

SCS: Canadian Maritimes 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	11
2. Summary of sampling intensity	12
3. Summary of sampling variables from the survey	13
4. Summary of biological variables	14
5. Extreme values	15
6. Summary of variables against swept area	16
7. Abundance or Weight trends of the six most abundant species	17
8. Distribution mapping	18
9. Taxonomic flagging	18
10. Spatio-temporal standardization: SCS-SUMMER	18
a. Standardization method 1	18
b. Standardization method 2	22
c. Standardization summary	22
11. Spatio-temporal standardization: SCS-FALL	23
a. Standardization method 1	23
b. Standardization method 2	26
c. Standardization summary	26
12. Spatio-temporal standardization: SCS-SPRING	27
a. Standardization method 1	27
b. Standardization method 2	30
c. Standardization summary	30

General info

This document presents the cleaning code and summary of the Canadian Maritimes bottom trawl survey provided by Mike McMahon, Don Clark, and Brian Bower. It contains data from 1970 and up to 2020.

Data cleaning in R

```
#####
##### R code to clean trawl survey for Canadian Maritimes
##### Public data Ocean Adapt
##### Contacts: Mike McMahon mike.mcmahon@dfo-mpo.gc.ca Aquatic Science Biologist
##### Population Ecology Division, DFO Canada
##### Don Clark don.clark@dfo-mpo.gc.ca Biologist, DFO Canada
##### Brian Bower brian.bower@dfo-mpo.gc.ca
##### GIS Analyst/ Physical Scientist at Fisheries and Oceans Canada
##### Coding: Michelle Stuart, Dan Forrest, Zoë Kitchel November 2021
```



```

"https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/MAR_SPRING_SPP.csv",
"https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/MAR_SUMMER_SPP.csv")

#spp_files <- as.list(dir(pattern = "_SPP", path = "data_raw", full.names = T))
mar_spp <- spp_files %>% #this pulls in species from all three surveys, so there are
  #some repeats which I remove below
  map_dfr(~ read_csv(.x, col_types = cols(
    SPEC = col_character()
  )))

mar_spp <- mar_spp %>%
  rename(spp = SPEC,
         SPEC = CODE) %>%
  distinct()

mission_files <- list(
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/MAR_FALL_MISSION.csv",
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/MAR_SPRING_MISSION.csv",
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/MAR_SUMMER_MISSION.csv")
#mission_files <- as.list(dir(pattern = "_MISSION", path = "data_raw", full.names = T))
mar_missions <- mission_files %>%
  map_dfr(~ read_csv(.x, col_types = cols(
    .default = col_double(),
    MISSION = col_character(),
    VESEL = col_character(),
    SEASON = col_character()
  )))

info_files <- list(
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/MAR_FALL_INF.csv",
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/MAR_SPRING_INF.csv",
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/MAR_SUMMER_INF.csv")
#info_files <- as.list(dir(pattern = "_INF", path = "data_raw", full.names = T))
mar_info <- info_files %>%
  map_dfr(~ read_csv(.x, col_types = cols(
    .default = col_double(),
    MISSION = col_character(),
    SDATE = col_character(),
    GEARDESC = col_character(),
    STRAT = col_character()
  )))

catch_files <- list(
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/MAR_FALL_CATCH.csv",
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/MAR_SPRING_CATCH.csv",
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/MAR_SUMMER_CATCH.csv")
#catch_files <- as.list(dir(pattern = "_CATCH", path = "data_raw", full.names = T))
mar_catch <- catch_files %>%
  map_dfr(~ read_csv(.x, col_types = cols(
    .default = col_double(),
    MISSION = col_character()
  )))

```

```

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


mar <- left_join(mar_catch, mar_missions, by = "MISSION")

mar <- mar %>%
  # Create a unique haul_id
  mutate(
    haul_id = paste(formatC(MISSION, width=3, flag=0),
                    formatC(SETNO, width=3, flag=0), sep = "_"))

mar_info <- mar_info %>%
  # Create a unique haul_id
  mutate(
    haul_id = paste(formatC(MISSION, width=3, flag=0),
                    formatC(SETNO, width=3, flag=0), sep = "_"))

mar <- left_join(mar, mar_info, by = c("haul_id", "MISSION", "SETNO")) #206202 rows
mar <- left_join(mar, mar_spp, by = "SPEC")
mar$survey <- "SCS"

names(mar) <- tolower(names(mar))

mar <- mar %>%
  # convert mission to haul_id
  rename(wgt = totwgt,
         num = totno,
         latitude = slat,
         longitude = slong,
         stratum = strat,
         gear = geardesc,
         sbt = bott_temp,
         sst = surf_temp,
         verbatim_name = spp,
         year = year,
         depth = depth) %>%
  # area swept by net in km^2 = 33.6 m door spread *
  #DIST in nautical miles * 1852 m/1 nautical mile * 1 km^2/1000000 m^2
  mutate(area_swept = 33.6 * dist * 1852 *(1/1000000),
         month = month(as.Date(sdate)),
         day = day(as.Date(sdate)),
         haul_dur = dur/60) #minutes to hours

# Does the spp column contain any eggs or non-organism notes?
#As of 2021, only "UNIDENTIFIED" to be removed
test <- mar %>%
  select(verbatim_name) %>%
  filter(!is.na(verbatim_name)) %>%
  distinct() %>%

```

```

filter((grepl("egg", verbatim_name) & grepl("", verbatim_name)) |
       grepl("UNIDENTIFIED", verbatim_name)) #does it contain egg or unidentified?
stopifnot(nrow(test)==0)

#delete any rows with any of these
mar <- mar %>% #206202 to 205205 rows
  filter(!grepl("UNIDENTIFIED",verbatim_name))

#check that the number of unique haul_ids * spp combinations is the same as
#the number of rows in mar
nrow(mar) == nrow(unique(mar[,c("haul_id","verbatim_name")]))

#it's not, so let's see why we have extras
#which(duplicated(mar[,c("haul_id","verbatim_name")]))

# combine the wtcpue for each species by haul
mar <- mar %>%
  mutate(
    wgt_cpue = wgt/area_swept,
    wgt_h = wgt/haul_dur, #may need to change this unit, currently in minutes
    num_cpue = num/area_swept,
    num_h = num/haul_dur

  )

mar <- mar %>%
# Adding extra columns and setting proper format
  mutate(
    country = "Canada",
    source = "DFO",
    timestamp = mdy("02/08/2021"),
    sub_area = NA,
    continent = "n_america",
    stat_rec = NA,
    station = NA,
    quarter = ifelse(month %in% c(1,2,3),1,
                     ifelse(month %in% c(4,5,6),2,
                           ifelse(month %in% c(7,8,9),3,
                                 4
                               )
                             )
                   ),
    verbatim_aphia_id = 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, verbatim_name, num, num_h, num_cpue,
         wgt, wgt_h, wgt_cpue, verbatim_name, verbatim_aphia_id)

#check for duplicates, should not be any with more than 1 obs
#check for duplicates
count_mar <- mar %>%
  group_by(haul_id, verbatim_name) %>%

```

```

    mutate(count = n())

#none!

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

unique_name_match
#empty (fixed earlier in ~178)

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


# Get WoRMS id for sourcing
wrms <- gnr_datasources() %>%
  filter(title == "World Register of Marine Species") %>%
  pull(id)

### Automatic cleaning
# Set Survey code
scs_survey_code <- "SCS"

scs <- mar %>%
  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(scs$taxa2), input_survey = scs_survey_code,
                           save = F, output=NA, fishbase=T)
#takes 3.9 minutes

#This leaves out the following species, of which 2 are fish that need to be added back
#Caelorinchus caelorinchus      #fish
#Porania pulvillus
#Poraniomorpha borealis
#Notoscopelus elongatus kroyeri #fish, different fishbase record for Noto elon and
#                                Noto kroy
#                                Noto elon is endemic to Mediterranean, so we
#                                will move forward as if this is Notoscopelus kroyeri
#Spirontocaris fabricii
#Nereidae
#Coelenterata

cae_cae <- c("Caelorinchus caelorinchus", "398381", "1726", "Coelorinchus caelorrhincus",
            "Animalia", "Chordata", "Actinopteri", "Gadiformes", "Macrouridae",
            "Coelorinchus", "Species",

```

```

    "SCS")
not_el0 <- c("Notoscopelus elegatus kroyeri", "272728", "27753", "Notoscopelus kroyeri",
           "Animalia", "Chordata", "Actinopteri", "Myctophiformes", "Myctophidae",
           "Notoscopelus", "Species",
           "SCS")

clean_auto_missing <- rbind(clean_auto, cae_cae, not_el0)

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


correct_taxa <- clean_auto_missing %>%
  select(-survey)

clean_scs <- left_join(scs, correct_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_scs <- clean_scs %>%
  group_by(haul_id, accepted_name) %>%
  mutate(count = n())

#none!

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

unique_name_match
#empty

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

```

```

# -----#
# Just run this routine should be good for all
write_clean_data(data = clean_scs, survey = "SCS", 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_scs$survey))

#run flag_spp function in a loop
for (r in regions) {
  flag_spp(clean_scs, 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_scs, 7)

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

```

```

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

#-----#
#### ADD STANDARDIZATION FLAGS #####
#-----#
surveys <- sort(unique(clean_scs$survey))
survey_units <- sort(unique(clean_scs$survey_unit))
survey_std <- clean_scs %>%
  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], "_flagsppl.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 = "SCS_std",
                 overwrite = T, rdata=TRUE)

```

1. Overview of the survey data table

survey	source	timestamp	haul_id	country	sub_area
SCS	DFO	2021-02-08	HAM1982085_054	Canada	NA
SCS	DFO	2021-02-08	HAM1982084_011	Canada	NA
SCS	DFO	2021-02-08	HAM1982084_019	Canada	NA
SCS	DFO	2021-02-08	HAM1981064_025	Canada	NA
SCS	DFO	2021-02-08	HAM1982084_054	Canada	NA

continent	stat_rec	station	stratum	year	month	day	quarter	season
n_america	NA	NA	481	1982	10	20	4	FALL
n_america	NA	NA	454	1982	9	29	3	FALL
n_america	NA	NA	455	1982	9	29	3	FALL
n_america	NA	NA	464	1981	10	3	4	FALL
n_america	NA	NA	443	1982	10	4	4	FALL

latitude	longitude	haul_dur	area_swept	gear	depth	sbt	sst
42.48333	-65.40000	0.5	0.1057862	Western IIA trawl	98.76	7.20	12.8
43.26667	-61.26667	0.5	0.1057862	Western IIA trawl	118.87	2.56	17.9
43.86667	-60.33333	0.5	0.1120090	Western IIA trawl	31.09	15.42	16.2
43.73333	-61.38333	0.5	0.1057862	Western IIA trawl	51.21	12.45	15.5
45.35000	-59.26667	0.5	0.1120090	Western IIA trawl	91.44	1.46	13.5

