SEUS: Southeast US survey data processing summary

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

November, 2023

Contents

General info

This document presents the cleaning code and summary of the Southeast US bottom trawl survey provided by Sarah Murray Fisheries Science Coordinator, Atlantic States Marine Fisheries Commission and Tracey Smart Associate Marine Scientist, Data Manager, Department of Natural Resources, South Carolina. It contains annual data from 1989 and up to 2019.

Data cleaning in R

```
################################################################################
#### R code to clean trawl survey Southeast US
#### Public data Ocean Adapt
#### Contacts:  Sarah Murray    smurray@asmfc.org   Fisheries Science Coordinator
####              Atlantic States Marine Fisheries Commission
####            Tracey Smart    smartt@dnr.sc.gov   Associate Marine Scientist
####              Data Manager, Department of Natural Resources, South Carolina
```

```
#### Coding: Michelle Stuart, Dan Forrest, Zoë Kitchel November 2021
################################################################################

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


library(rfishbase) #needs R 4.0 or more recent
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 Southeast US 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 ####
#--------------------------------------------------------------------------------------#


# turns everything into a character so import as character anyway
temp <- tempfile()
download.file(
  "https://github.com/pinskylab/OceanAdapt/raw/master/data_raw/seus_catch.csv.zip",
  temp)

seus_catch <- read_csv(unz(temp, "seus_catch.csv"),
                       col_types = cols(.default = col_character())) %>%
  # remove symbols
  mutate_all(list(~str_replace(., "=", ""))) %>%
  mutate_all(list(~str_replace(., '"', ''))) %>%
  mutate_all(list(~str_replace(., '\"', '')))

# The 9 parsing failures are due to the metadata at the end of the file that does not
#fit into the data columns

# problems should have 0 obs
problems <- problems(seus_catch) %>%
  filter(!is.na(col))
stopifnot(nrow(problems) == 0)
```

```r
# convert the columns to their correct formats
seus_catch <- type_convert(seus_catch, col_types = cols(
  PROJECTNAME = col_character(),
  PROJECTAGENCY = col_character(),
  DATE = col_character(),
  EVENTNAME = col_character(),
  COLLECTIONNUMBER = col_character(),
  VESSELNAME = col_character(),
  GEARNAME = col_character(),
  GEARCODE = col_character(),
  SPECIESCODE = col_character(),
  MRRI_CODE = col_character(),
  SPECIESSCIENTIFICNAME = col_character(),
  SPECIESCOMMONNAME = col_character(),
  NUMBERTOTAL = col_integer(),
  SPECIESTOTALWEIGHT = col_double(),
  SPECIESSUBWEIGHT = col_double(),
  SPECIESWGTPROCESSED = col_character(),
  WEIGHTMETHODDESC = col_character(),
  ORGWTUNITS = col_character(),
  EFFORT = col_character(),
  CATCHSUBSAMPLED = col_logical(),
  CATCHWEIGHT = col_double(),
  CATCHSUBWEIGHT = col_double(),
  TIMESTART = col_character(),
  DURATION = col_integer(),
  TOWTYPETEXT = col_character(),
  LOCATION = col_character(),
  REGION = col_character(),
  DEPTHZONE = col_character(),
  ACCSPGRIDCODE = col_character(),
  STATIONCODE = col_character(),
  EVENTTYPEDESCRIPTION = col_character(),
  TEMPSURFACE = col_double(),
  TEMPBOTTOM = col_double(),
  SALINITYSURFACE = col_double(),
  SALINITYBOTTOM = col_double(),
  SDO = col_character(),
  BDO = col_character(),
  TEMPAIR = col_double(),
  LATITUDESTART = col_double(),
  LATITUDEEND = col_double(),
  LONGITUDESTART = col_double(),
  LONGITUDEEND = col_double(),
  SPECSTATUSDESCRIPTION = col_character(),
  LASTUPDATED = col_character()
))

seus_haul <- read_csv(
  "https://github.com/pinskylab/OceanAdapt/raw/master/data_raw/seus_haul.csv",
                      col_types = cols(.default = col_character())) %>%
  distinct(EVENTNAME, DEPTHSTART)  %>%
  # remove symbols
  mutate_all(list(~str_replace(., "=", ""))) %>%
  mutate_all(list(~str_replace(., '"', ''))) %>%
  mutate_all(list(~str_replace(., '"', '')))

# problems should have 0 obs
problems <- problems(seus_haul) %>%
  filter(!is.na(col))
stopifnot(nrow(problems) == 0)

seus_haul <- type_convert(seus_haul, col_types = cols(
  EVENTNAME = col_character(),
  DEPTHSTART = col_integer()
))

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

seus <- left_join(seus_catch, seus_haul, by = "EVENTNAME")

# contains strata areas
seus_strata <- read_csv(
"https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/seus_strata.csv",
  col_types = cols(
  STRATA = col_integer(),
  STRATAHECTARE = col_double()
))


#Create STRATA column
seus <- seus %>%
  mutate(STRATA = as.numeric(str_sub(STATIONCODE, 1, 2))) %>%
  # Drop OUTER depth zone because it was only sampled for 10 years
  filter(DEPTHZONE != "OUTER")

#add STRATAHECTARE to main file
seus <- left_join(seus, seus_strata, by = "STRATA")

#Create a 'SEASON' column using 'MONTH' as a criteria
seus <- seus %>%
  mutate(DATE = as.Date(DATE, "%m-%d-%Y"),
         MONTH = month(DATE),
         year = year(DATE),
         DAY = day(DATE)) %>%
  # create season column
  mutate(SEASON = NA,
         SEASON = ifelse(MONTH >= 1 & MONTH <= 3, "winter", SEASON),
         SEASON = ifelse(MONTH >= 4 & MONTH <= 6, "spring", SEASON),
         SEASON = ifelse(MONTH >= 7 & MONTH <= 8, "summer", SEASON),
         #September EVENTS were grouped with summer, should be fall because all
         #hauls made in late-September during fall-survey
         SEASON = ifelse(MONTH >= 9 & MONTH <= 12, "fall", SEASON))
```

```
#Data entry error fixes for lat/lon coordinates
seus <- seus %>%
  mutate(
# longitudes of less than -360 (like -700), do not exist.
    #This is a missing decimal.
    LONGITUDESTART = ifelse(LONGITUDESTART < -360, LONGITUDESTART/10, LONGITUDESTART),
    LONGITUDEEND = ifelse(LONGITUDEEND < -360, LONGITUDEEND/10, LONGITUDEEND),
# latitudes of more than 100 are outside the range of this survey.
    #This is a missing decimal.
    LATITUDESTART = ifelse(LATITUDESTART > 100, LATITUDESTART/10, LATITUDESTART),
    LATITUDEEND = ifelse(LATITUDEEND  > 100, LATITUDEEND/10, LATITUDEEND)
  )

# calculate trawl distance in order to calculate effort
# create a matrix of starting positions
start <- as.matrix(seus[,c("LONGITUDESTART", "LATITUDESTART")], nrow = nrow(seus),
                   ncol = 2)
# create a matrix of ending positions
end <- as.matrix(seus[,c("LONGITUDEEND", "LATITUDEEND")], nrow = nrow(seus), ncol = 2)
# add distance to seus table (note that this distance is covered twice
#because there are parallel
#trawls occurring)
seus <- seus %>%
  mutate(distance_m = geosphere::distHaversine(p1 = start, p2 = end),
         distance_km = distance_m / 1000.0,
         ) %>%
  # calculate effort = mean area swept
  # EFFORT = 0 where the boat didn't move, distance_m = 0
     #mean area swept in km^2 =
  #width of net (13.5m)*1m/1000km * distance boat moved (km) =
  #                                       km^2 area effort for one of two nets
  mutate(EFFORT = 13.5/1000 * distance_km,
         # Create a unique haul_id
         haul_id = EVENTNAME,
         haul_dur = DURATION/60 #convert haul duration from minutes to hours
  ) %>%
  rename(
    stratum = STRATA,
    lat = LATITUDESTART,
    lon = LONGITUDESTART,
    depth = DEPTHSTART,
    spp = SPECIESSCIENTIFICNAME,
    stratumarea = STRATAHECTARE) %>%
select("haul_id", "year", "lat", "lon", "stratum", "stratumarea",
"depth", "spp",  "SEASON", "STATIONCODE",
       "MONTH", "DAY", "EFFORT",
       "TEMPSURFACE",
       "TEMPBOTTOM", "haul_dur", "GEARNAME", "SPECIESTOTALWEIGHT",
"NUMBERTOTAL")

#In seus there are two 'COLLECTIONNUMBERS' per 'EVENTNAME', with no exceptions,
#for each side of the boat;
#EFFORT is always the same for each COLLECTIONNUMBER
```

```
# We sum the two tows in seus (port and starboard tows)
#this steps deletes any haul id x spp duplicates
seus <- seus %>%
  group_by(haul_id, year, lat, lon, stratum, stratumarea,
            depth, spp,  SEASON, STATIONCODE,
            MONTH, DAY,
            TEMPSURFACE,
            TEMPBOTTOM, haul_dur, GEARNAME, EFFORT) %>%
  # adjust spp names (we want to sum over these genuses)
  mutate(
    spp = ifelse(grepl("ANCHOA", spp), "ANCHOA", spp), #any observation of anchoa is only
    #resolved to genus
    spp = ifelse(grepl("LIBINIA", spp), "LIBINIA", spp)) %>%
  #any observation of Libinia is only
    #resolved to genus

  #now this accounts for both sides of the boat, and merging within specified genuses
  summarise(biomass = sum(SPECIESTOTALWEIGHT,na.rm = T),
            abundance = sum(NUMBERTOTAL,na.rm = T)) %>%
  mutate(wgt_cpue = biomass/(EFFORT*2), num_cpue = abundance/(EFFORT*2),
         num_h = abundance/haul_dur,
         wgt_h = biomass/haul_dur)


seus <- seus %>%
  # remove non-fish
  filter(
    !spp %in% c('MISCELLANEOUS INVERTEBRATES','XANTHIDAE','MICROPANOPE NUTTINGI',
                'ALGAE','DYSPANOPEUS SAYI',
                'PSEUDOMEDAEUS AGASSIZII')
  )  %>%
  mutate(survey = "SEUS") %>%
  select(survey, haul_id, year, lat, lon, stratum, stratumarea, depth, spp, wgt_cpue,
         wgt_h, num_cpue, num_h,abundance, SEASON, STATIONCODE, MONTH, DAY, EFFORT,
         TEMPSURFACE,
         TEMPBOTTOM, biomass, haul_dur, GEARNAME) %>%
  ungroup()

#remove infinite wtcpue values (where effort was 0, causes wtcpue to be inf)
seus <- seus[!is.infinite(seus$wgt_cpue),]

seus <- seus %>%
  mutate(
    # Create a unique haul_id
    haul_id = paste(haul_id, stratum, lon, lat, sep=''),#previous haul id just event name,
    #this adds extra info
    wgt_cpue = ifelse(wgt_cpue == "-9999", NA, wgt_cpue),
    wgt_h = ifelse(wgt_h == "-9999", NA, wgt_h)
    ) %>%
  rename(year = year,
         day = DAY,
         month = MONTH,
         latitude = lat,
         longitude = lon,
         station = STATIONCODE,
         sbt = TEMPBOTTOM,
         sst = TEMPSURFACE,
         area_swept = EFFORT,
         gear = GEARNAME,
         season = SEASON,
         num = abundance,
         wgt = biomass) %>%
  #convert date to month and day columns
  mutate(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),
  ) %>%
  mutate(country = "United States",
         sub_area = NA,
         continent = "n_america",
         stat_rec = NA,
         verbatim_name = spp) %>%
  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 for duplicates, should not be any with more than 1 obs
#check for duplicates
count_seus <- seus %>%
  group_by(haul_id, verbatim_name) %>%
  mutate(count = n())

#none!

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

#empty

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

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

### Automatic cleaning
# Set Survey code
```

```
seus_survey_code <- "SEUS"

seus <- seus %>%
  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 (setting save = T means we will get an output of missing taxa)
clean_auto <- clean_taxa(unique(seus$taxa2), input_survey = seus_survey_code,
                         fishbase=T)
# takes 1.57 mins!

#this function sometimes throws an error, but if you restart your computer,
#it typically resolves

#Check those with no match from clean_taxa()

# Portunus spinimanus
# Ophichthus ocellatus (fish)
# Podochela sidneyi
# Astroscopus y-graecum (fish)
# Callinectes larvatus
# Charybdis hellerii
# Cryptopodia concava
# Sesarma cinereum
# Tremoctopus violaceus (common blanket octopus)

#two are fish, manually add back in

#manually add two more rows
wph_oce <- c("Ophichthus ocellatus", 275486,2651, "Myrichthys ocellatus","Animalia",
             "Chordata",
             "Actinopteri","Anguilliformes","Ophichthidae","Myrichthys","Species","SEUS")

ast_ygr <- c("Astroscopus y-graecum", 159252,3704, "Astroscopus y-graecum","Animalia",
             "Chordata",
             "Actinopteri","Perciformes","Uranoscopidae","Astroscopus","Species","SEUS")

clean_auto.missing <- rbind(clean_auto, wph_oce, ast_ygr)

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

clean_taxa <- clean_auto.missing %>%
  select(-survey) %>%
  filter(!(query == "Astroscopus y-graecum" & is.na(SpecCode)))

clean_seus <- left_join(seus, 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,
```

```r
  #non marine or non fish
  rename(accepted_name = taxa,
         aphia_id = worms_id) %>%
  mutate(verbatim_aphia_id = 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, verbatim_aphia_id, accepted_name, aphia_id, SpecCode,
         kingdom, phylum, class, order, family, genus, rank)

#check again for duplicates

count_clean_seus <- clean_seus %>%
  group_by(haul_id, accepted_name) %>%
  mutate(count = n())

#none!

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

#add final columns

clean_seus <- clean_seus %>%
  mutate(source = "NOAA",
         timestamp = my("04/2021"),
         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`)

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

# Just run this routine should be good for all
write_clean_data(data = clean_seus, survey = "SEUS", overwrite = T)



# -------------------------------------------------------------------------------------#
#### FAGS ####
# -------------------------------------------------------------------------------------#
#install required packages that are not already installed
```

```
required_packages <- c("data.table",
                       "devtools",
                       "dggridR",
                       "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_seus$survey))

#run flag_spp function in a loop
for (r in regions) {
  flag_spp(clean_seus, 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_seus, 7)

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

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


#------------------------------------------------------------------------------------------#
#### ADD STRANDARDIZATION FLAGS ####
#------------------------------------------------------------------------------------------#
surveys <- sort(unique(clean_seus$survey))
survey_units <- sort(unique(clean_seus$survey_unit))
```

```
survey_std <- clean_seus %>%
  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= ";")
    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])
```

```r
    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 = "SEUS_std",
                 overwrite = T, rdata=TRUE)
```

