WCTRI: West Coast US Triennial survey data processing summary

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

January, 2023

Contents

General info

This document presents the cleaning code and summary of the West Coast US Triennial bottom trawl survey provided by Aimee Keller and John Buchanan. It contains data from 1977 and up to 2004.

Data cleaning in R

```
##############################################################################
#### R code to clean trawl survey West Coast US Triennial Survey (WCTRI)
#### Public data Ocean Adapt
#### Contacts:  Aimee Keller    smartt@dnr.sc.gov,  Fisheries Research Surveys Supervisor,
####            NOAA, NMFS, NWFSC, FRAM
#####         John Buchanan john.buchanan@noaa.gov  Fisheries Biologist,
####              Groundfish Ecology Program, Northwest Fisheries Science Center
#### 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 West Coast US can be best accessed using the public Pinsky
#Lab Ocean Adapt Git Hub Repository.
#Contact malin.pinsky@rutgers.edu for questions or help accessing

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

wctri_catch <- read_csv(
  "https://github.com/pinskylab/OceanAdapt/raw/master/data_raw/wctri_catch.csv",
  col_types = cols(
  CRUISEJOIN = col_integer(),
  HAULJOIN = col_integer(),
  CATCHJOIN = col_integer(),
  REGION = col_character(),
  VESSEL = col_integer(),
  CRUISE = col_integer(),
  HAUL = col_integer(),
  SPECIES_CODE = col_integer(),
  WEIGHT = col_double(),
  NUMBER_FISH = col_integer(),
  SUBSAMPLE_CODE = col_character(),
  VOUCHER = col_character(),
  AUDITJOIN = col_integer()
))

wctri_haul <- read_csv(
  "https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/wctri_haul.csv",
  col_types =
                          cols(
                            CRUISEJOIN = col_integer(),
                            HAULJOIN = col_integer(),
                            REGION = col_character(),
                            VESSEL = col_integer(),
                            CRUISE = col_integer(),
                            HAUL = col_integer(),
                            HAUL_TYPE = col_integer(),
                            PERFORMANCE = col_double(),
                            START_TIME = col_character(),
                            DURATION = col_double(),
                            DISTANCE_FISHED = col_double(),
```

```
                          NET_WIDTH = col_double(),
    #Net widths ranged from 9.8 to 17.6 m, with a standard deviation of 0.96 m.
                          NET_MEASURED = col_character(),
                          NET_HEIGHT = col_double(),
                          STRATUM = col_integer(),
                          START_LATITUDE = col_double(),
                          END_LATITUDE = col_double(),
                          START_LONGITUDE = col_double(),
                          END_LONGITUDE = col_double(),
                          STATIONID = col_character(),
                          GEAR_DEPTH = col_integer(),
                          BOTTOM_DEPTH = col_integer(),
                          BOTTOM_TYPE = col_integer(),
                          SURFACE_TEMPERATURE = col_double(),
                          GEAR_TEMPERATURE = col_double(),
                          WIRE_LENGTH = col_integer(),
                          GEAR = col_integer(),
                          ACCESSORIES = col_integer(),
                          SUBSAMPLE = col_integer(),
                          AUDITJOIN = col_integer()
                        ))

wctri_species <- read_csv(
"https://raw.githubusercontent.com/pinskylab/OceanAdapt/master/data_raw/wctri_species.csv",
  col_types = cols(
  SPECIES_CODE = col_integer(),
  SPECIES_NAME = col_character(),
  COMMON_NAME = col_character(),
  REVISION = col_character(),
  BS = col_character(),
  GOA = col_character(),
  WC = col_character(),
  AUDITJOIN = col_integer()
))


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


# Add haul info to catch data
wctri <- left_join(wctri_catch, wctri_haul, by = c(
                          "CRUISEJOIN", "HAULJOIN", "VESSEL", "CRUISE", "HAUL"))
#  add species names
wctri <- left_join(wctri, wctri_species, by = "SPECIES_CODE")


wctri <- wctri %>%
  # trim to standard hauls and good performance (applicable to fishglob too)
  filter(HAUL_TYPE == 3 & PERFORMANCE == 0) %>%
  # Create a unique haul_id
  mutate(
```

```
    haul_id = paste(formatC(VESSEL, width=3, flag=0), CRUISE,
                    formatC(HAUL, width=3, flag=0), START_LONGITUDE,
                    START_LATITUDE, sep=''),
    # Extract year where needed
    year = substr(CRUISE, 1, 4),
    month = substr(CRUISE, 5,6),
    day = NA,
    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_character_,