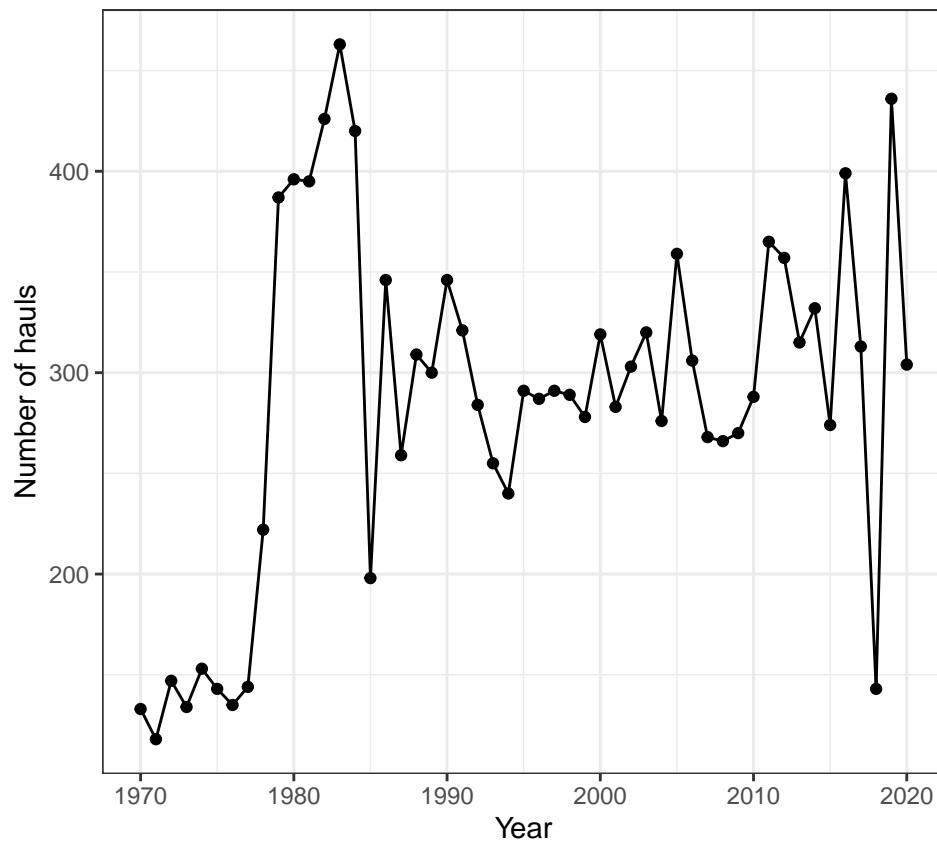
num	num_cpue	num_ccpuia	wgt	wgt_cpue	wgt_ccpuia	verbatim_name
1	2	9.453025	0	0	0	BALISTES CAPRISCUS
1	2	9.453025	0	0	0	STEPHANOLEPIS HISPIDUS
1	2	8.927857	0	0	0	STEPHANOLEPIS HISPIDUS
1	2	9.453025	0	0	0	STEPHANOLEPIS HISPIDUS
1	2	8.927857	0	0	0	STEPHANOLEPIS HISPIDUS

verbatim_aphia_id	accepted_name	aphia_id	SpecCode	kingdom
NA	Balistes capriscus	154721	7327	Animalia
NA	Stephanolepis hispida	307126	4281	Animalia
NA	Stephanolepis hispida	307126	4281	Animalia
NA	Stephanolepis hispida	307126	4281	Animalia
NA	Stephanolepis hispida	307126	4281	Animalia

phylum	class	order	family	genus	rank	survey_unit
Chordata	Teleostei	Tetraodontiformes	Balistidae	Balistes	Species	SCS-FALL
Chordata	Teleostei	Tetraodontiformes	Monacanthidae	Stephanolepis	Species	SCS-FALL
Chordata	Teleostei	Tetraodontiformes	Monacanthidae	Stephanolepis	Species	SCS-FALL
Chordata	Teleostei	Tetraodontiformes	Monacanthidae	Stephanolepis	Species	SCS-FALL
Chordata	Teleostei	Tetraodontiformes	Monacanthidae	Stephanolepis	Species	SCS-FALL

2. Summary of sampling intensity

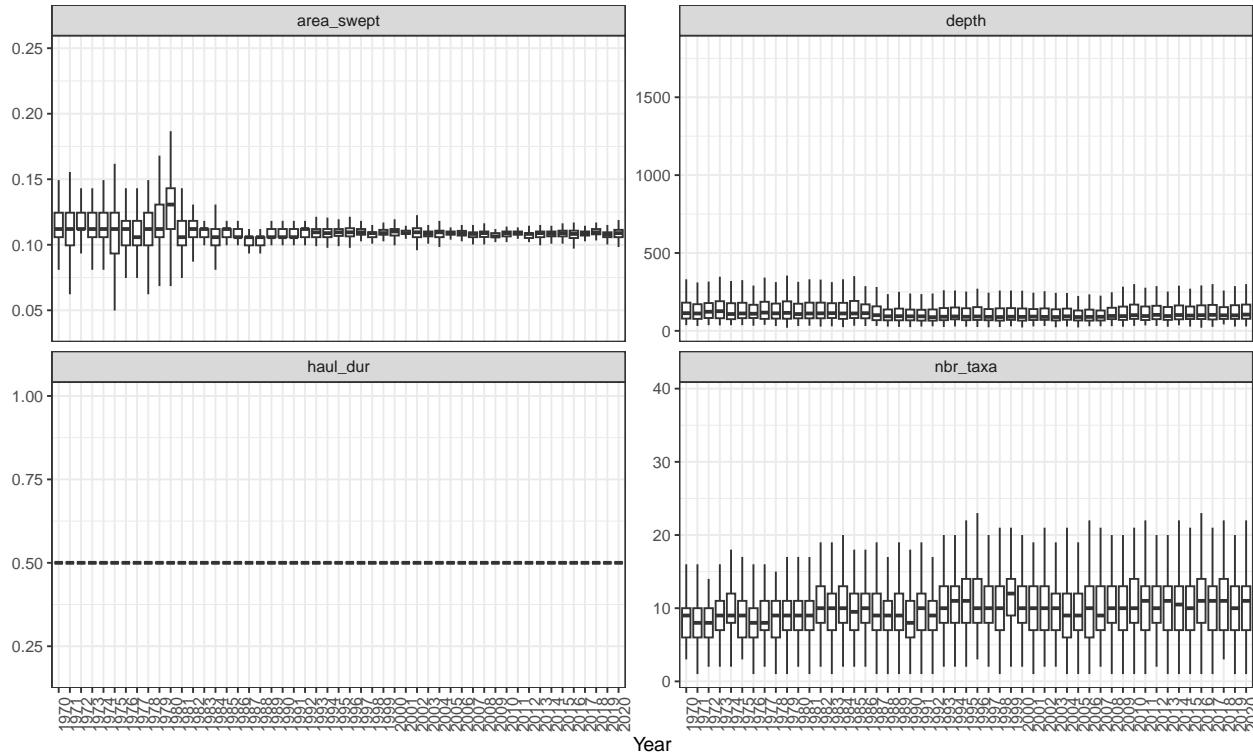
Number of hauls per year performed during the survey after data processing.



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:

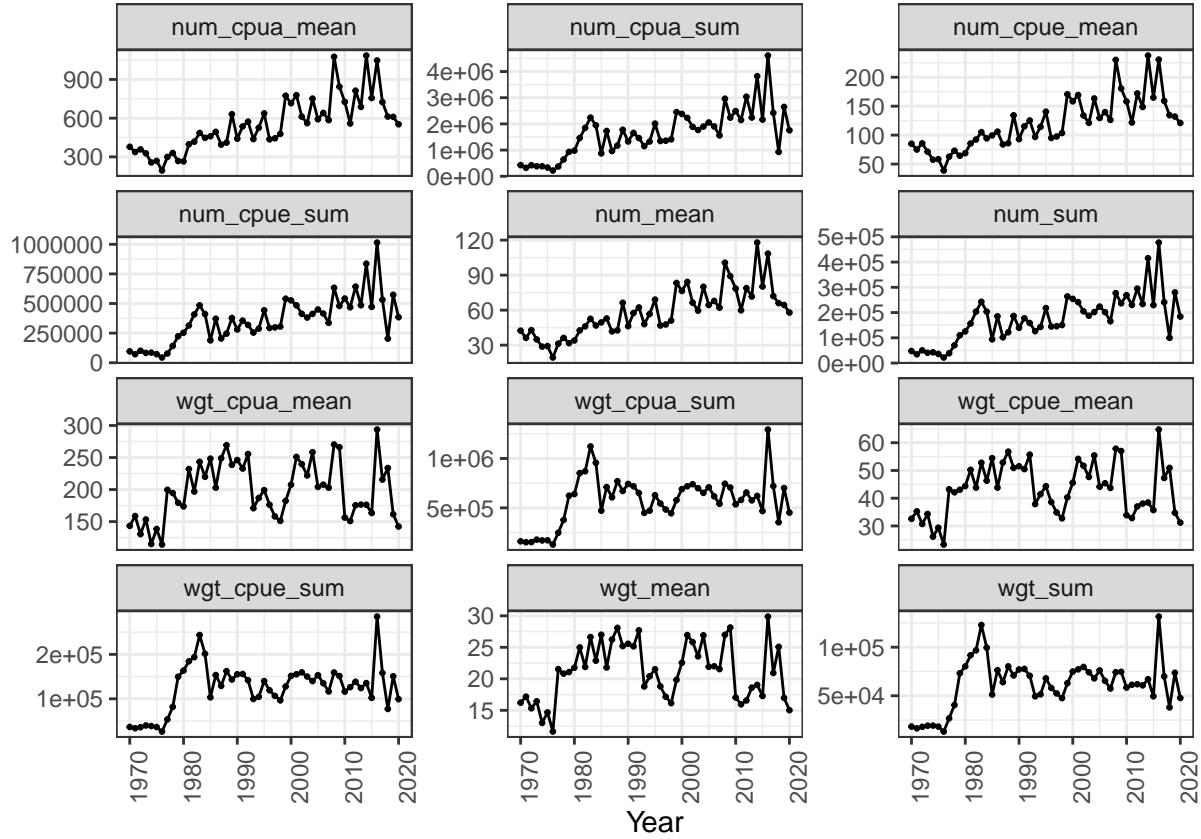
- *area_swept*, swept area by the bottom trawl gear km^2
- *depth*, sampling depth in m
- *haul_dur*, haul sampling duration *hour*
- *number of marine fish taxa*, taxa were cleaned following the last version of taxonomy from the World Register of Marine Species (<https://www.marinespecies.org/>, October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