1. Overview of the survey data table

survey	source	timestamp	haul_id	country	sub_area
SEUS	NOAA	2021-04-01	198900143-80.28832.395	United States	NA
SEUS	NOAA	2021-04-01	198900143-80.28832.395	United States	NA
SEUS	NOAA	2021-04-01	198900143-80.28832.395	United States	NA
SEUS	NOAA	2021-04-01	198900143-80.28832.395	United States	NA
SEUS	NOAA	2021-04-01	198900143-80.28832.395	United States	NA

continent	stat_rec	station	stratum	year	month	day	quarter	season
n_america	NA	43M1	43	1989	4	12	2	spring
n_america	NA	43M1	43	1989	4	12	2	spring
n_america	NA	43M1	43	1989	4	12	2	spring
n_america	NA	43M1	43	1989	4	12	2	spring
n_america	NA	43M1	43	1989	4	12	2	spring

latitude	longitude	haul_dur	area_swept	gear	depth	sbt
32.395	-80.288	0.3333333	0.0181352	SC FALCON TRAWL W/O TED - DOUBLE RIGGED	7	14.4
32.395	-80.288	0.3333333	0.0181352	SC FALCON TRAWL W/O TED - DOUBLE RIGGED	7	14.4
32.395	-80.288	0.3333333	0.0181352	SC FALCON TRAWL W/O TED - DOUBLE RIGGED	7	14.4
32.395	-80.288	0.3333333	0.0181352	SC FALCON TRAWL W/O TED - DOUBLE RIGGED	7	14.4
32.395	-80.288	0.3333333	0.0181352	SC FALCON TRAWL W/O TED - DOUBLE RIGGED	7	14.4

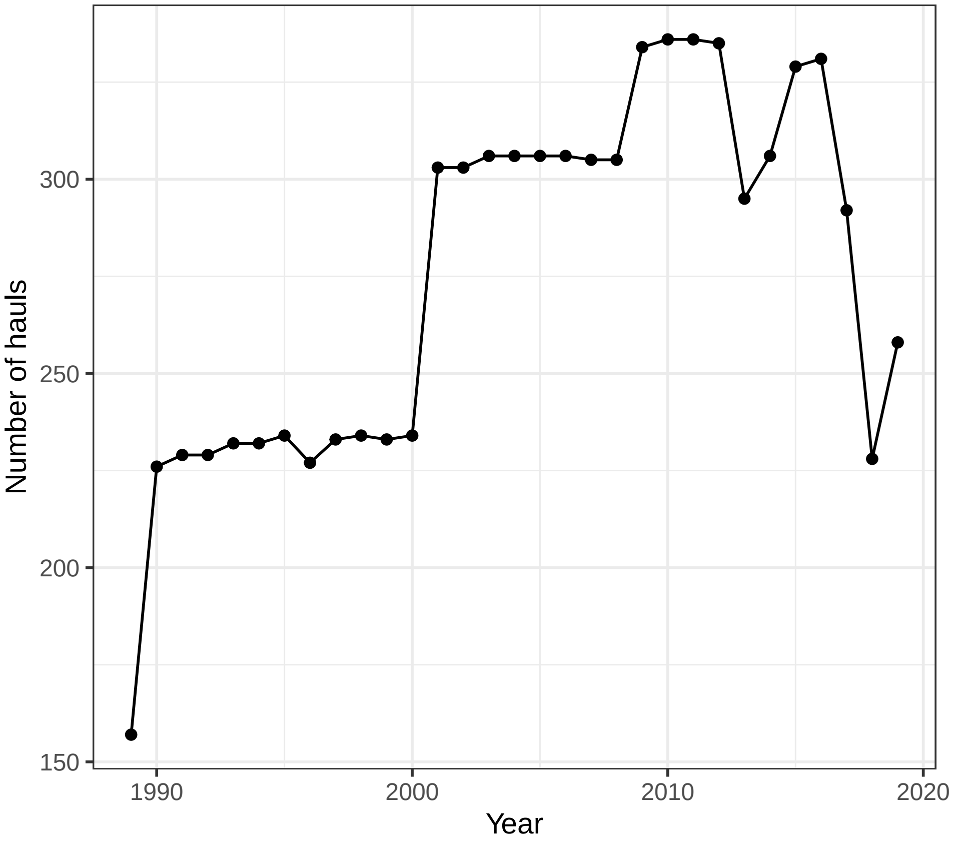
num	num_cpue	num_cpua	wgt	wgt_cpue	wgt_cpua	verbatim_name
42	126	1157.96768	0.144	0.432	3.9701749	ANCHOA
3	9	82.71198	0.034	0.102	0.9374024	ANCYLOPSETTA QUADROCELLATA
2	6	55.14132	0.014	0.042	0.3859892	CITHARICHTHYS MACROPS
1	3	27.57066	0.012	0.036	0.3308479	CYNOSCION NOTHUS
7	21	192.99461	9.400	28.200	259.1641962	DASYATIS AMERICANA

verbatim_aphia_id	accepted_name	aphia_id	SpecCode	kingdom
NA	Anchoa	158697	NA	Animalia
NA	Ancylopsetta quadrocellata	308768	NA	Animalia
NA	Citharichthys macrops	159165	4214	Animalia
NA	Cynoscion nothus	159305	1175	Animalia
NA	Hypanus americanus	1042856	1247	Animalia

phylum	class	order	family	genus	rank	survey_unit
Chordata	Teleostei	Clupeiformes	Engraulidae	Anchoa	Genus	SEUS-spring
Chordata	Teleostei	Pleuronectiformes	Paralichthyidae	Ancylopsetta	Species	SEUS-spring
Chordata	Teleostei	Pleuronectiformes	Paralichthyidae	Citharichthys	Species	SEUS-spring
Chordata	Teleostei	Eupercaria incertae sedis	Sciaenidae	Cynoscion	Species	SEUS-spring
Chordata	Elasmobranchii	Myliobatiformes	Dasyatidae	Dasyatis	Species	SEUS-spring

2. Summary of sampling intensity

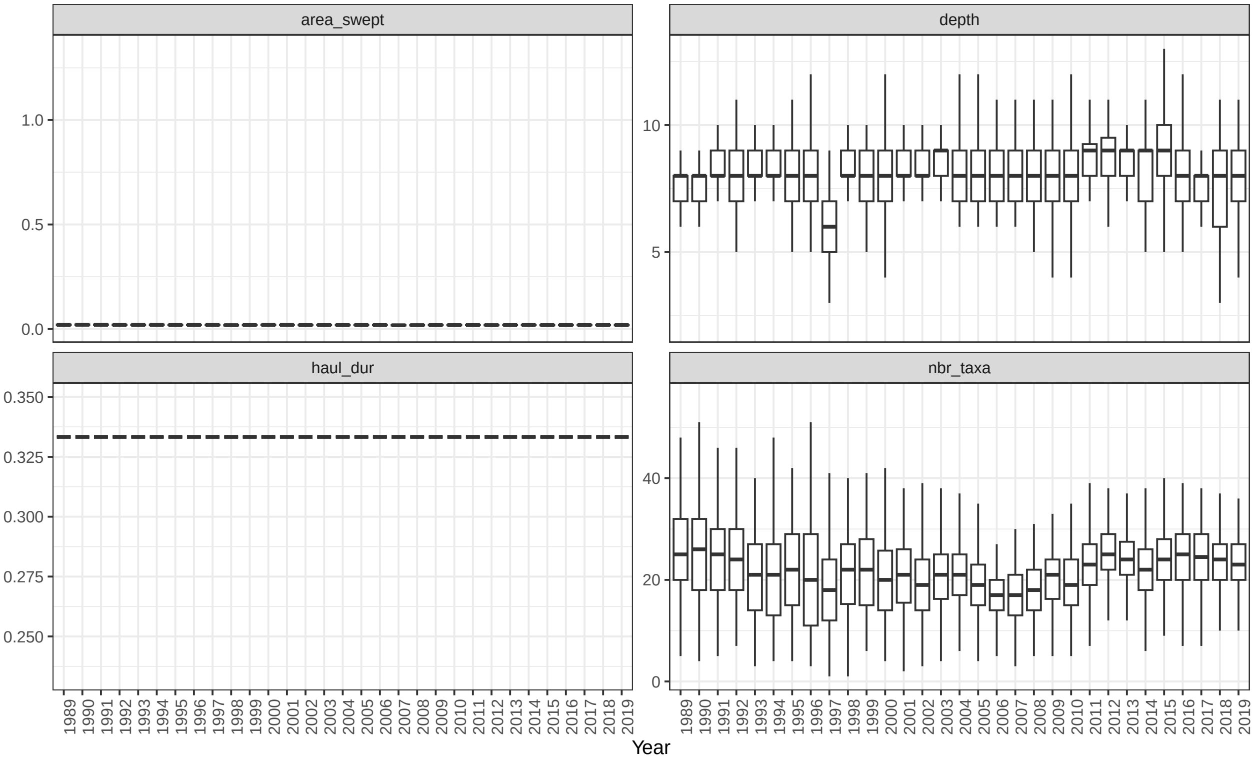
Number of hauls per year performed during the survey after data processing.