    # Add "strata" (define by lat, lon and depth bands) where needed # degree bins
    # 100 m bins # no need to use lon grids on west coast (so narrow)
    stratum = paste(floor(START_LATITUDE)+0.5, floor(BOTTOM_DEPTH/100)*100 + 50, sep= "-"),
    area_swept = DISTANCE_FISHED*(NET_WIDTH/1000), #distanced_fished in km *
    #net_width in meters / 1000 m/km
    # adjust for tow area # weight per km (1000 m2)
    wgt_cpue = WEIGHT/area_swept, #kg/km^2
    wgt_h = WEIGHT/DURATION, #kg/hour
    num_h = NUMBER_FISH/DURATION, # ind/hour
    num_cpue = NUMBER_FISH/area_swept #ind/km2
  )

wctri <- wctri %>%
  rename(
    haul_dur = DURATION, #DURATION is in hours
    svvessel = VESSEL,
    latitude = START_LATITUDE,
    longitude = START_LONGITUDE,
    depth = BOTTOM_DEPTH,
    spp = SPECIES_NAME,
    sst = SURFACE_TEMPERATURE,
    num = NUMBER_FISH,
    gear = GEAR,
    station = STATIONID,
    verbatim_name = SPECIES_NAME,
    wgt = WEIGHT
  ) %>%
  filter(
    verbatim_name != "" &
      !grepl("egg", verbatim_name)
  ) %>%
  # adjust spp names
  mutate(verbatim_name = ifelse(grepl("Lepidopsetta", verbatim_name),
                                "Lepidopsetta sp.", verbatim_name),
         verbatim_name = ifelse(grepl("Bathyraja", verbatim_name),
                                'Bathyraja sp.', verbatim_name),
         verbatim_name = ifelse(grepl("Squalus", verbatim_name),
                                'Squalus suckleyi', verbatim_name),
         sbt = NA,) %>%
  # add survey column
  mutate(survey = "WCTRI",
```

```
         source = "NOAA",
         timestamp = mdy("02/06/2019"),
         country = "United States",
         continent = "n_america",
         sub_area = NA,
         stat_rec = NA) %>%
  select(survey, haul_id, source, timestamp,
         country, sub_area, continent, stat_rec, station,
         stratum, year, month, day, quarter, season, latitude, longitude,
         haul_dur, area_swept, gear, depth, sbt, sst,
         num, num_h, num_cpue, wgt, wgt_h, wgt_cpue, verbatim_name)

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

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

#none!

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

unique_name_match
#empty

#the following verbatim names are duplicated within haul_id without fix above
#Bathyraja sp.
#Actinauge verrilli
#Rossia pacifica
#Sebastes alutus


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

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

### Automatic cleaning
# Set Survey code
wctri_survey_code <- "WCTRI"

wctri <- wctri %>%
  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(wctri$taxa2), input_survey = wctri_survey_code,
                         fishbase=T)
# takes 20 mins!

#This cuts out the following species which are all inverts

#Cheiraster dawsoni
#Crangon communis
#Cancer gracilis
#Cancer anthonyi

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

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

clean_wctri <- left_join(wctri, clean_taxa, by=c("taxa2"="query")) %>%
  filter(!is.na(taxa)) %>% # query does not indicate taxa entry that were removed in the cleaning proce
  # 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_wctri <- clean_wctri %>%
  group_by(haul_id, accepted_name) %>%
```

```r
  mutate(count = n())

#none!

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

unique_name_match
#check if empty


#### ------------------------------------------------------------------------------------------ #
# Save database in Google drive
#### ------------------------------------------------------------------------------------------ #

# Just run this routine should be good for all
write_clean_data(data = clean_wctri, survey = "WCTRI", 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)
```

```r
#load pipe operator
library(magrittr)

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

#run flag_spp function in a loop
for (r in regions) {
  flag_spp(clean_wctri, 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_wctri, 7)

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

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


#-----------------------------------------------------------------------------------------------#
#### ADD STRANDARDIZATION FLAGS ####
#-----------------------------------------------------------------------------------------------#
surveys <- sort(unique(clean_wctri$survey))
survey_units <- sort(unique(clean_wctri$survey_unit))
survey_std <- clean_wctri %>%
  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)
  }
}
```

```r
# 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 = "WCTRI_std",
                 overwrite = T, rdata=TRUE)
```