4. Summary of biological variables

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

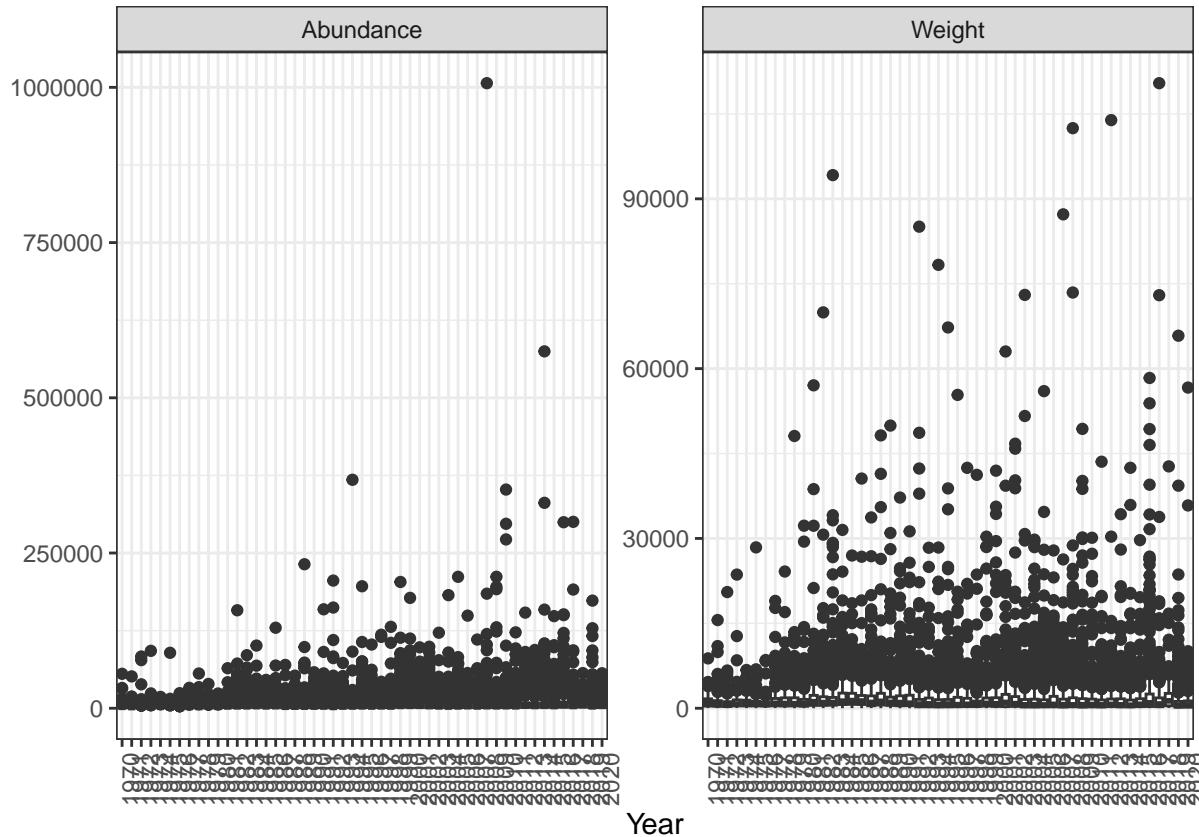
- num_cpua , number of individuals (abundance) in $\frac{individuals}{km^2}$
- num_cpue , number of individuals (abundance) in $\frac{individuals}{h}$
- num , number of individuals (abundance)
- wgt_cpua , weight in $\frac{kg}{km^2}$
- wgt_cpue , weight in $\frac{kg}{h}$
- wgt , weight in kg



5. Extreme values

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

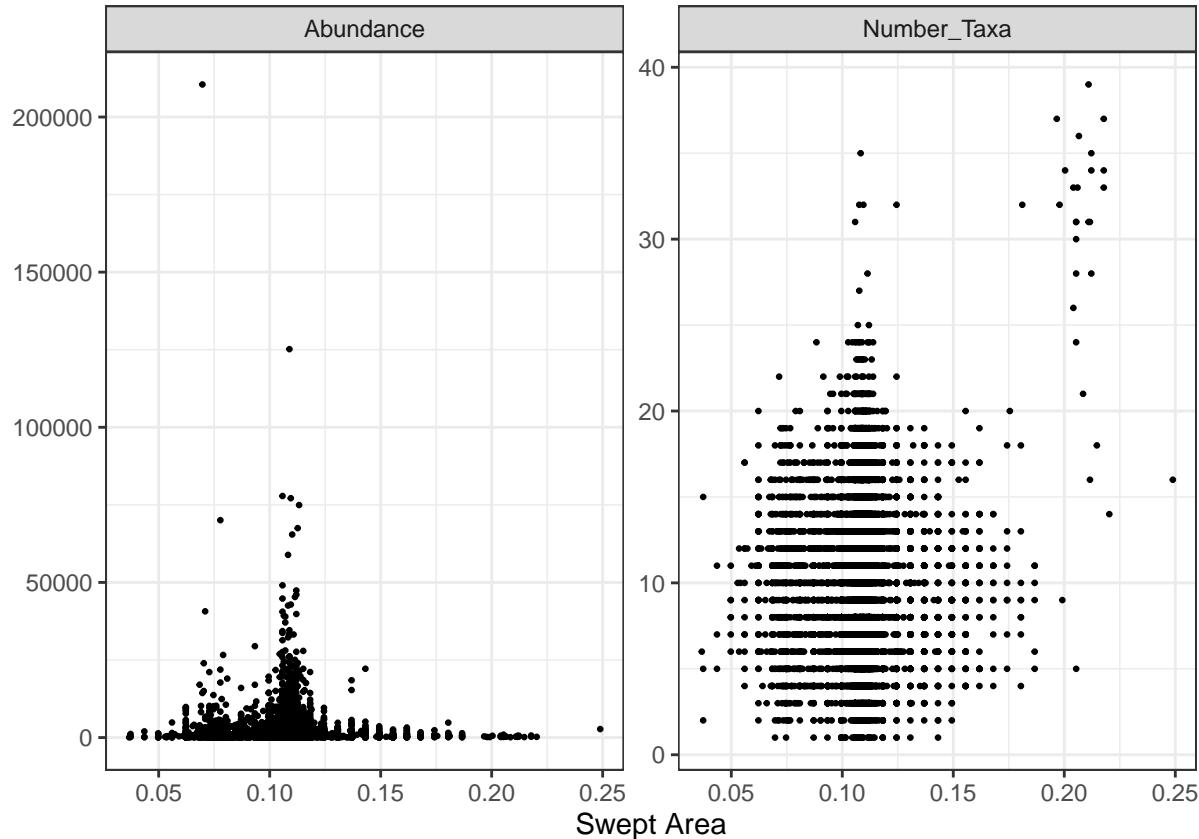
- num_cpue , number of individuals (abundance) in $\frac{individuals}{km^2}$
- wgt_cpue , weight in $\frac{kg}{km^2}$



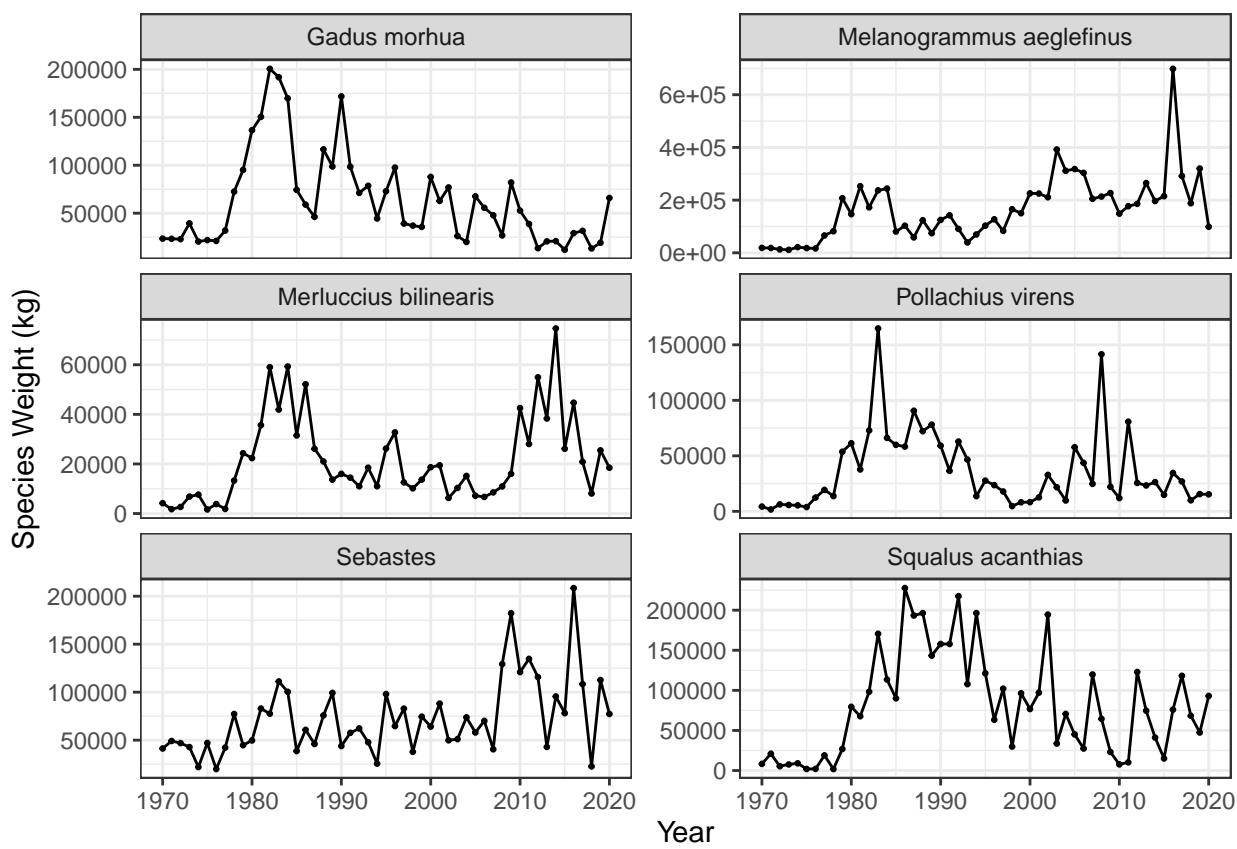
6. Summary of variables against swept area

