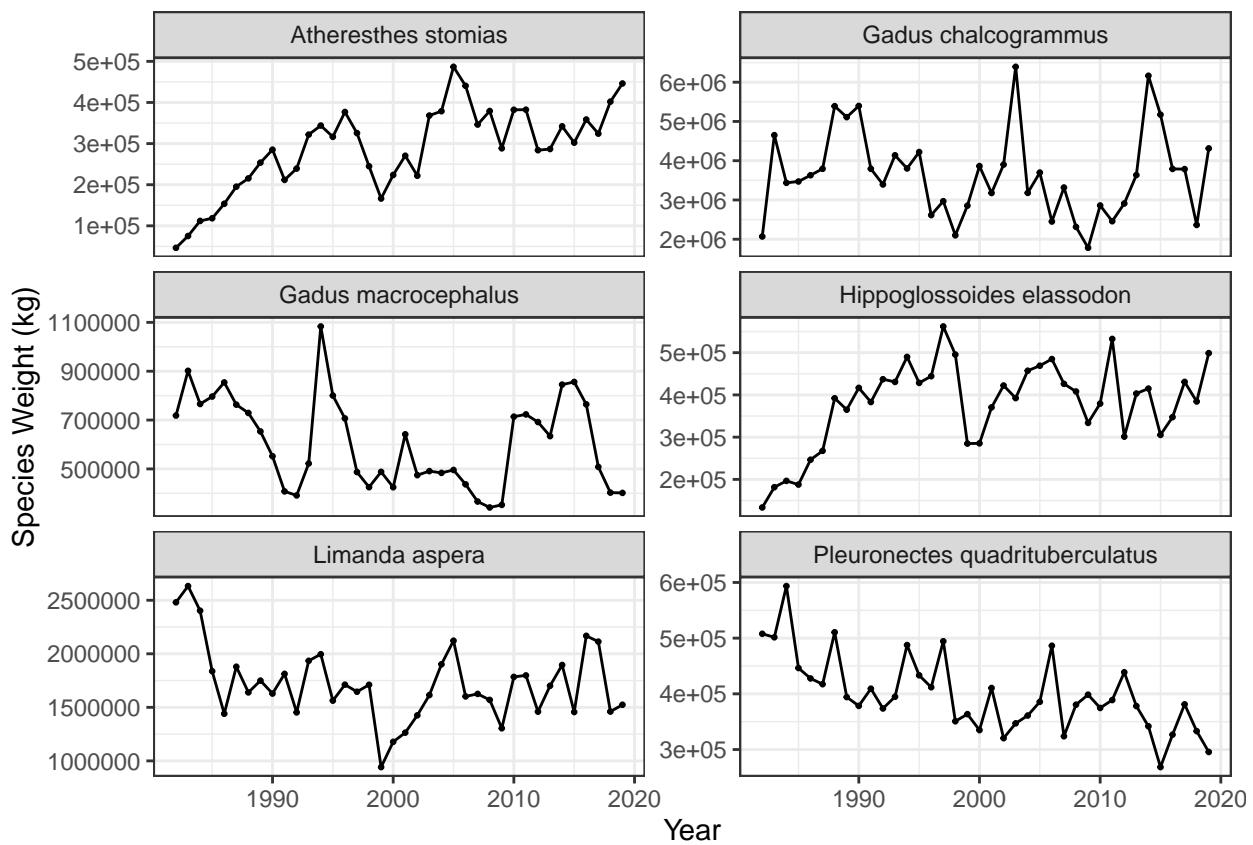
## 6. Summary of variables against swept area