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:

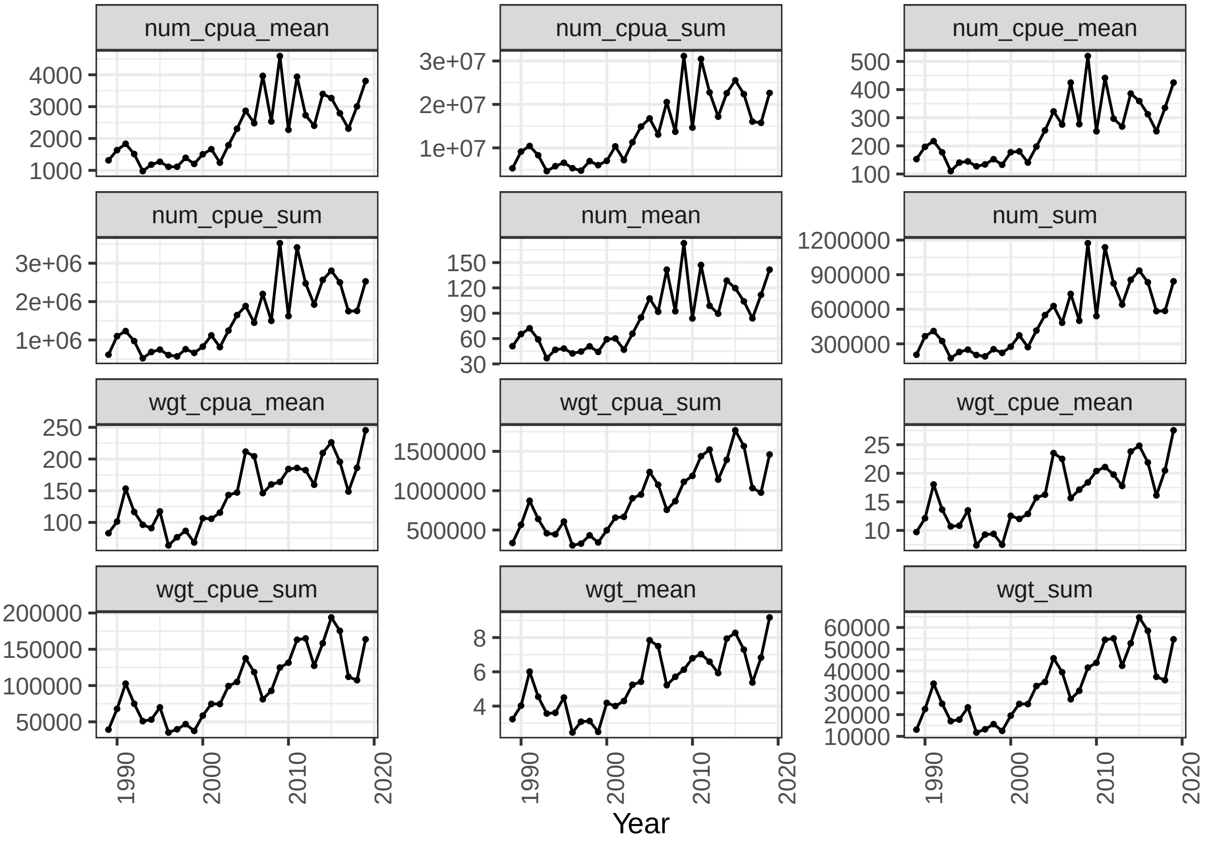
- *area_swept*, swept area by the bottom trawl gear km^2
- *depth*, sampling depth in m
- *haul_dur*, haul sampling duration *hour*
- *number of marine fish taxa*, taxa were cleaned following the last version of taxonomy from the World Register of Marine Species (https://www.marinespecies.org/, October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



4. Summary of biological variables

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

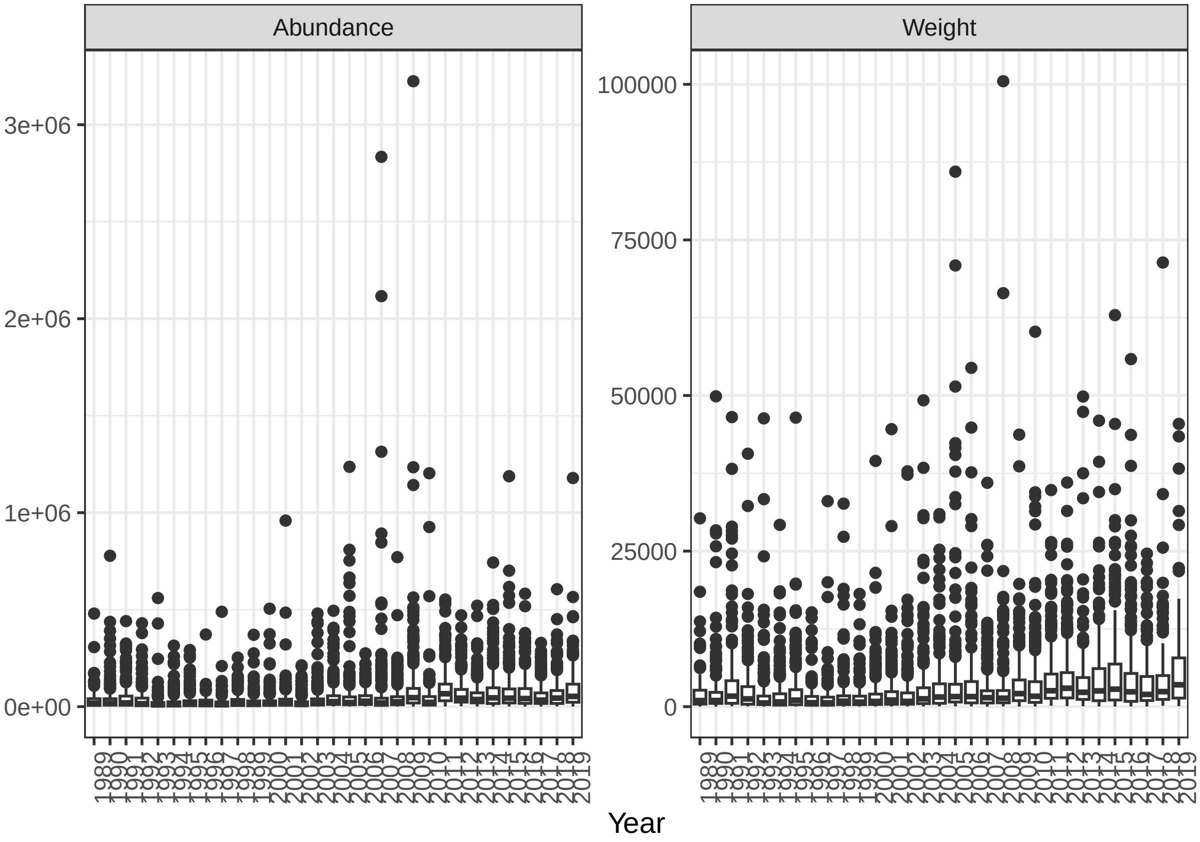
- *num_cpua*, number of individuals (abundance) in $\frac{individuals}{km^2}$
- *num_cpue*, number of individuals (abundance) in $\frac{individuals}{h}$
- *num*, number of individuals (abundance)
- *wgt_cpua*, weight in $\frac{kg}{km^2}$
- *wgt_cpue*, weight in $\frac{kg}{h}$
- *wgt*, weight in kg



5. Extreme values

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

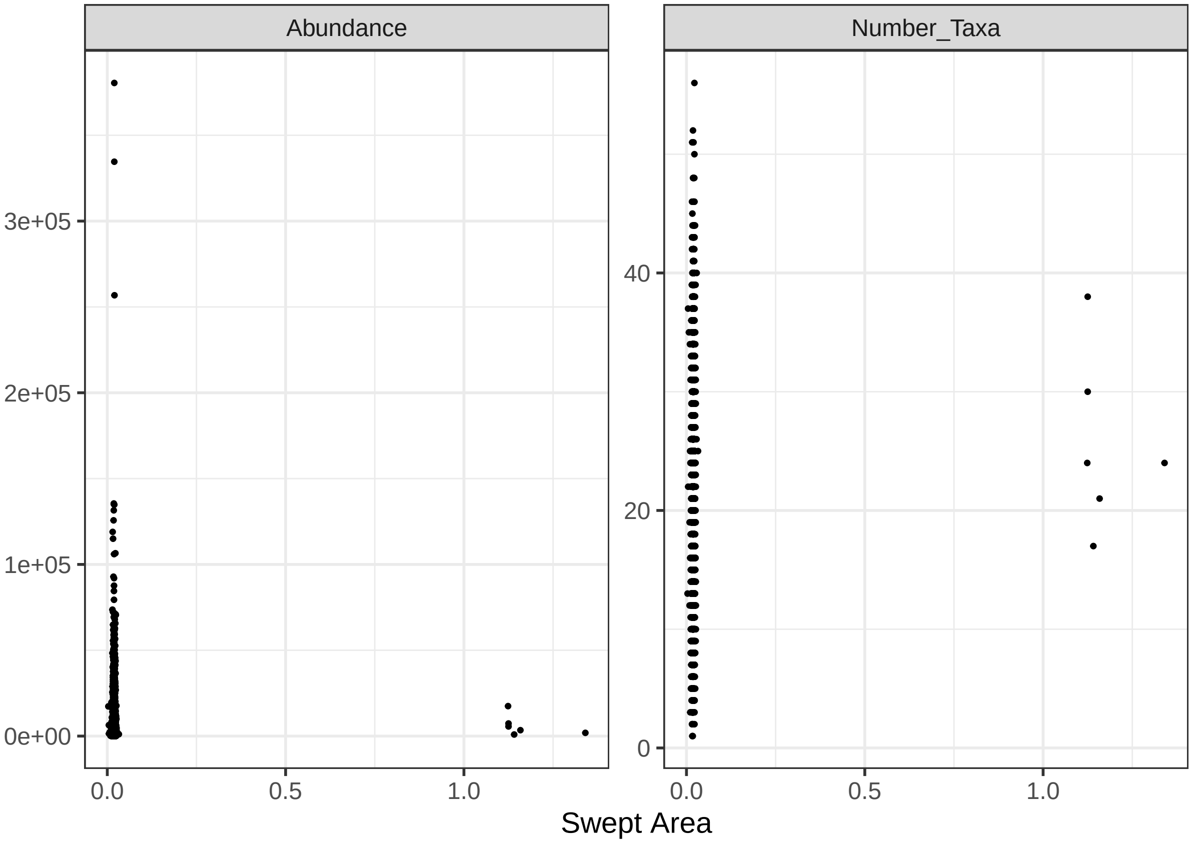
- *num_cpue*, number of individuals (abundance) in $\frac{individuals}{km^2}$
- *wgt_cpue*, weight in $\frac{kg}{km^2}$



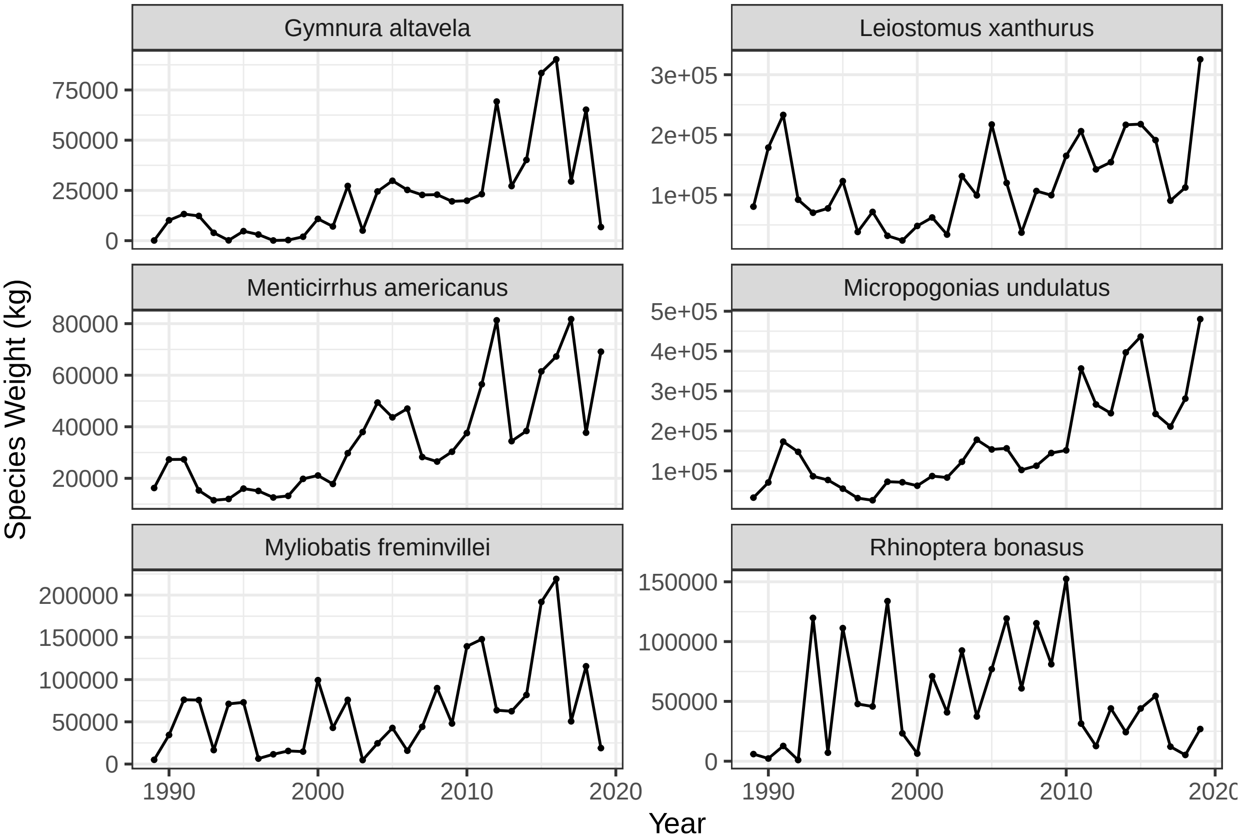
6. Summary of variables against swept area