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

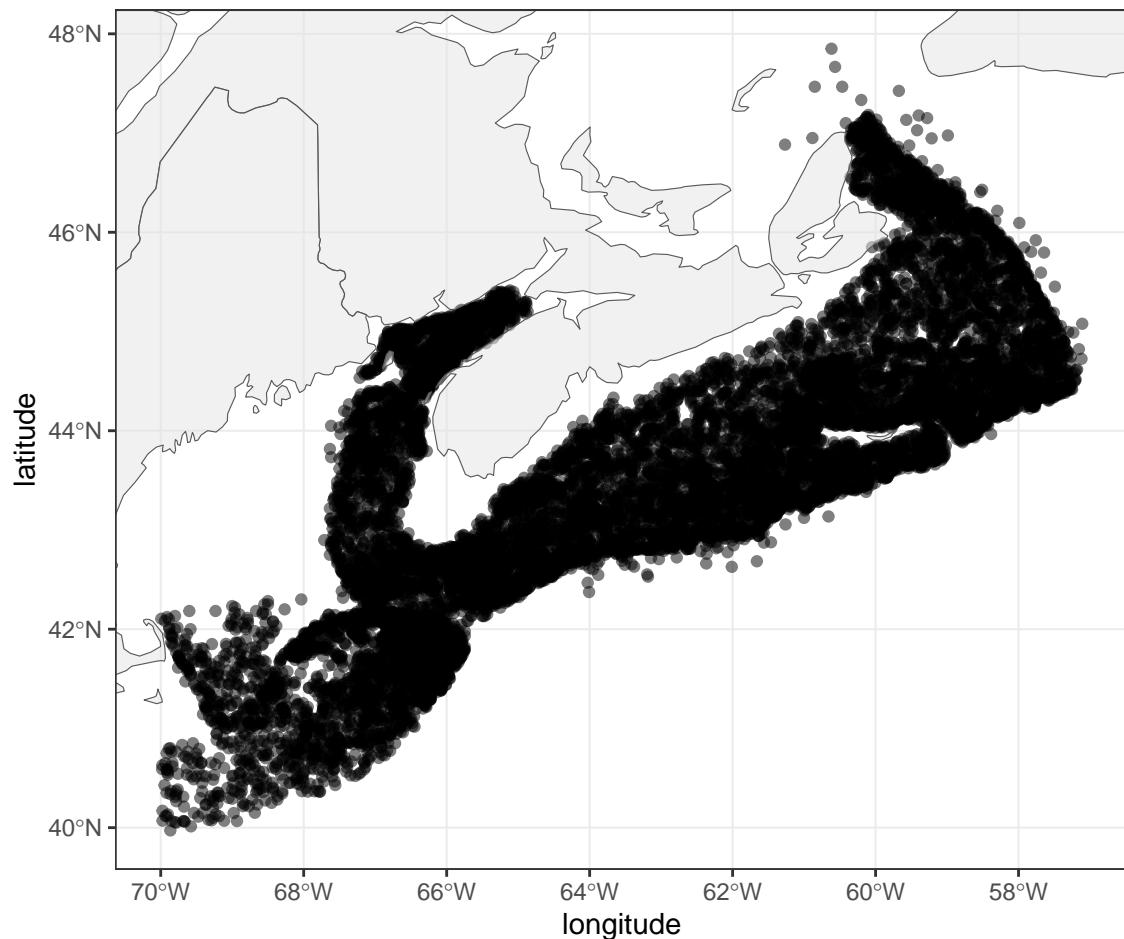


7. Abundance or Weight trends of the six most abundant species



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

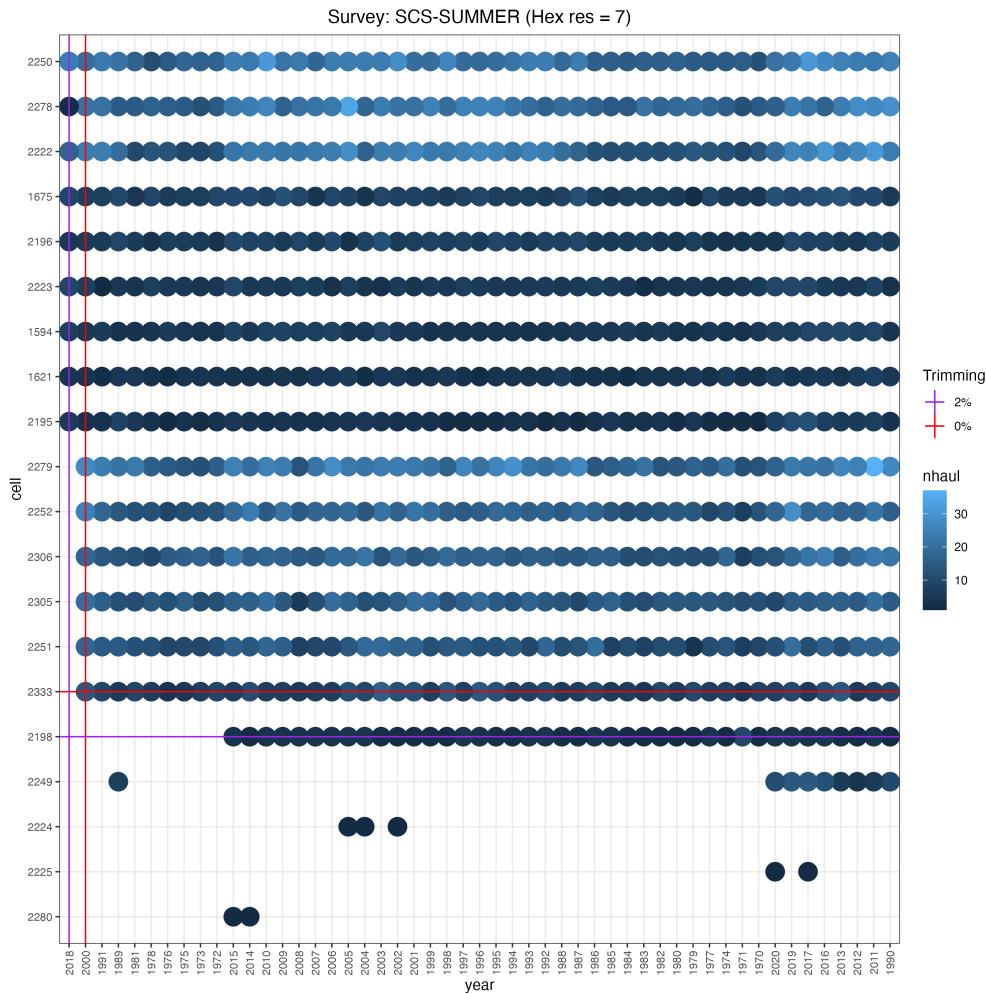
Statistics related to the taxonomic flagging outputs

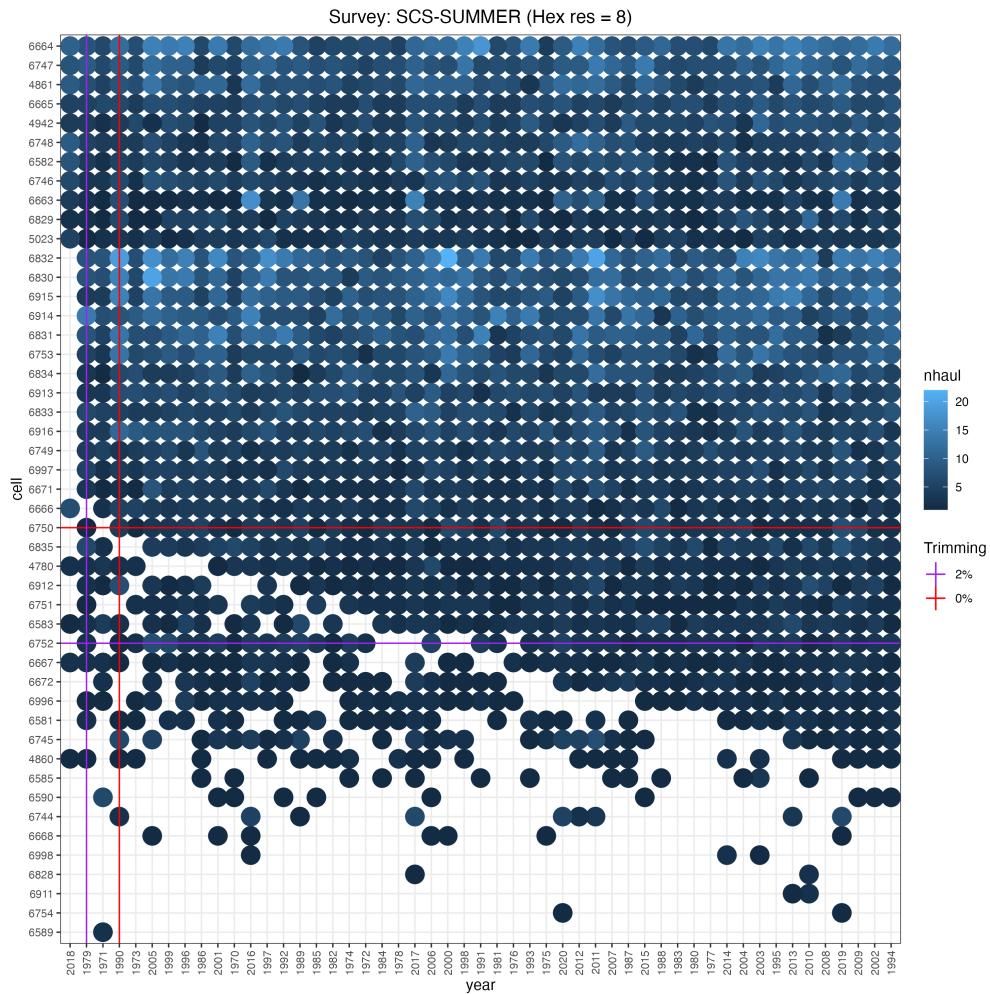
10. Spatio-temporal standardization: SCS-SUMMER