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  $\frac{kg}{km^2}$

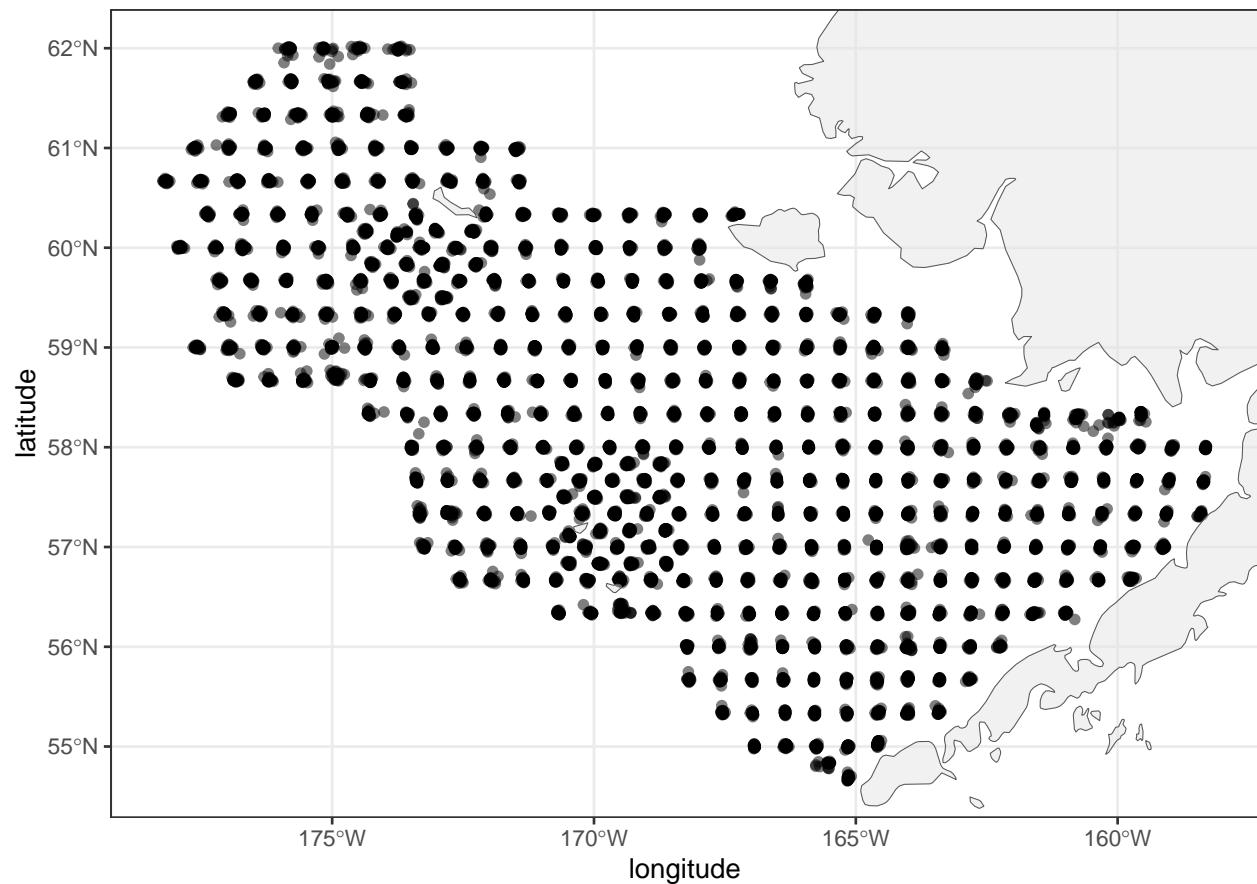


## 7. Abundance or Weight trends of the six most abundant species



## 8. Distribution mapping

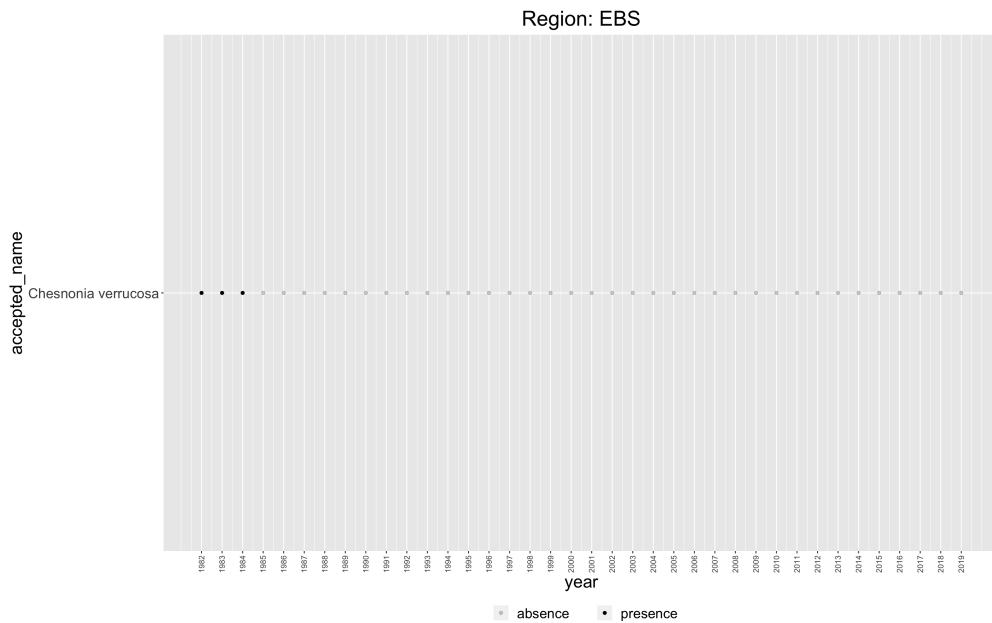
Map of the sampling distribution in space. Note that we only show one year per coordinate.