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

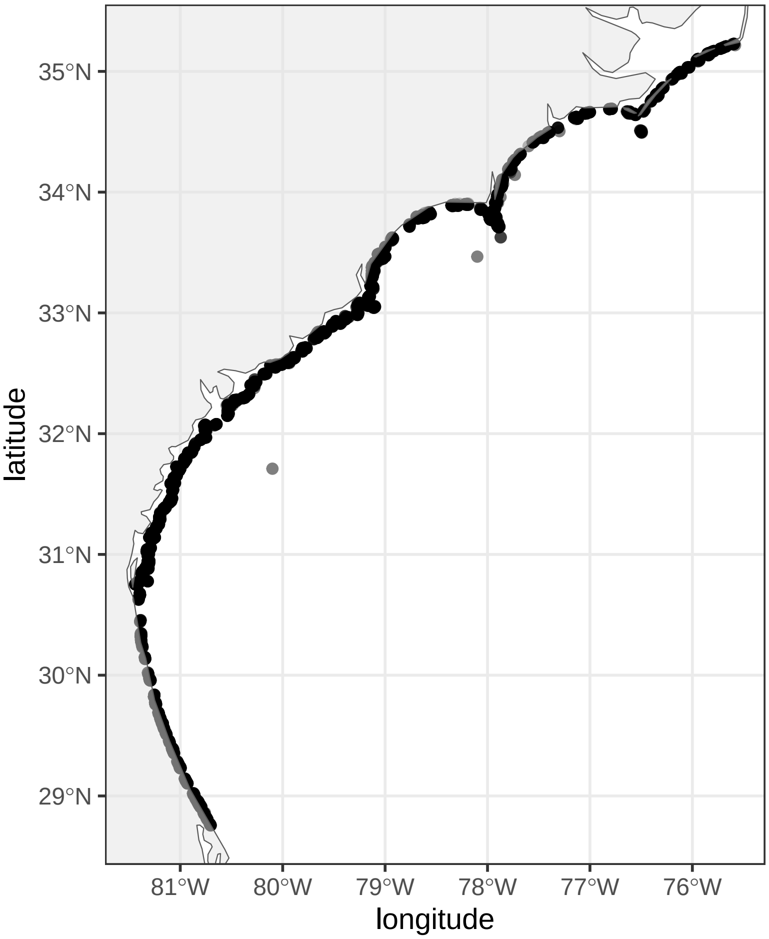


7. Abundance or Weight trends of the six most abundant species



8. Distribution mapping

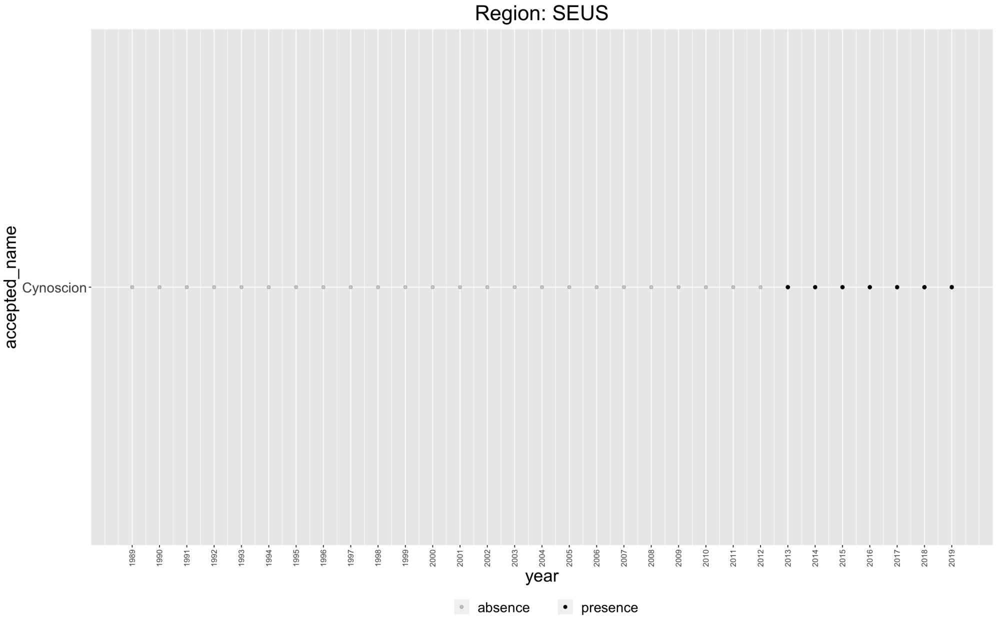
Map of the sampling distribution in space. Note that we only show one year per coordinate.



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

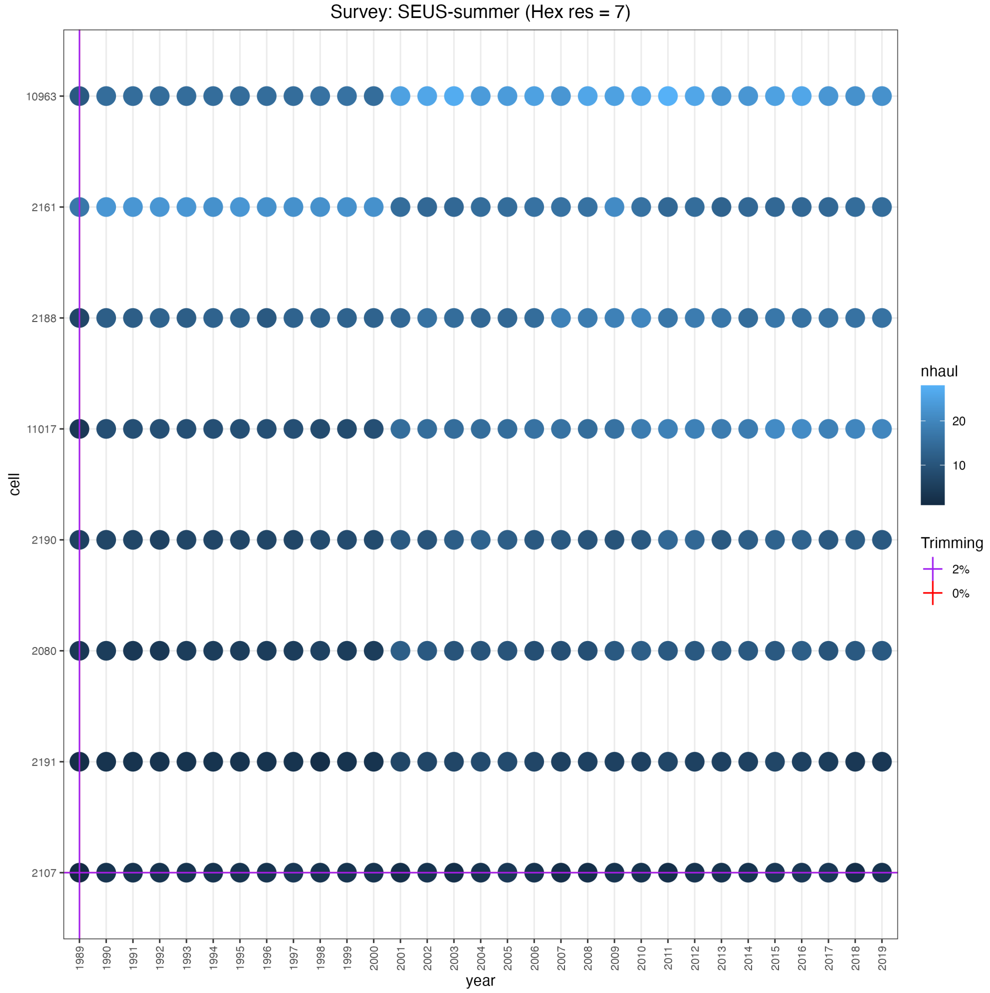
Total number of species	251.0
Percentage of species flagged	0.4

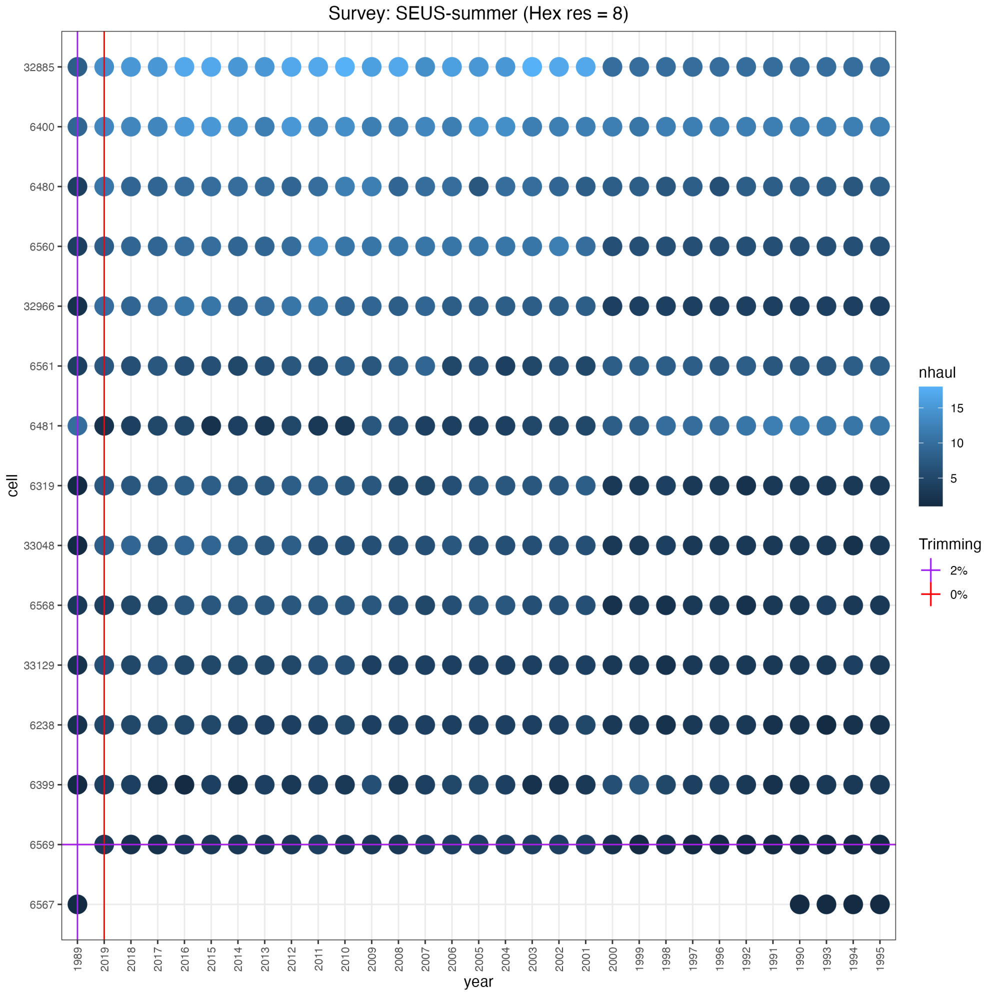
10. Spatio-temporal standardization: SEUS-summer

a. Standardization method 1

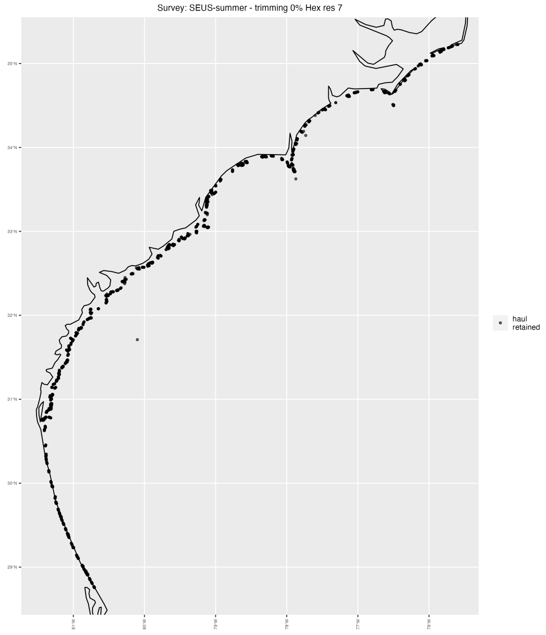
This standardization method was adapted from https://github.com/zoekitchel/trawl_spatial_turnover/blob/master/data_prep_code/species/explore_NorthSea_trimming.Rmd
It was run for hex resolution 7 and 8.