a. Standardization method 1

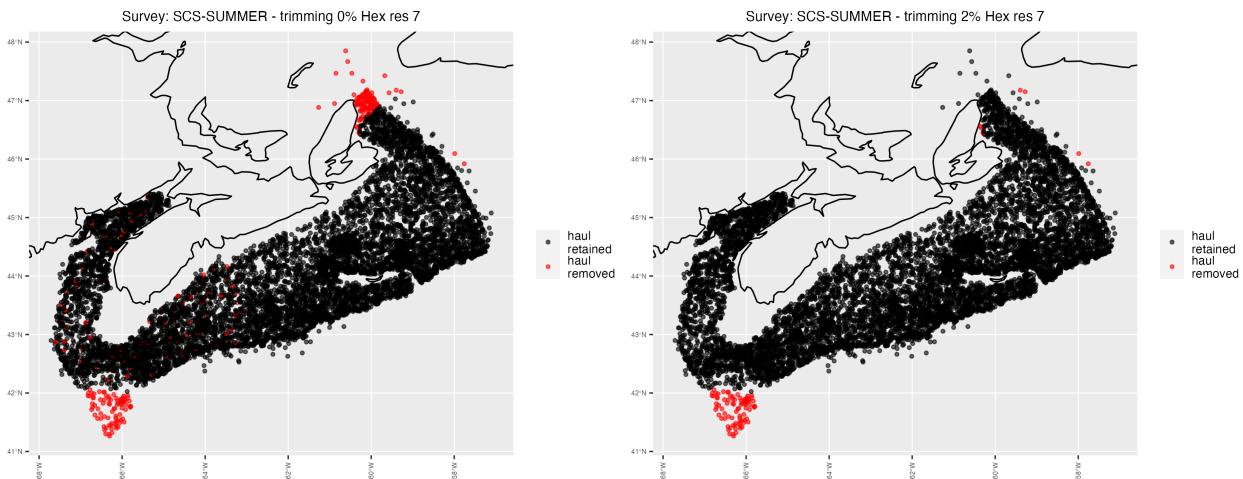
This standardization method was adapted from https://github.com/zoekitchel/trawl_spatial_turnover/blob/master/data_prep_code/species/explore_NorthSea_trimming.Rmd
It was run for hex resolution 7 and 8.

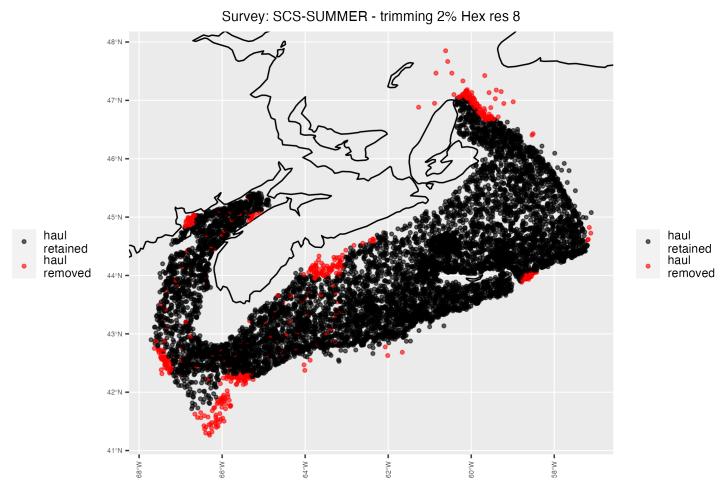
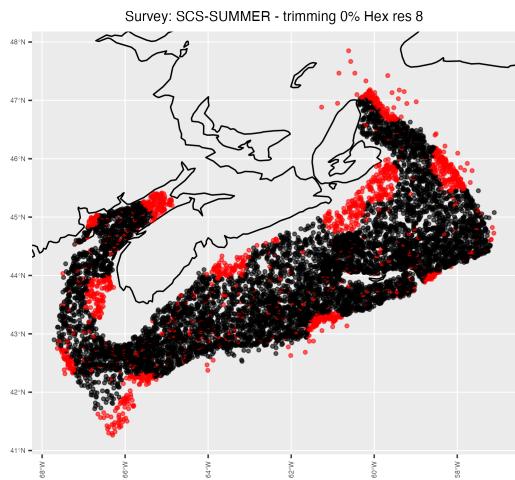
Plot of number of cells x years with overlaid flagging options



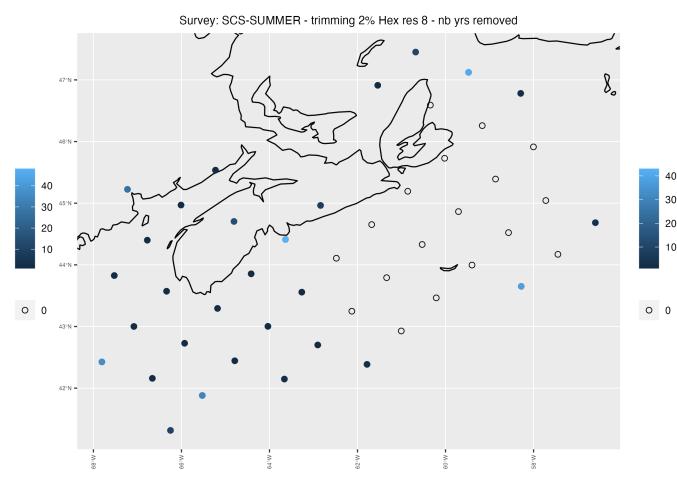
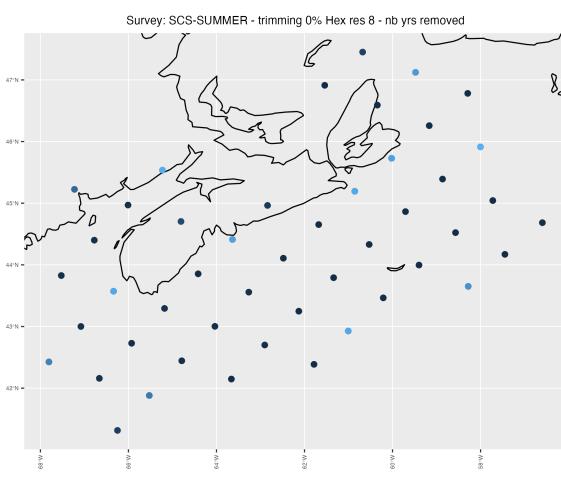
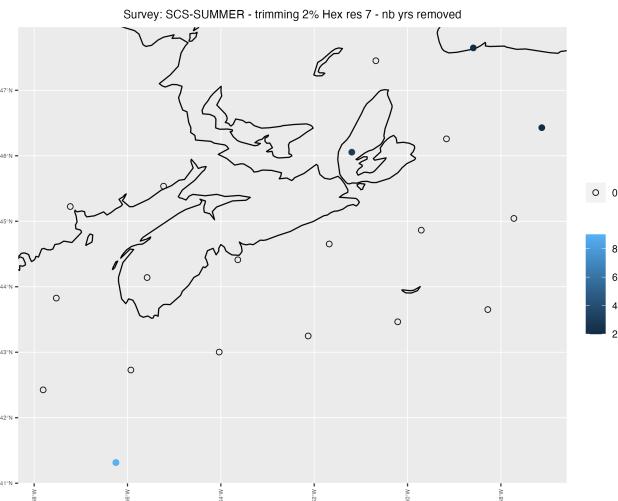
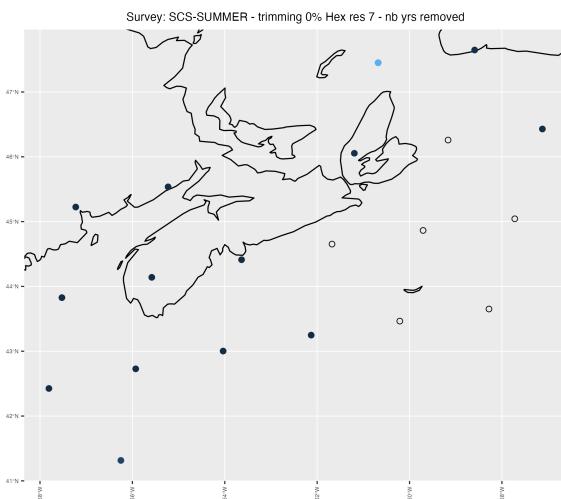


Map of hauls retained and removed per flagging method and threshold





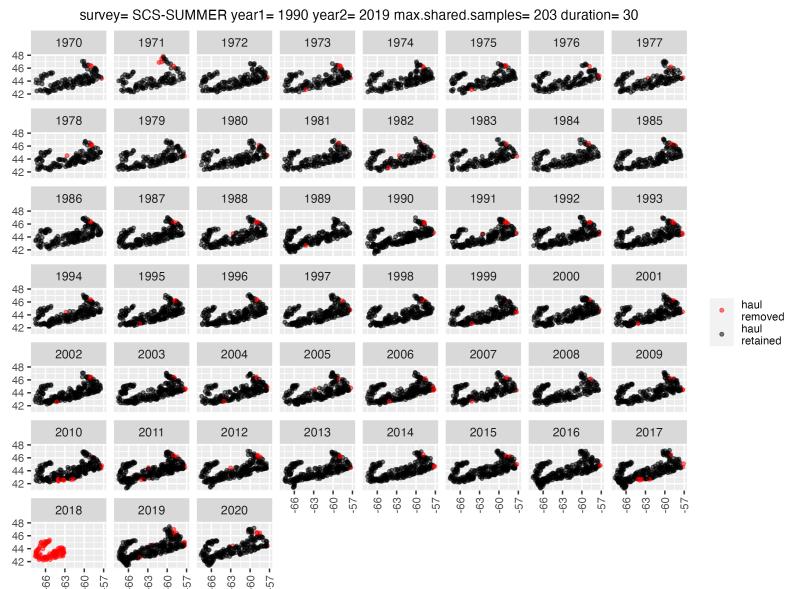
Map of numbers of years removed per grid cell and flagging method/threshold