## 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

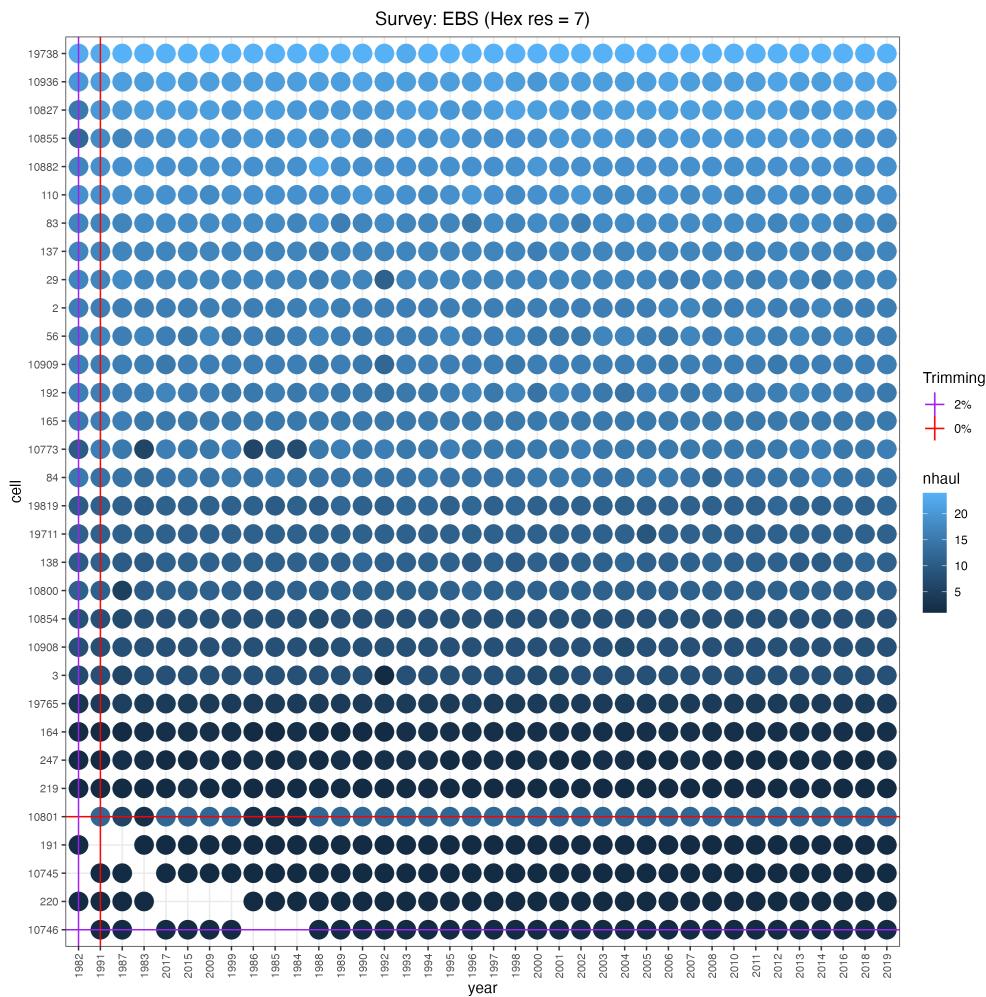
|                               |       |
|-------------------------------|-------|
| Total number of species       | 170.0 |
| Percentage of species flagged | 0.6   |