1. Overview of the survey data table

survey	source	timestamp	haul_id	country	sub_area
WCTRI	NOAA	2019-02-06	004197701002-119.3334.09	United States	NA
WCTRI	NOAA	2019-02-06	004197701002-119.3334.09	United States	NA
WCTRI	NOAA	2019-02-06	004197701002-119.3334.09	United States	NA
WCTRI	NOAA	2019-02-06	004197701002-119.3334.09	United States	NA
WCTRI	NOAA	2019-02-06	004197701002-119.3334.09	United States	NA

continent	stat_rec	station	stratum	year	month	day	quarter	season
n_america	NA	NA	34.5-250	1977	01	NA	NA	NA
n_america	NA	NA	34.5-250	1977	01	NA	NA	NA
n_america	NA	NA	34.5-250	1977	01	NA	NA	NA
n_america	NA	NA	34.5-250	1977	01	NA	NA	NA
n_america	NA	NA	34.5-250	1977	01	NA	NA	NA

latitude	longitude	haul_dur	area_swept	gear	depth	sbt	sst
34.09	-119.33	0.5	0.032256	160	254	NA	16.2
34.09	-119.33	0.5	0.032256	160	254	NA	16.2
34.09	-119.33	0.5	0.032256	160	254	NA	16.2
34.09	-119.33	0.5	0.032256	160	254	NA	16.2
34.09	-119.33	0.5	0.032256	160	254	NA	16.2

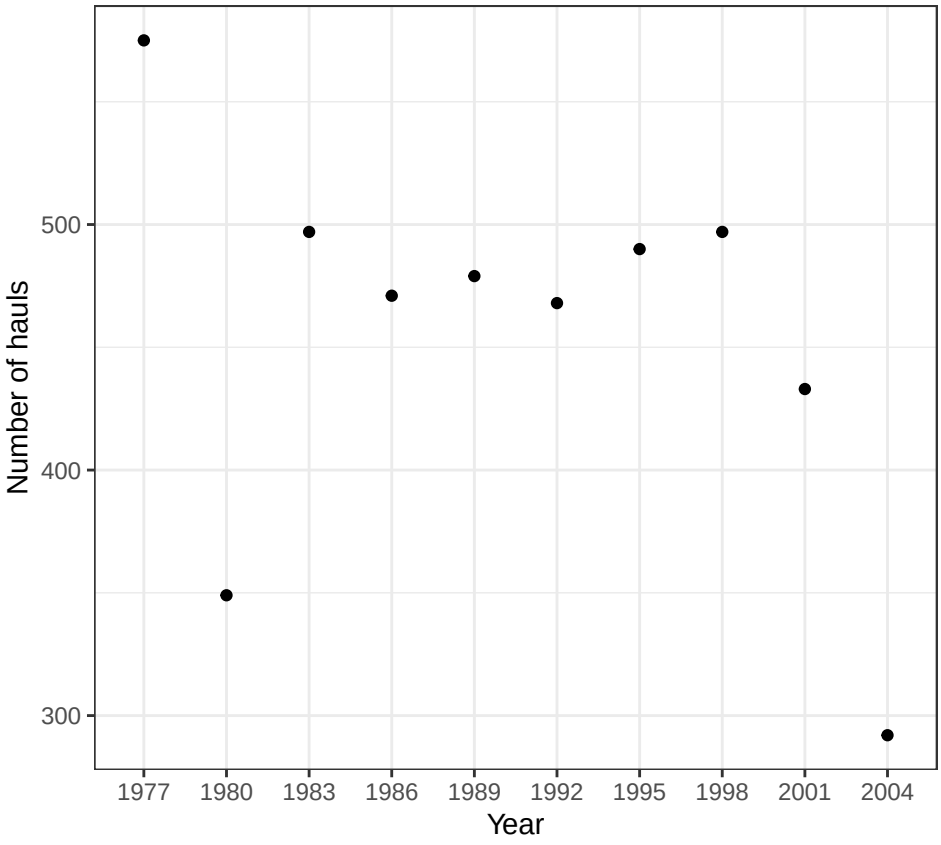
num	num_cpue	num_cpua	wgt	wgt_cpue	wgt_cpua	verbatim_name
2	4	62.00397	1.81	3.62	56.113591	Eopsetta jordani
1	2	31.00198	0.45	0.90	13.950893	Hippoglossina stomata
2	4	62.00397	0.13	0.26	4.030258	Microstomus pacificus
9	18	279.01786	2.72	5.44	84.325397	Parophrys vetulus
1	2	31.00198	0.45	0.90	13.950893	Pleuronichthys verticalis