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

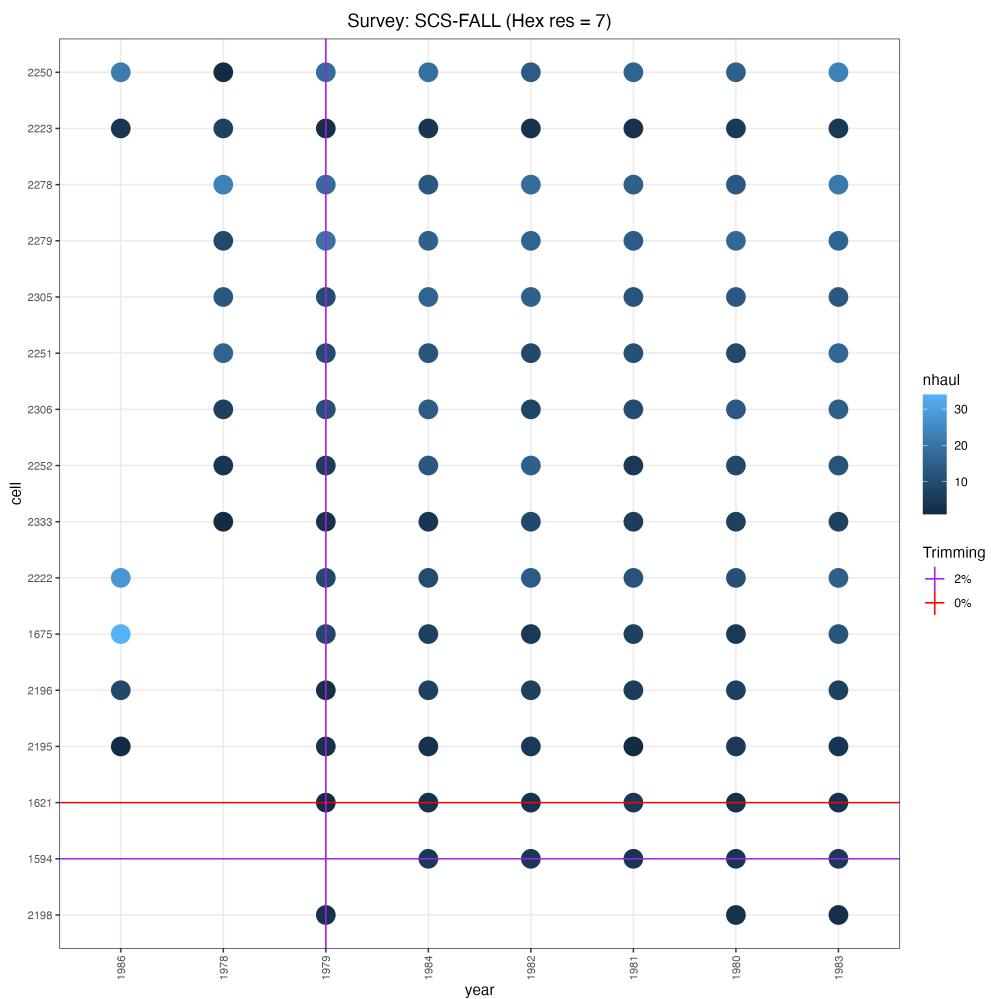
summary	grid cell 7, 0% threshold	grid cell 7, 2% threshold	grid cell 8, 0% threshold	grid cell 8, 2% threshold	method 2 (biotime)
number of hauls removed	287	91	1524.0	529.0	3337.0
percentage of hauls removed	3	1	16.2	5.6	3.6

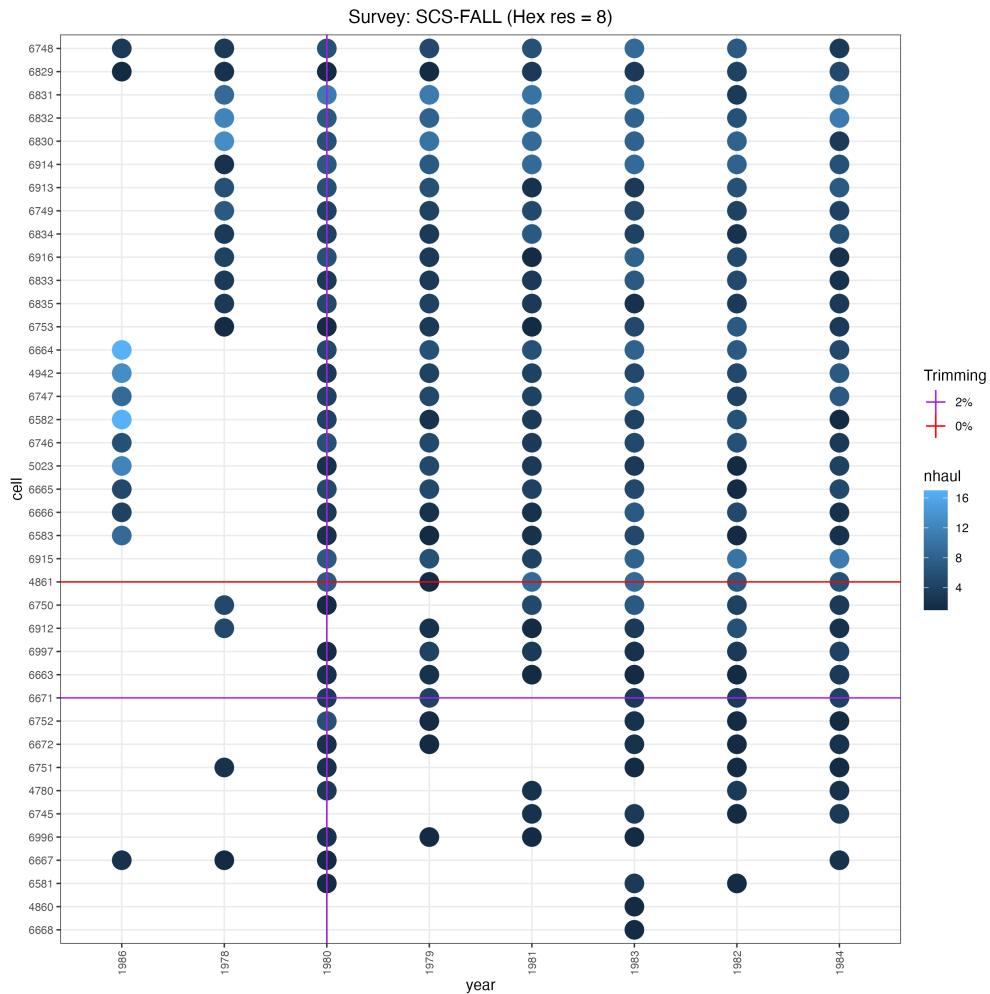
11. Spatio-temporal standardization: SCS-FALL

a. Standardization method 1

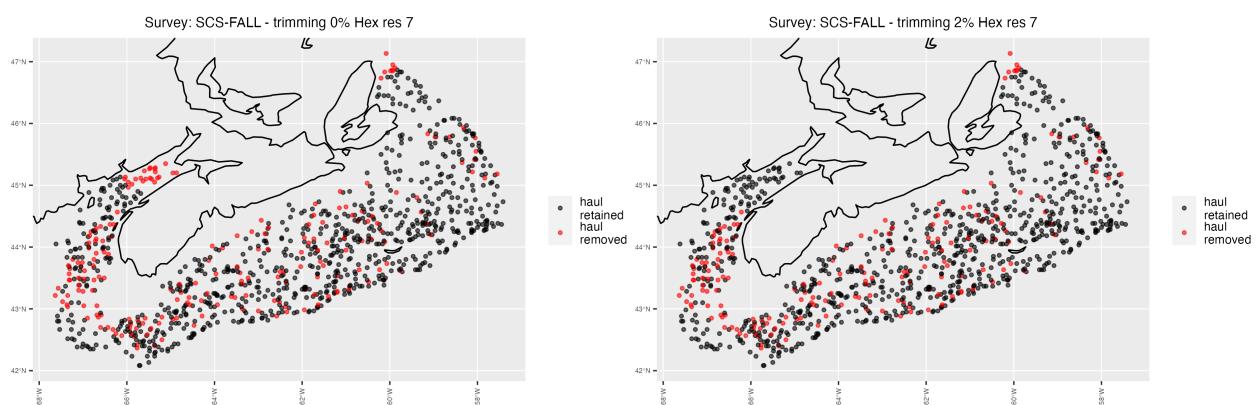
This standardization method was adapted from https://github.com/zoekitchel/trawl_spatial_turnover/blob/master/data_prep_code/species/explore_NorthSea_trimming.Rmd
It was run for hex resolution 7 and 8.