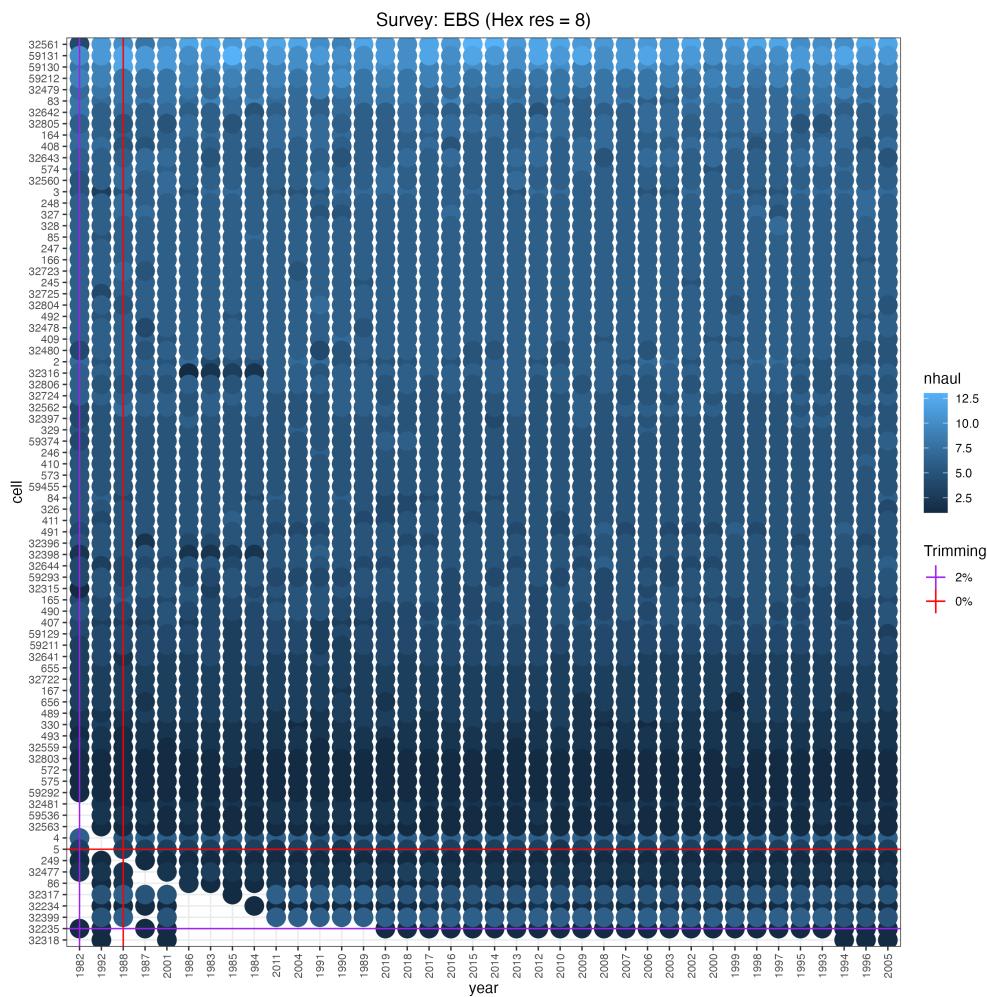
## 10. Spatio-temporal standardization

### a. Standardization method 1

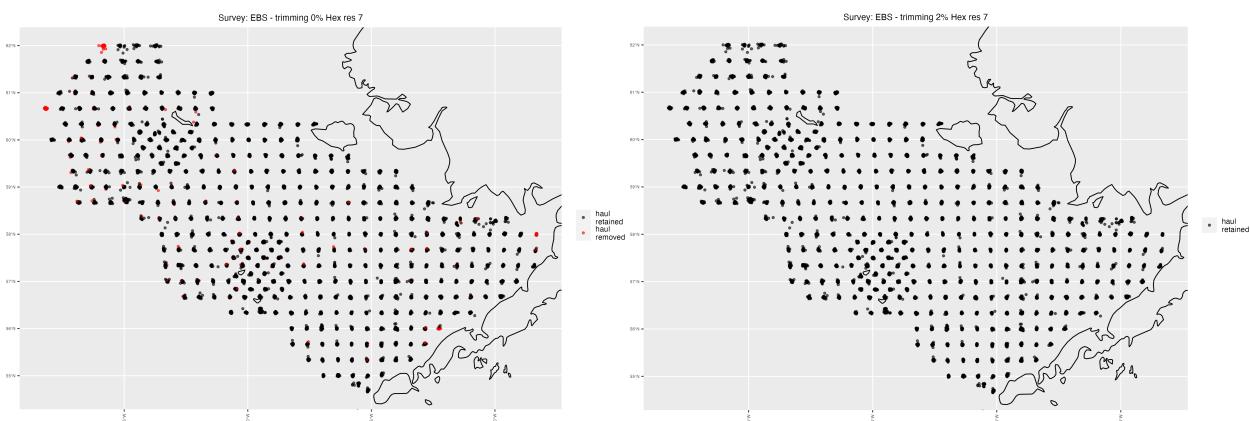
This standardization method was adapted from [https://github.com/zoeKitchen/trawl\\_spatial\\_turnover/blob/master/data\\_prep\\_code/species/explore\\_NorthSea\\_trimming.Rmd](https://github.com/zoeKitchen/trawl_spatial_turnover/blob/master/data_prep_code/species/explore_NorthSea_trimming.Rmd)  
It was run for hex resolution 7 and 8.