verbatim_aphia_id	accepted_name	aphia_id	SpecCode	kingdom
NA	Eopsetta jordani	280690	4237	Animalia
NA	Hippoglossina stomata	275827	4225	Animalia
NA	Microstomus pacificus	274294	4247	Animalia
NA	Parophrys vetulus	254393	4248	Animalia
NA	Pleuronichthys verticalis	282295	4254	Animalia

phylum	class	order	family	genus	rank	survey_unit
Chordata	Teleostei	Pleuronectiformes	Pleuronectidae	Eopsetta	Species	WCTRI
Chordata	Teleostei	Pleuronectiformes	Paralichthyidae	Hippoglossina	Species	WCTRI
Chordata	Teleostei	Pleuronectiformes	Pleuronectidae	Microstomus	Species	WCTRI
Chordata	Teleostei	Pleuronectiformes	Pleuronectidae	Parophrys	Species	WCTRI
Chordata	Teleostei	Pleuronectiformes	Pleuronectidae	Pleuronichthys	Species	WCTRI

2. Summary of sampling intensity

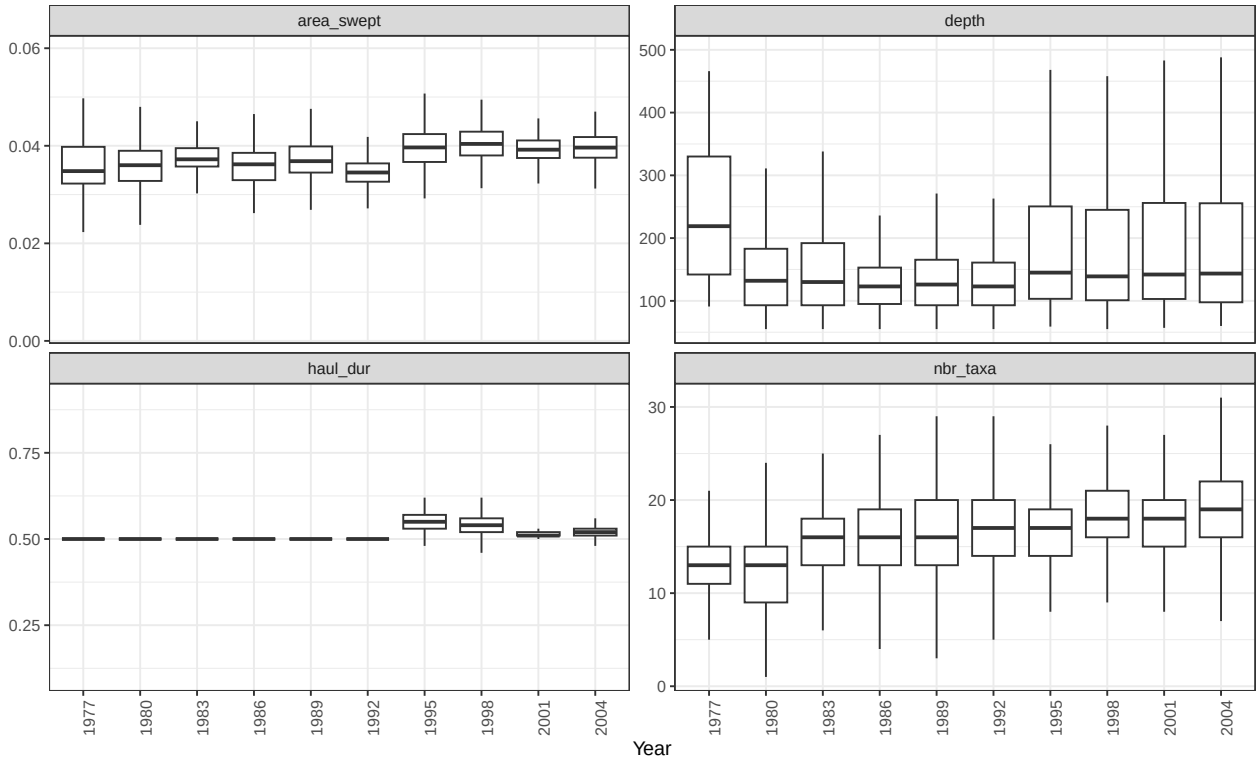
Number of hauls per year performed during the survey after data processing.