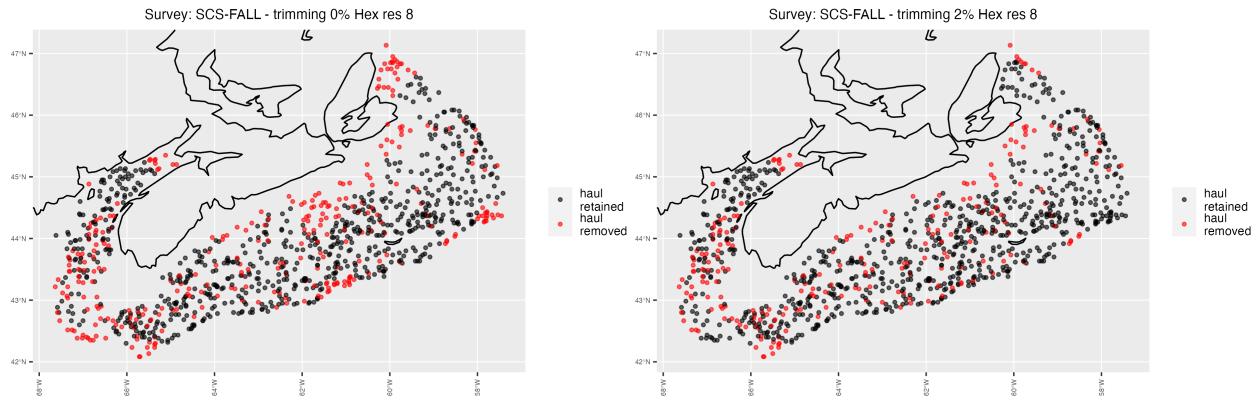
Plot of number of cells x years with overlaid flagging options



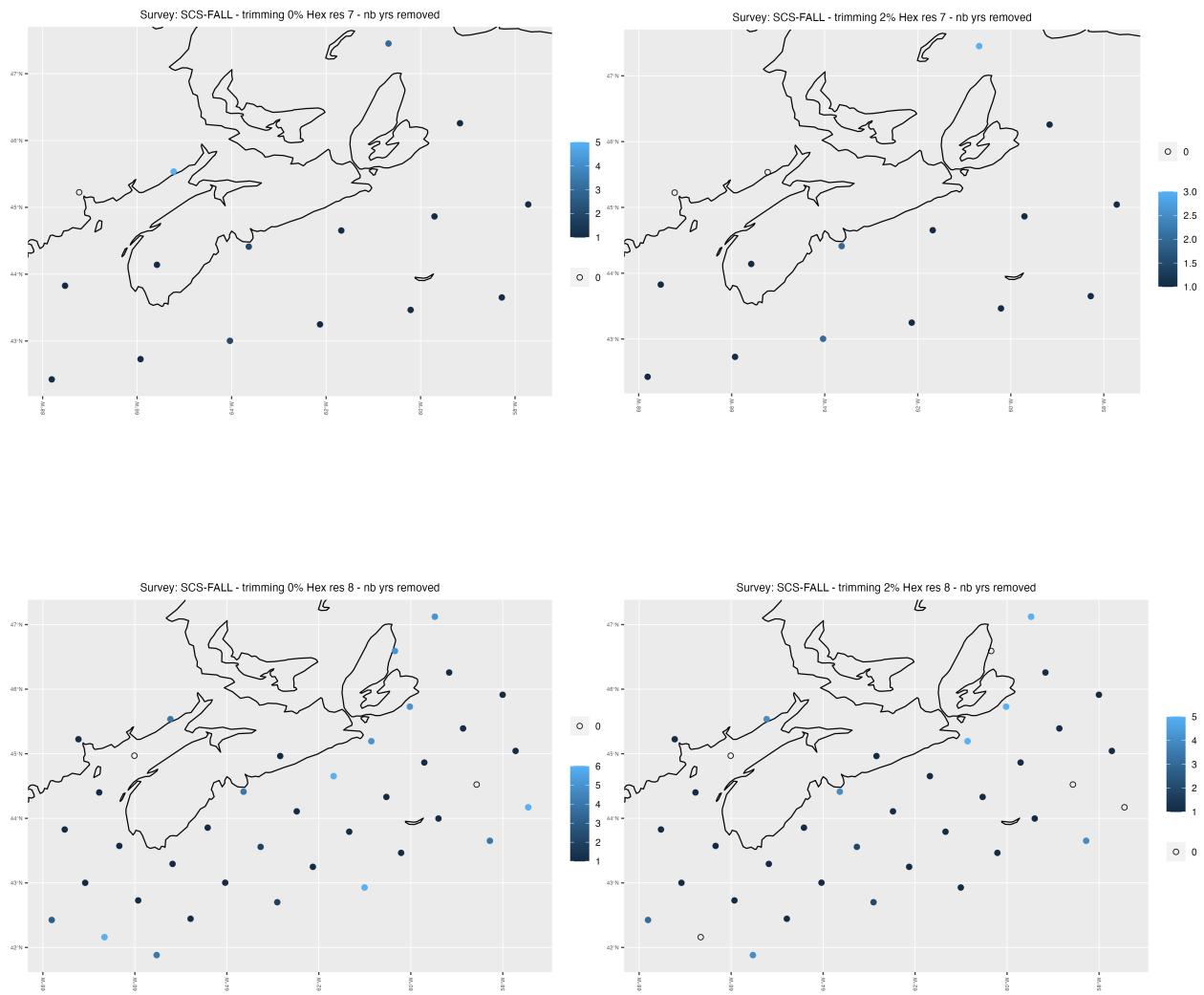


Map of hauls retained and removed per flagging method and threshold





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



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

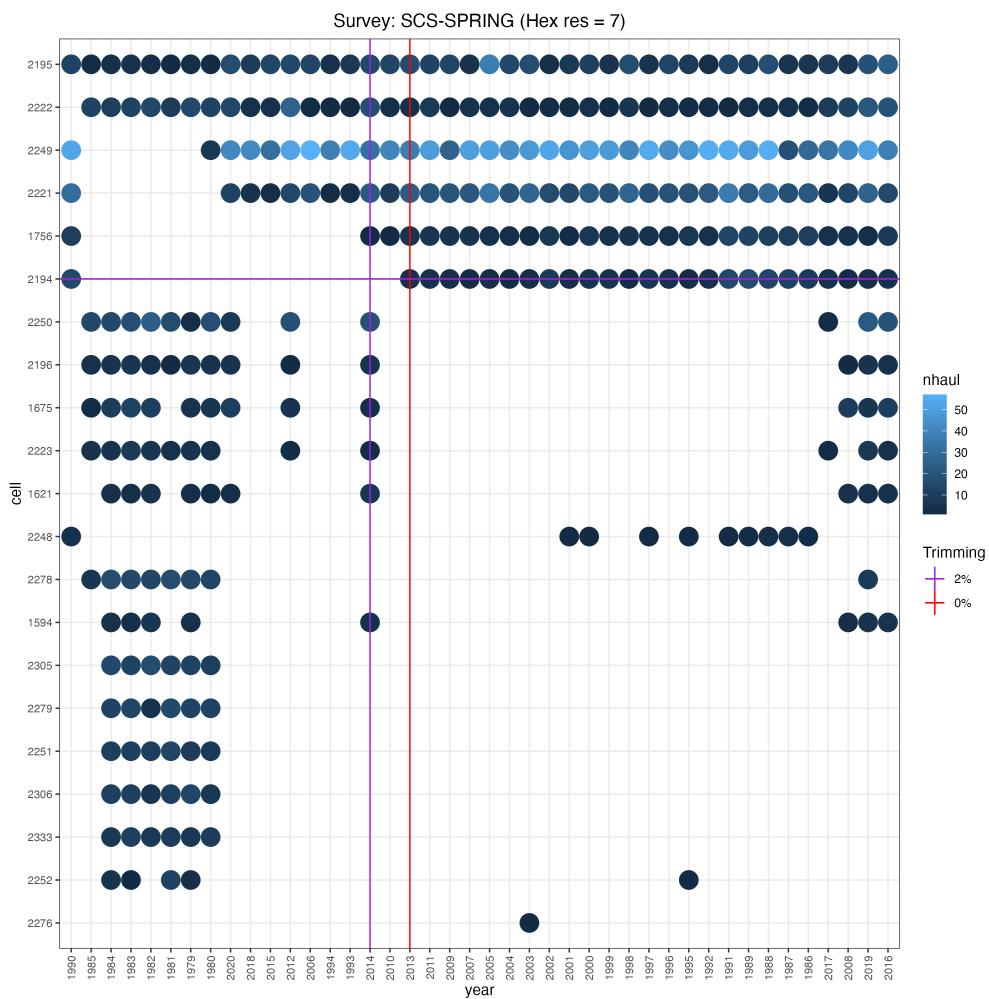
summary	grid cell 7, 0% threshold	grid cell 7, 2% threshold	grid cell 8, 0% threshold	grid cell 8, 2% threshold
number of hauls removed	210.0	187	315.0	237.0
percentage of hauls removed	20.2	18	30.3	22.8

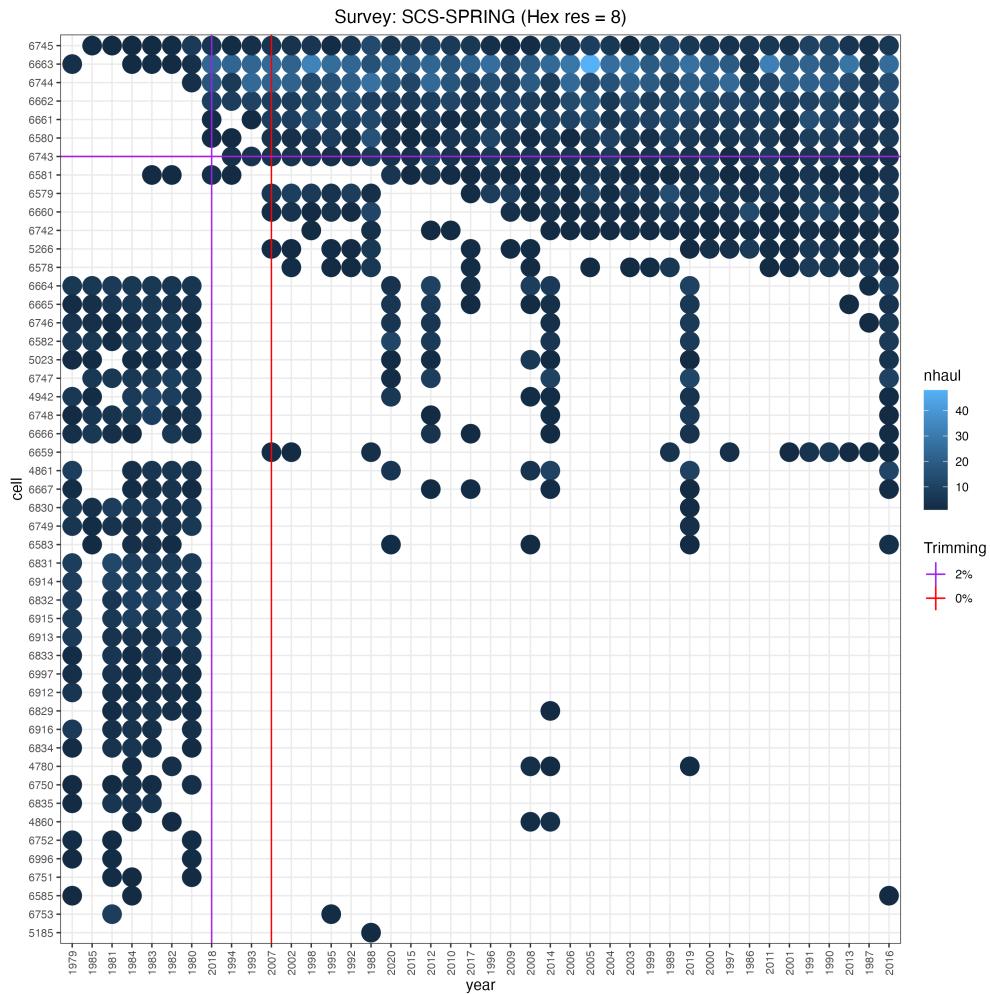
12. Spatio-temporal standardization: SCS-SPRING