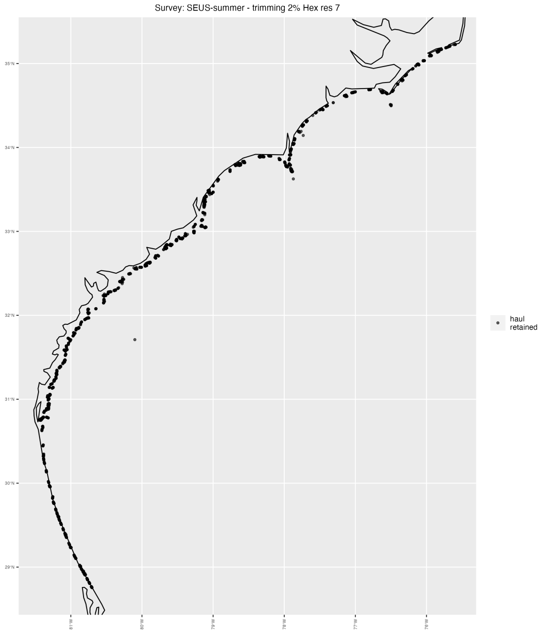
Plot of number of cells x years with overlaid flagging options

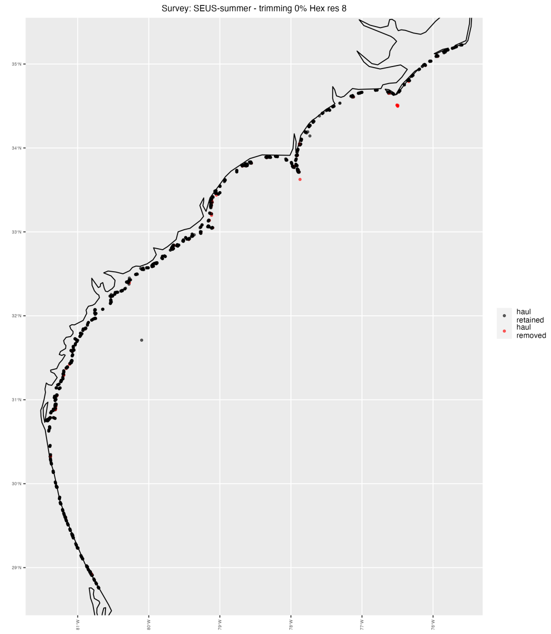


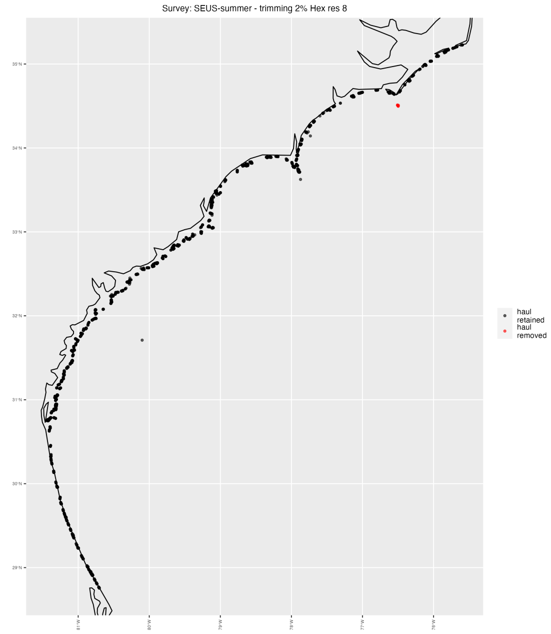


Map of hauls retained and removed per flagging method and threshold

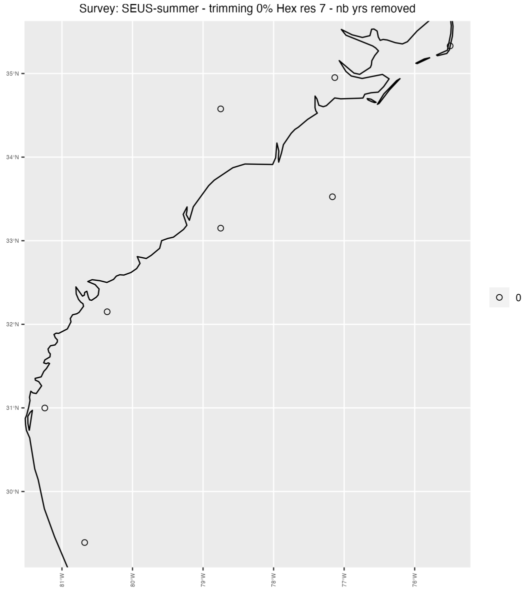


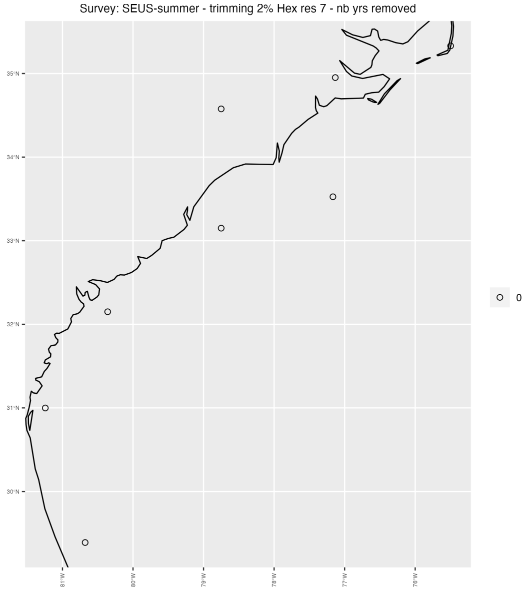


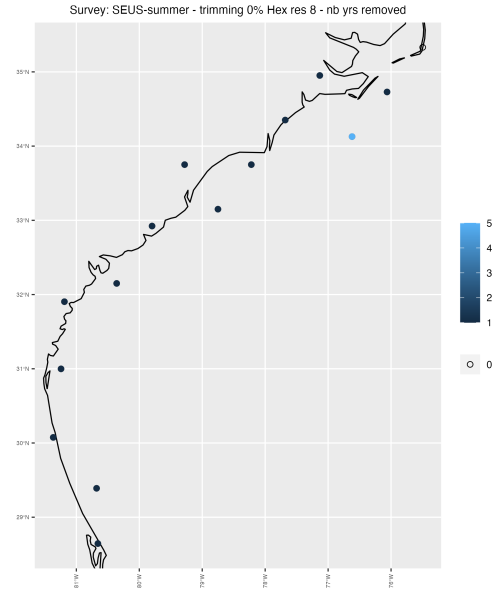


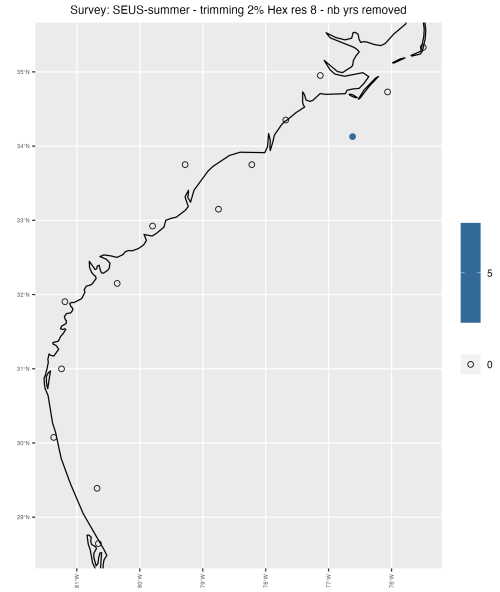


Map of numbers of years removed per grid cell and flagging method/threshold


Survey: SEUS-summer - trimming 0% Hex res 7 - nb yrs removed
35°N
34°N
33°N
32°N
31°N
30°N
81°W
80°W
79°W
78°W
77°W
76°W
0


Survey: SEUS-summer - trimming 2% Hex res 7 - nb yrs removed
35°N
34°N
33°N
32°N
31°N
30°N
81°W
80°W
79°W
78°W
77°W
76°W
0

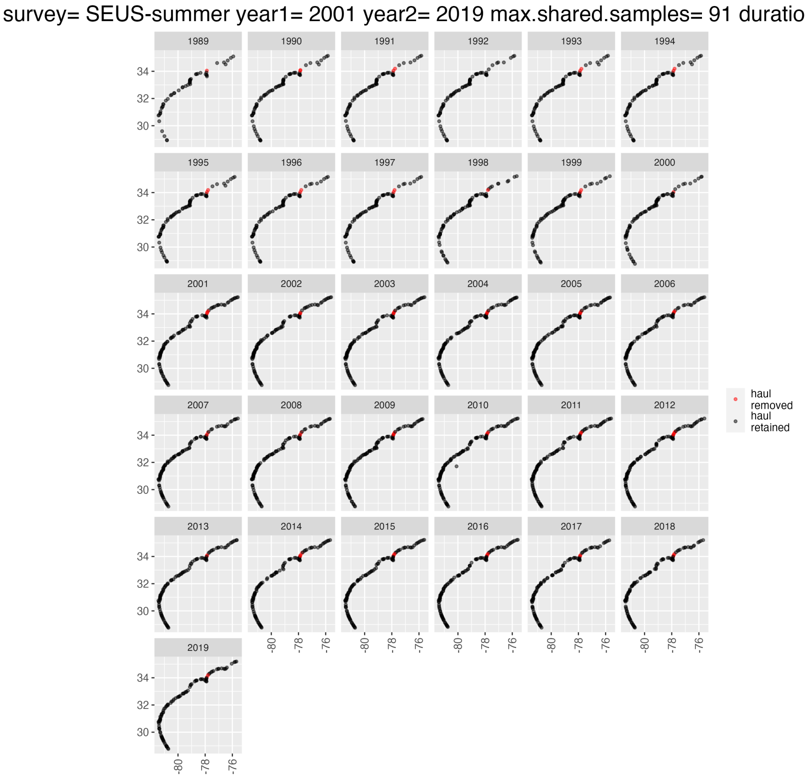

Survey: SEUS-summer - trimming 0% Hex res 8 - nb yrs removed
35°N
34°N
33°N
32°N
31°N
30°N
29°N
81°W
80°W
79°W
78°W
77°W
76°W
5
4
3
2
1
0


Survey: SEUS-summer - trimming 2% Hex res 8 - nb yrs removed
35°N
34°N
33°N
32°N
31°N
30°N
29°N
81°W
80°W
79°W
78°W
77°W
76°W
5
0