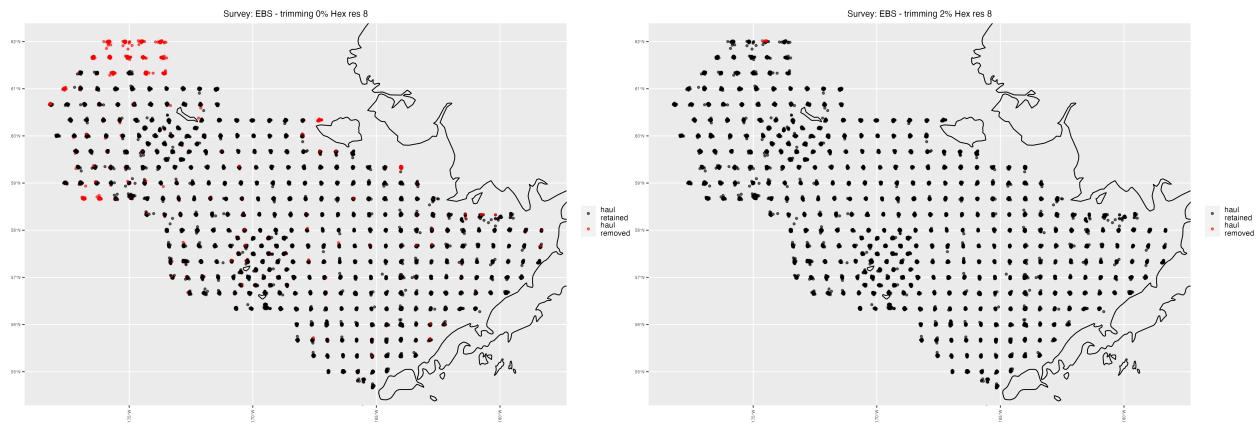
Plot of number of cells x years with overlaid flagging options