a. Standardization method 1

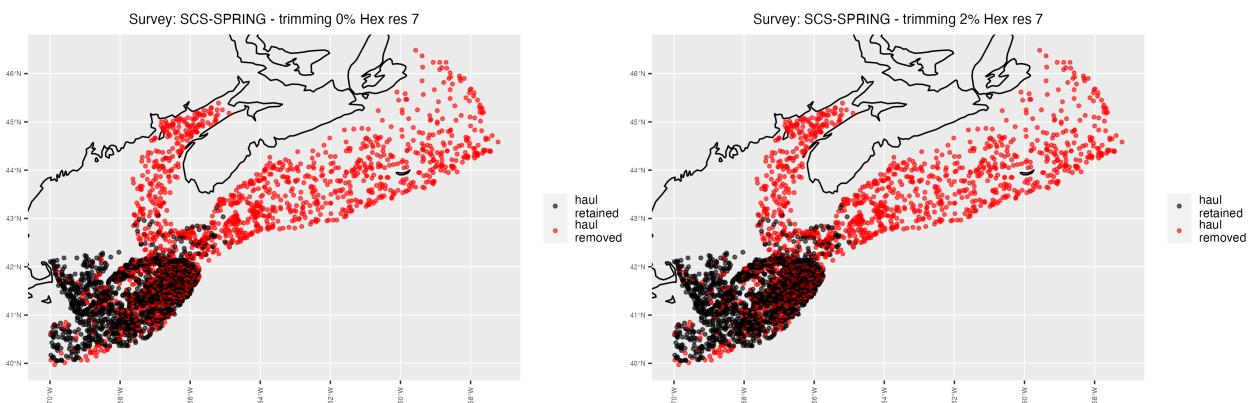
This standardization method was adapted from https://github.com/zookitchel/trawl_spatial_turnover/blob/master/data_prep_code/species/explore_NorthSea_trimming.Rmd
It was run for hex resolution 7 and 8.

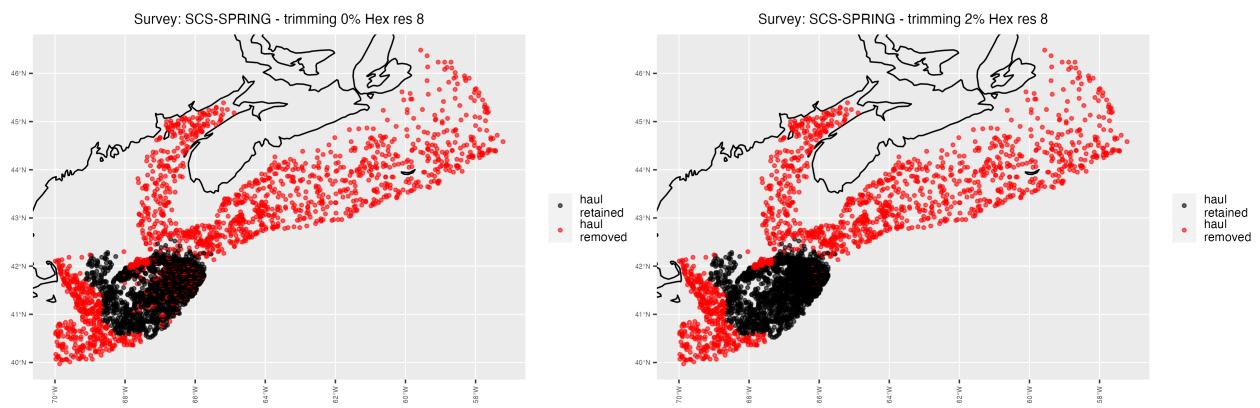
Plot of number of cells x years with overlaid flagging options



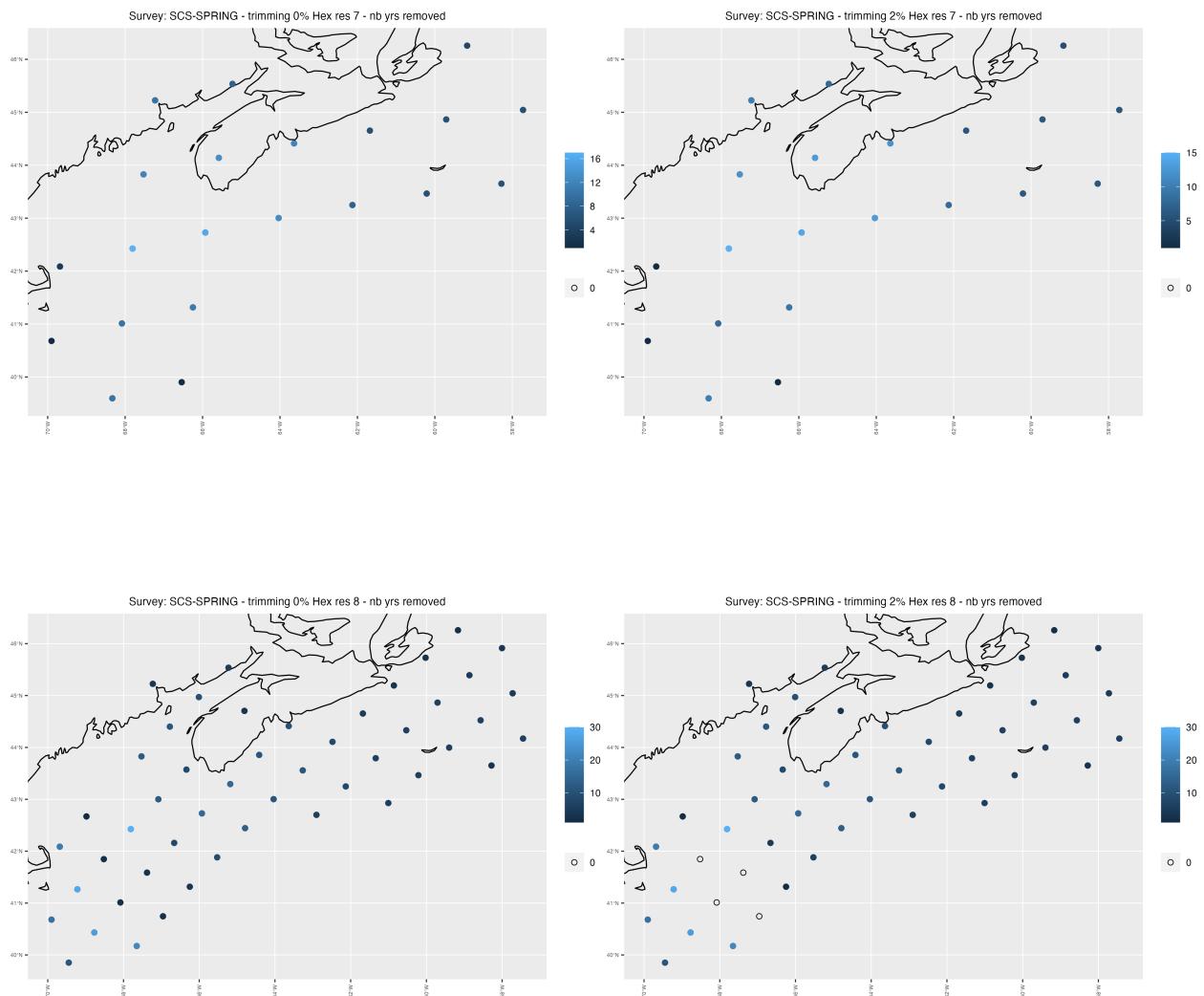


Map of hauls retained and removed per flagging method and threshold





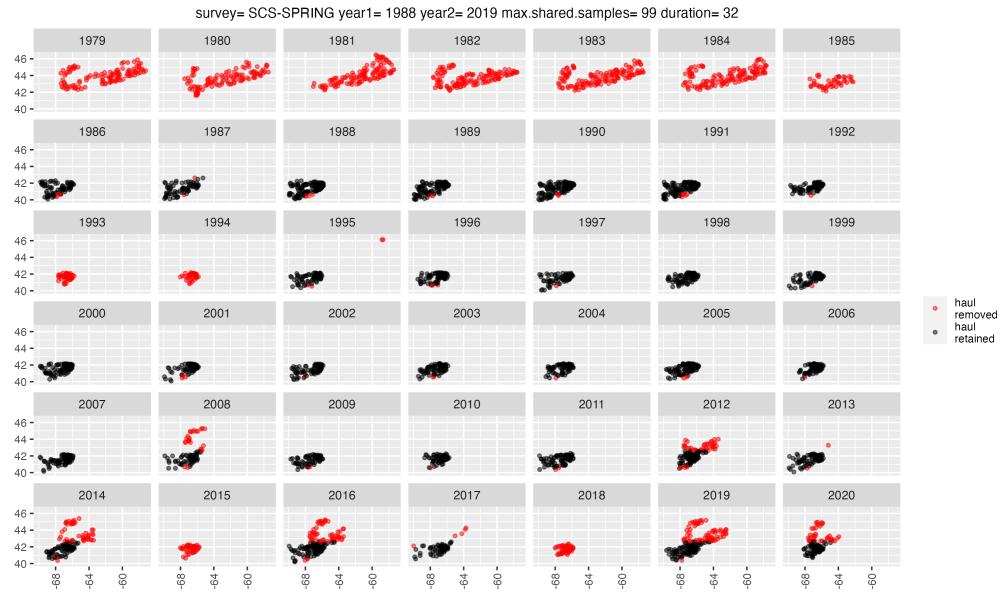
Map of numbers of years removed per grid cell and flagging method/threshold



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	1804.0	1652.0	1786.0	1620.0	14410.0
percentage of hauls removed	43.6	39.9	43.1	39.1	34.6