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

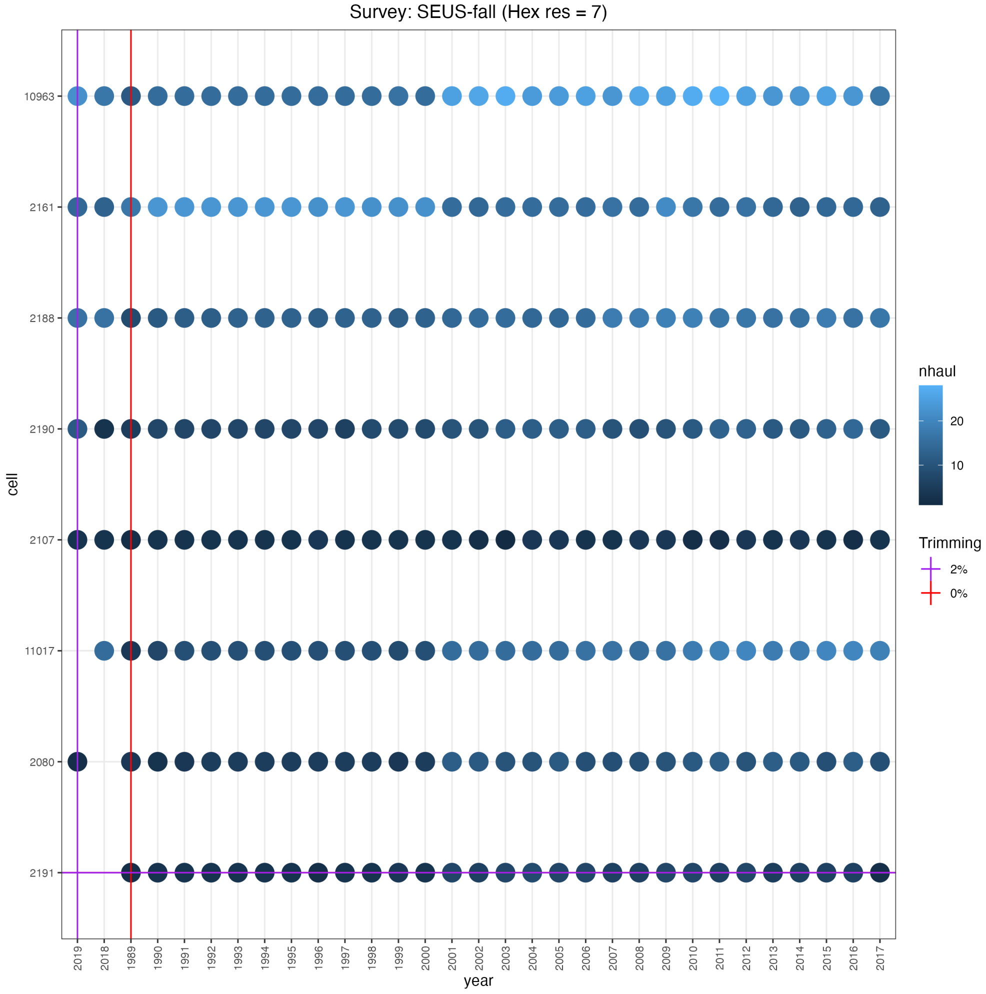
summary	grid cell 7, 0% threshold	grid cell 7, 2% threshold	grid cell 8, 0% threshold	grid cell 8, 2% threshold	method 2 (biotime)
number of hauls removed	0	0	56.0	5.0	1628.0
percentage of hauls removed	0	0	1.9	0.2	2.6

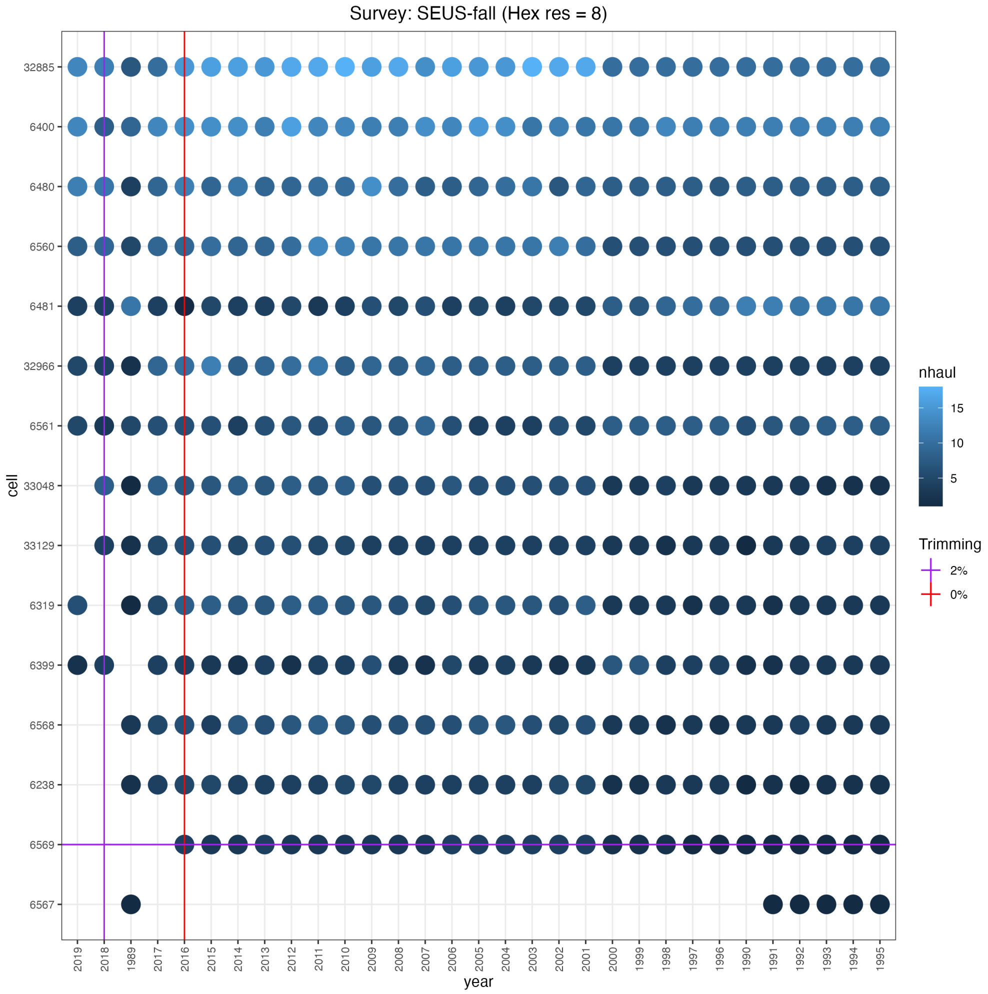
11. Spatio-temporal standardization: SEUS-fall

a. Standardization method 1

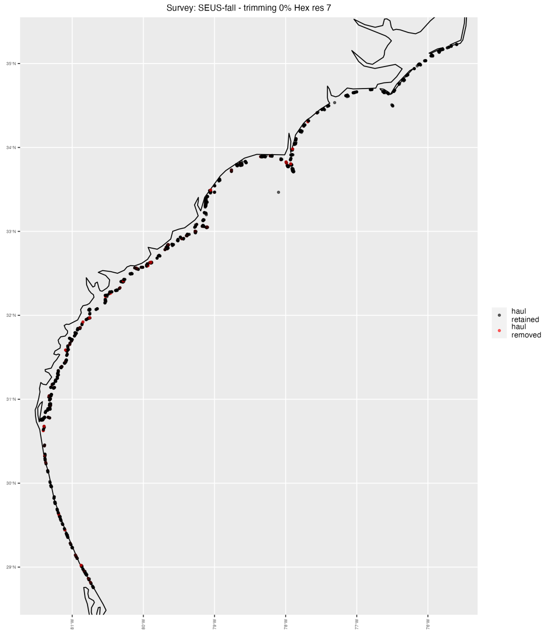
This standardization method was adapted from https://github.com/zoekitchel/trawl__spatial__turnover/blob/master/data__prep__code/species/explore__NorthSea__trimming.Rmd
It was run for hex resolution 7 and 8.

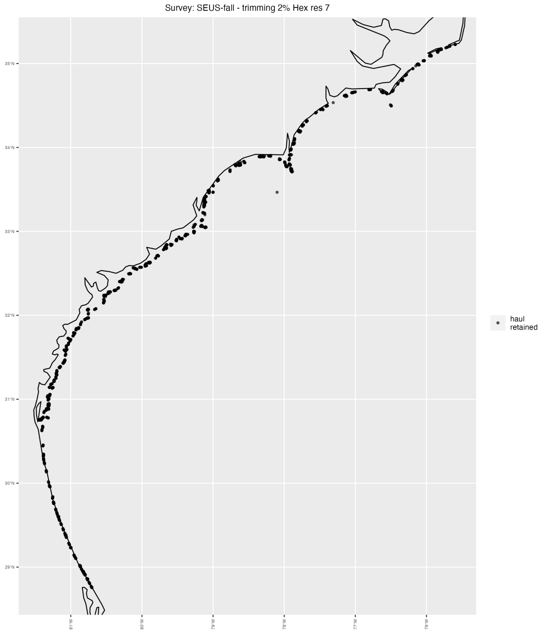
Plot of number of cells x years with overlaid flagging options