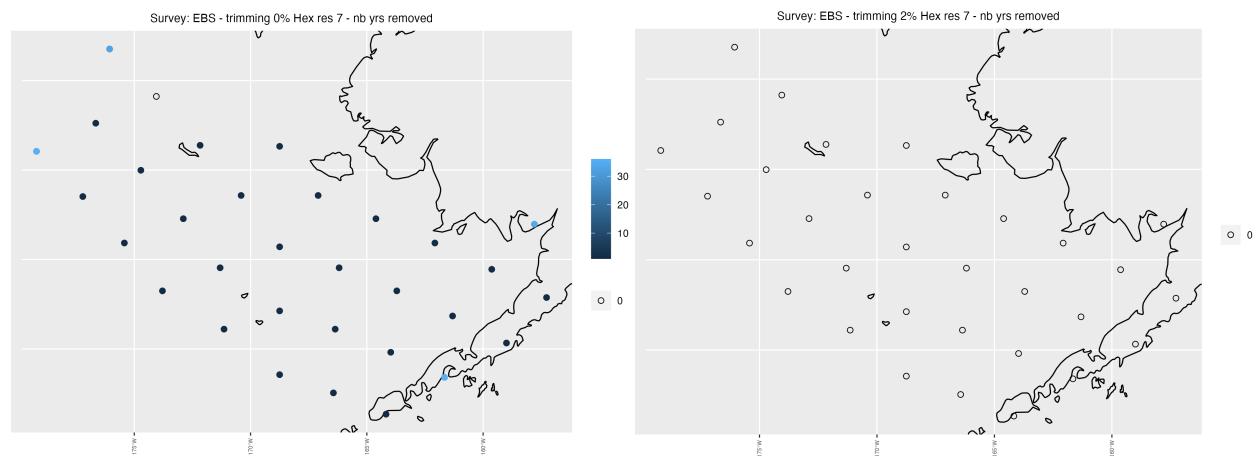


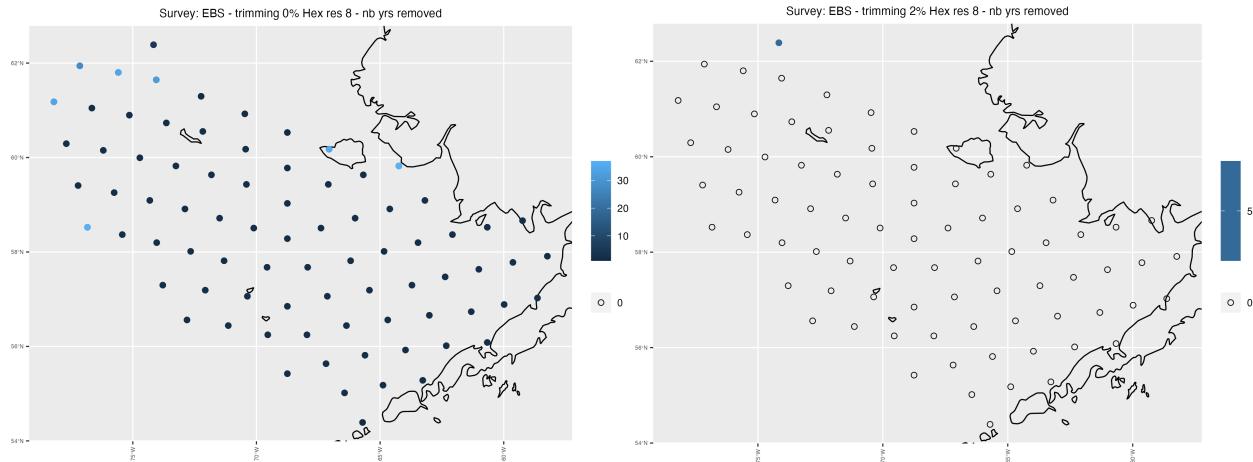
Map of hauls retained and removed per flagging method and threshold





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

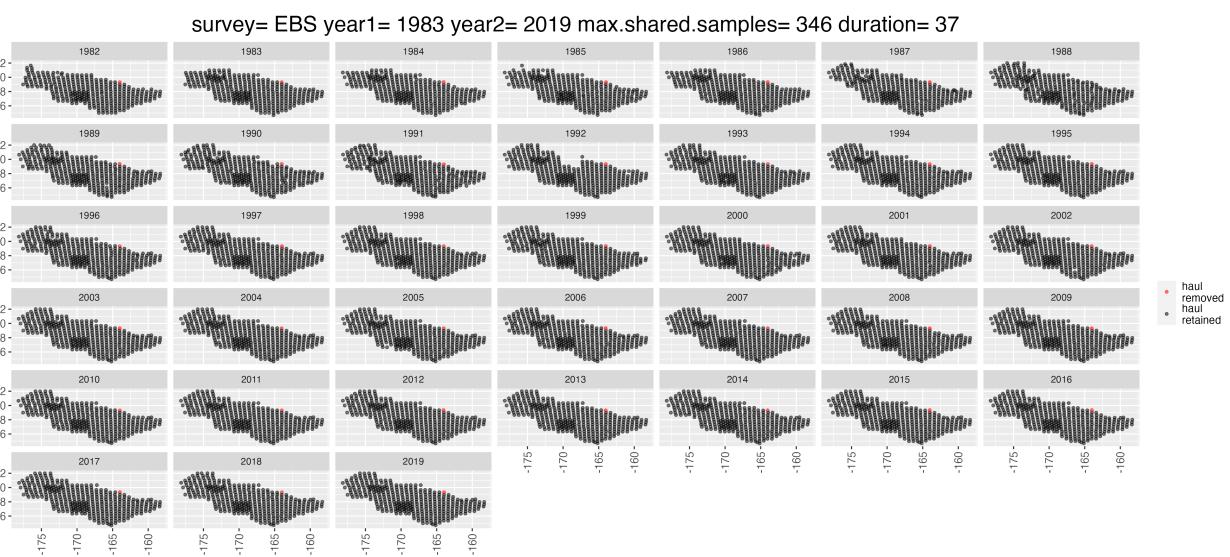




### b. Standardization method 2

This standardization method was adapted from BioTIME code from [https://github.com/Wubing-Xu/Range\\_size\\_winners\\_losers](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     | 471.0                     | 0                         | 1260.0                    | 5                         | 429.0              |
| percentage of hauls removed | 3.3                       | 0                         | 8.9                       | 0                         | 0.3                |