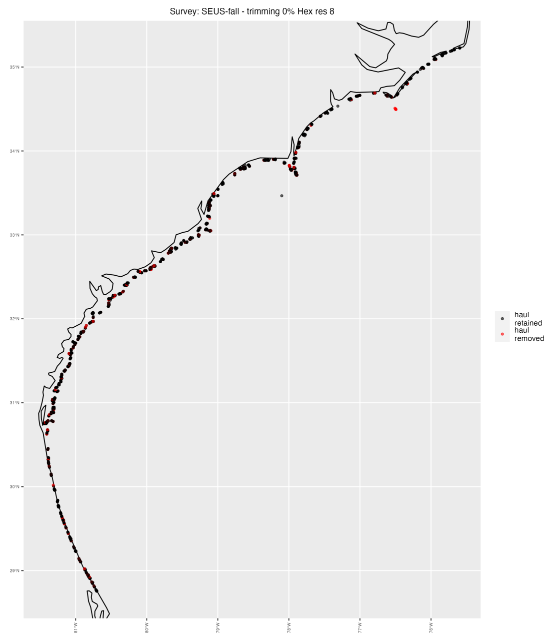


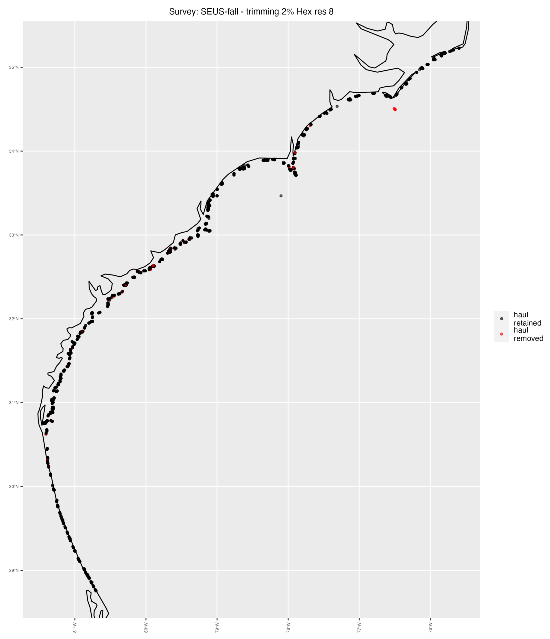


Map of hauls retained and removed per flagging method and threshold

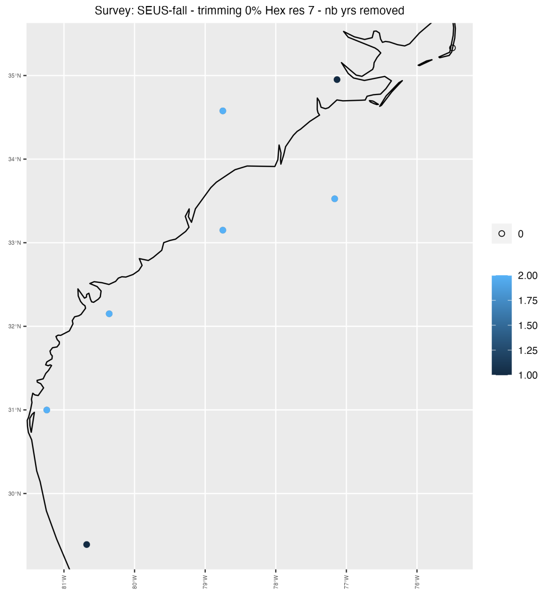


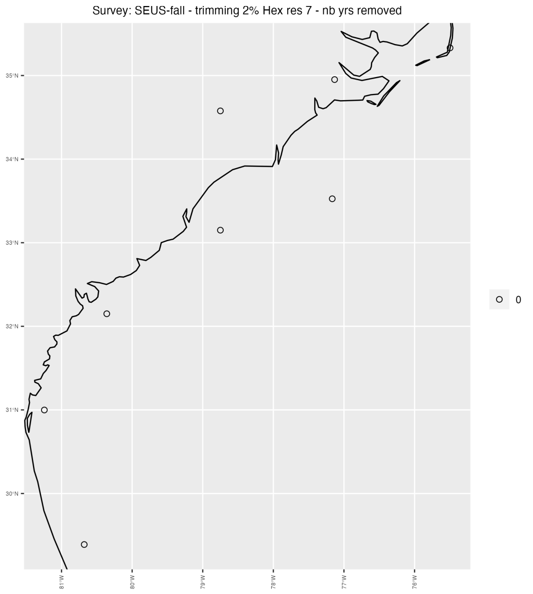


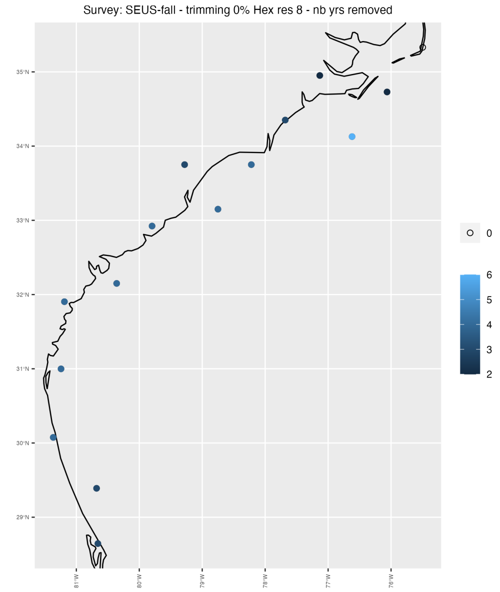




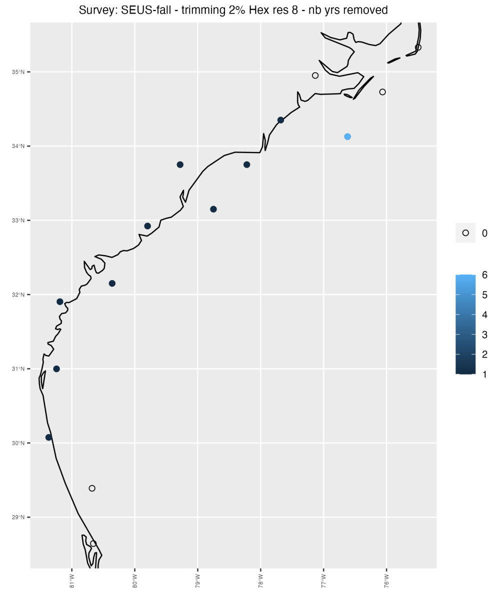
Map of numbers of years removed per grid cell and flagging method/threshold







Survey: SEUS-fall - trimming 0% Hex res 8 - nb yrs removed

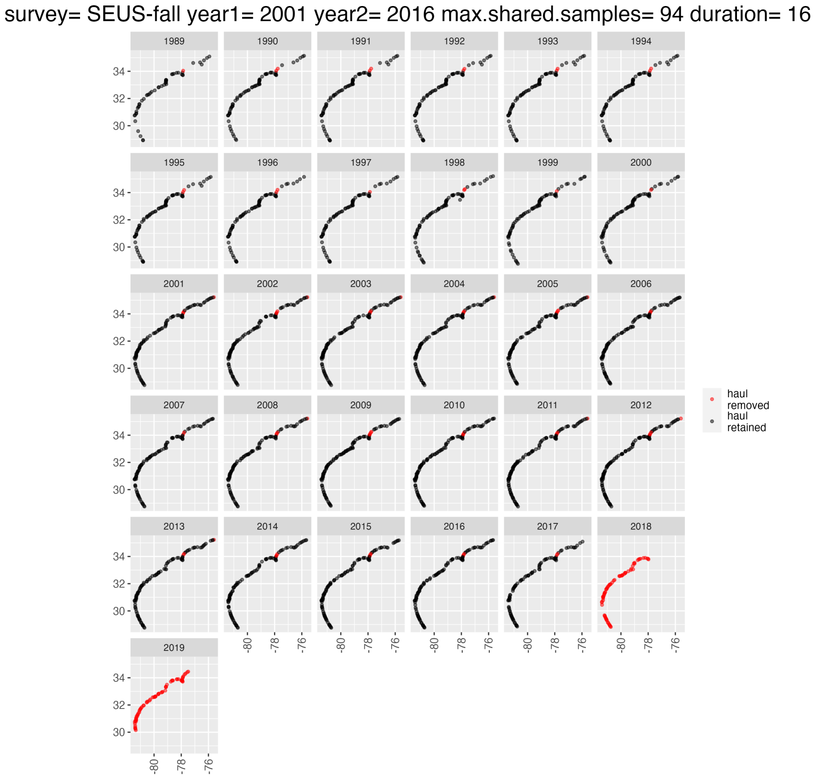


Survey: SEUS-fall - trimming 2% Hex res 8 - nb yrs removed

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



survey= SEUS-fall year1= 2001 year2= 2016 max.shared.samples= 94 duration= 16

c. Standardization summary

Statistics of hauls removed for each standardization method

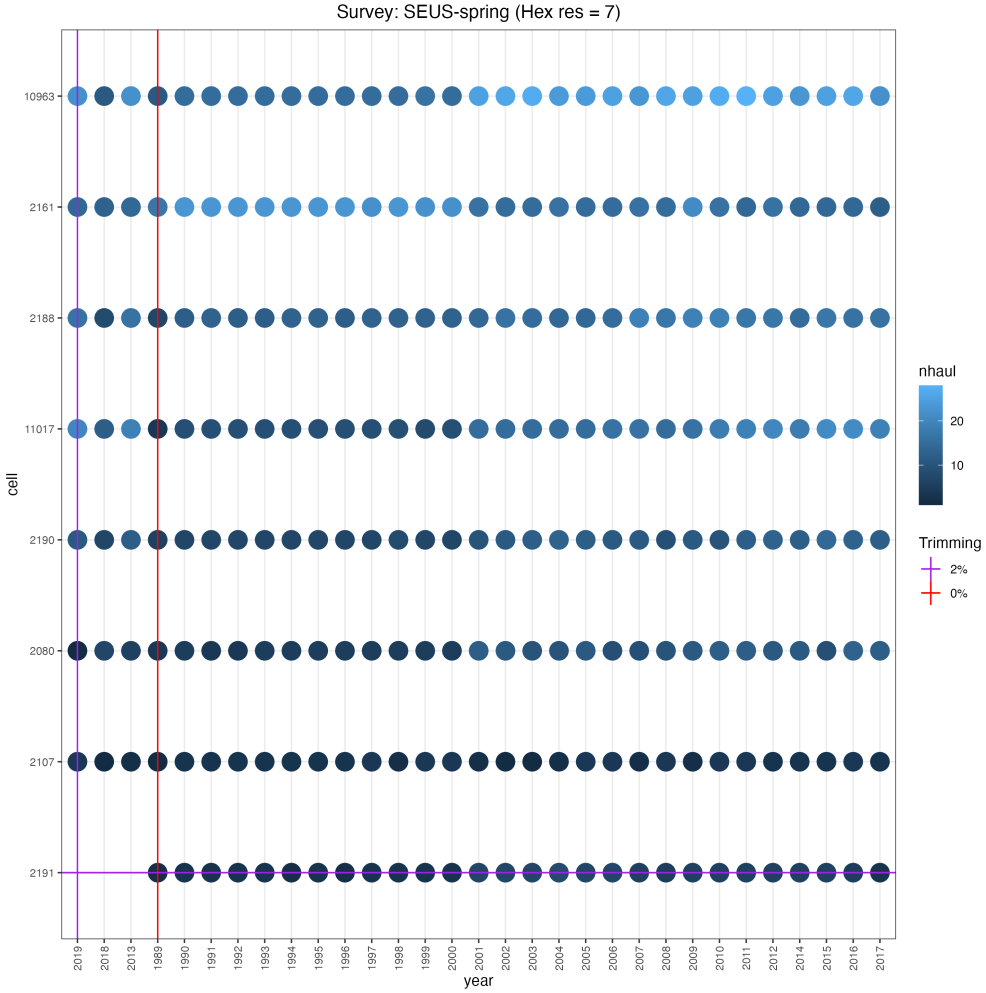
summary	grid cell 7, 0% threshold	grid cell 7, 2% threshold	grid cell 8, 0% threshold	grid cell 8, 2% threshold	method 2 (biotime)
number of hauls removed	135.0	0	284.0	74.0	5515.0
percentage of hauls removed	4.8	0	10.1	2.6	8.6

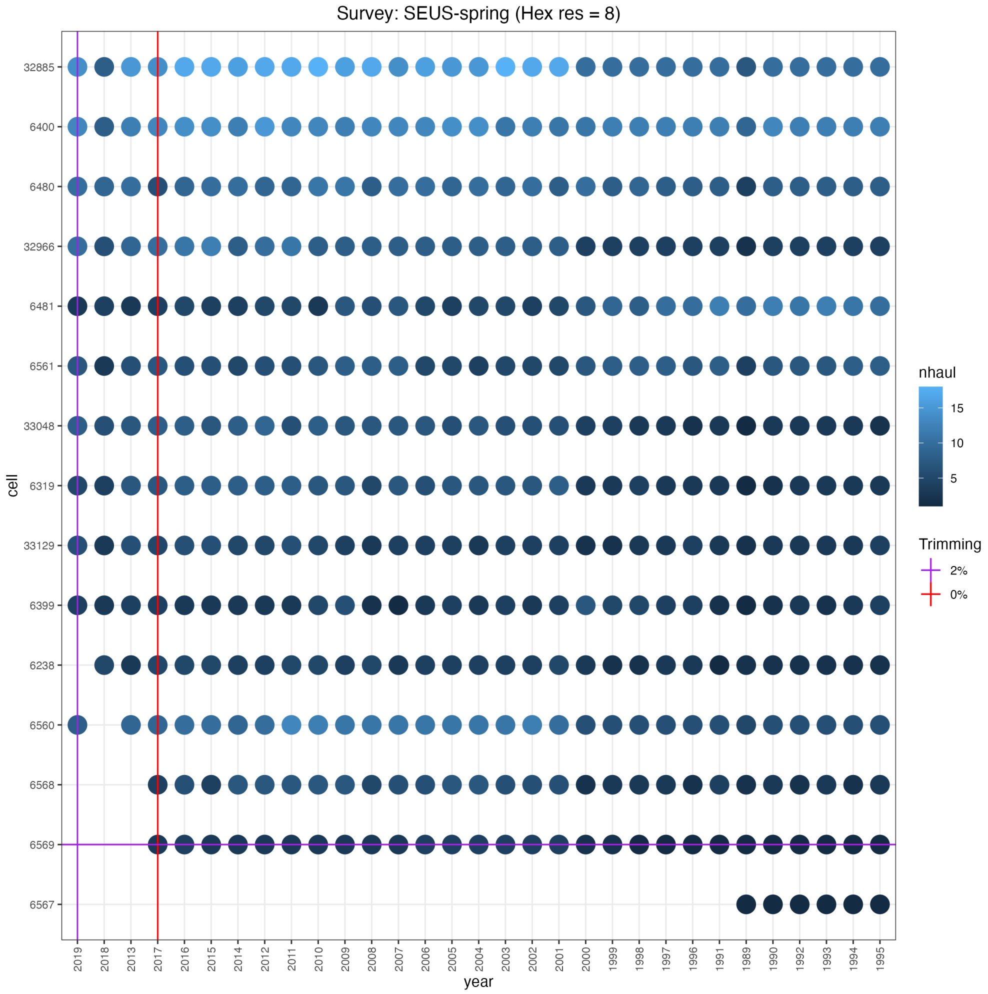
12. Spatio-temporal standardization: SEUS-spring

a. Standardization method 1

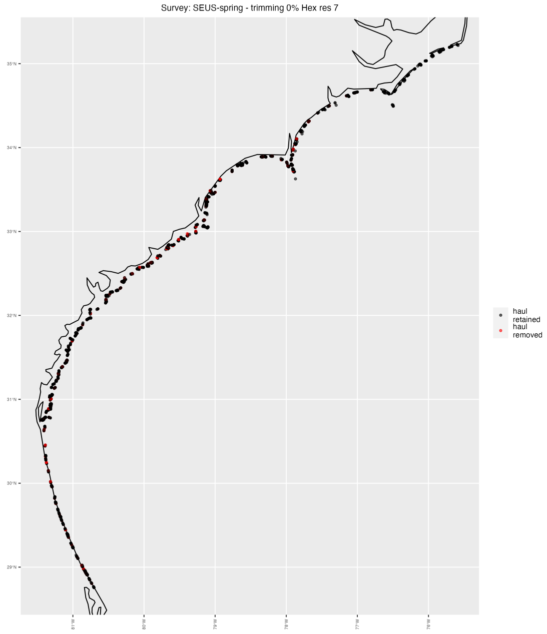
This standardization method was adapted from https://github.com/zoekitchel/trawl_spatial_turnover/blob/master/data_prep_code/species/explore_NorthSea_trimming.Rmd
It was run for hex resolution 7 and 8.

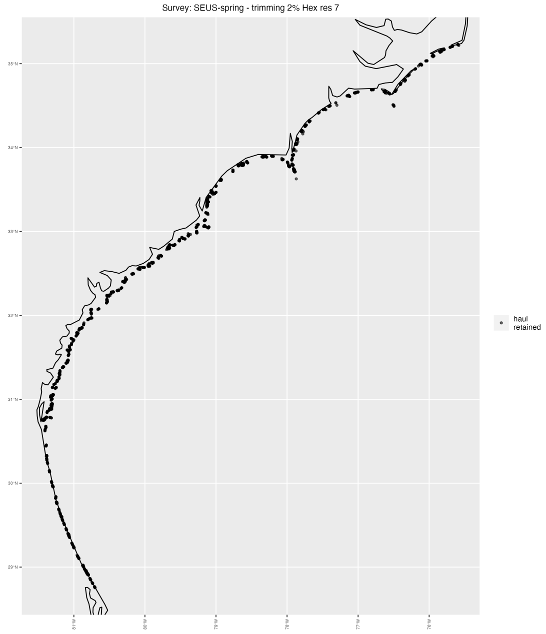
Plot of number of cells x years with overlaid flagging options

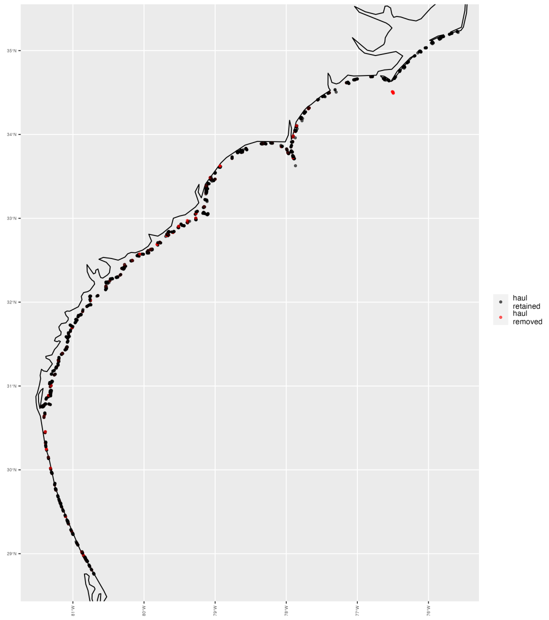




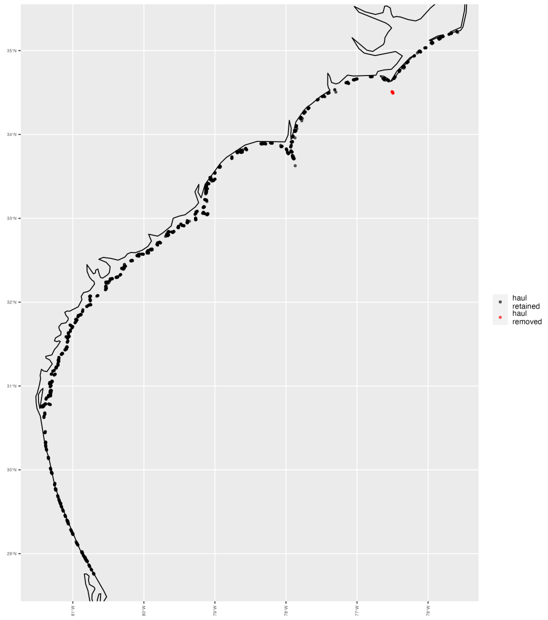
Map of hauls retained and removed per flagging method and threshold





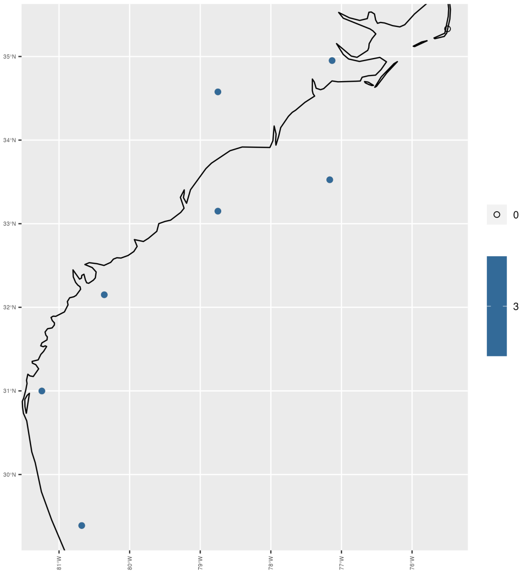


Survey: SEUS-spring - trimming 0% Hex res 8

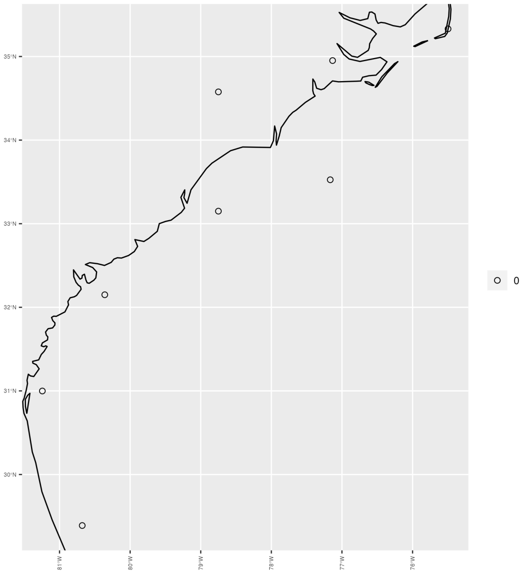


Survey: SEUS-spring - trimming 2% Hex res 8

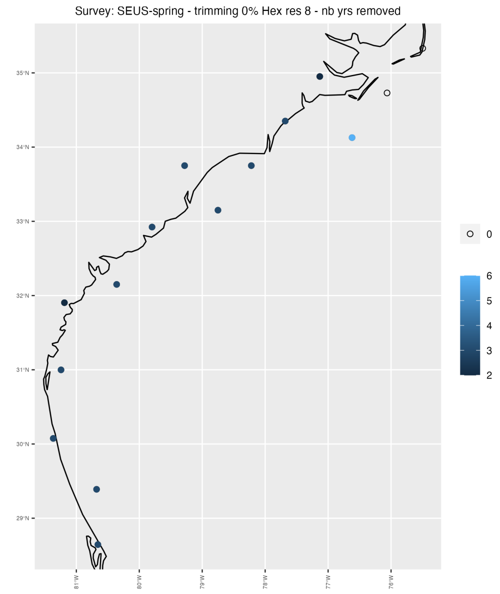
Map of numbers of years removed per grid cell and flagging method/threshold

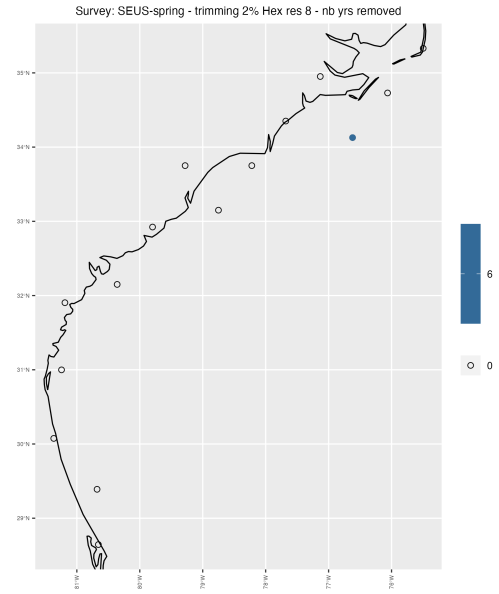


Survey: SEUS-spring - trimming 0% Hex res 7 - nb yrs removed



Survey: SEUS-spring - trimming 2% Hex res 7 - nb yrs removed

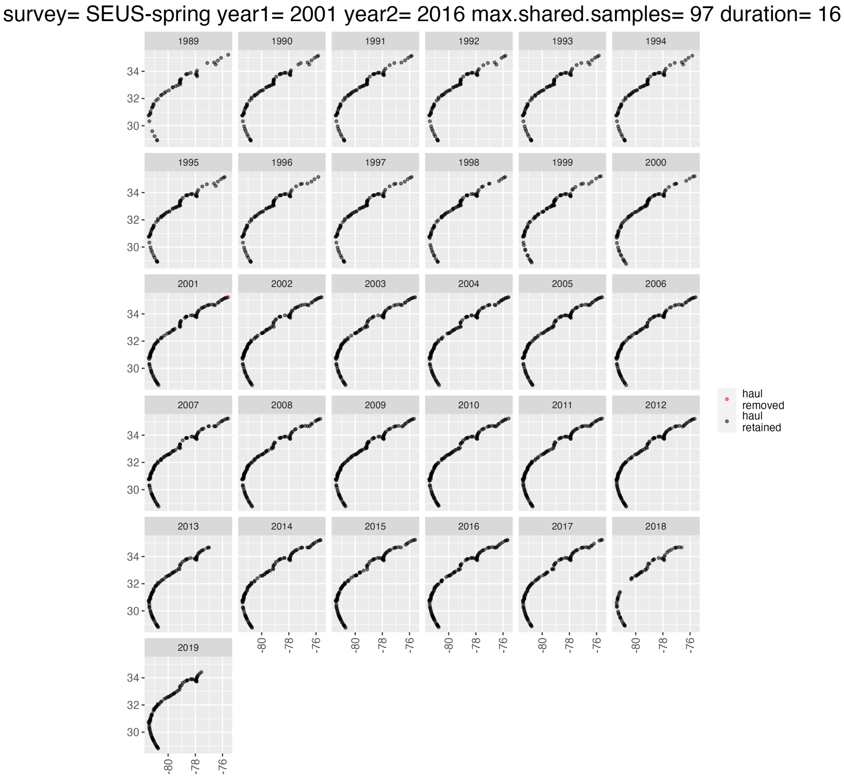

Survey: SEUS-spring - trimming 0% Hex res 8 - nb yrs removed
35°N
34°N
33°N
32°N
31°N
30°N
29°N
81°W
80°W
79°W
78°W
77°W
76°W
0
6
5
4
3
2


Survey: SEUS-spring - trimming 2% Hex res 8 - nb yrs removed
35°N
34°N
33°N
32°N
31°N
30°N
29°N
81°W
80°W
79°W
78°W
77°W
76°W
6
0

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	238.0	0	244.0	6.0	19
percentage of hauls removed	8.4	0	8.6	0.2	0