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:

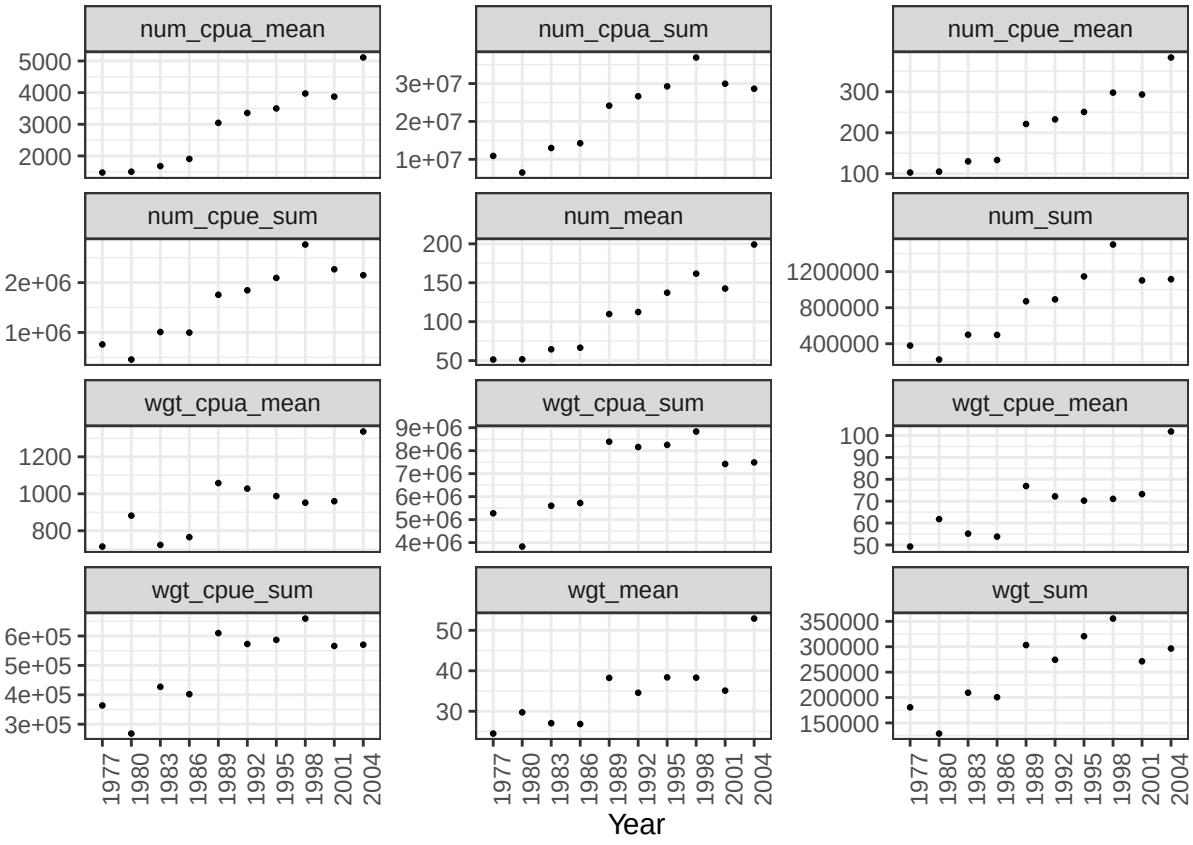
- *area_swept*, swept area by the bottom trawl gear km^2
- *depth*, sampling depth in m
- *haul_dur*, haul sampling duration *hour*
- *number of marine fish taxa*, taxa were cleaned following the last version of taxonomy from the World Register of Marine Species (https://www.marinespecies.org/, October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



4. Summary of biological variables

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

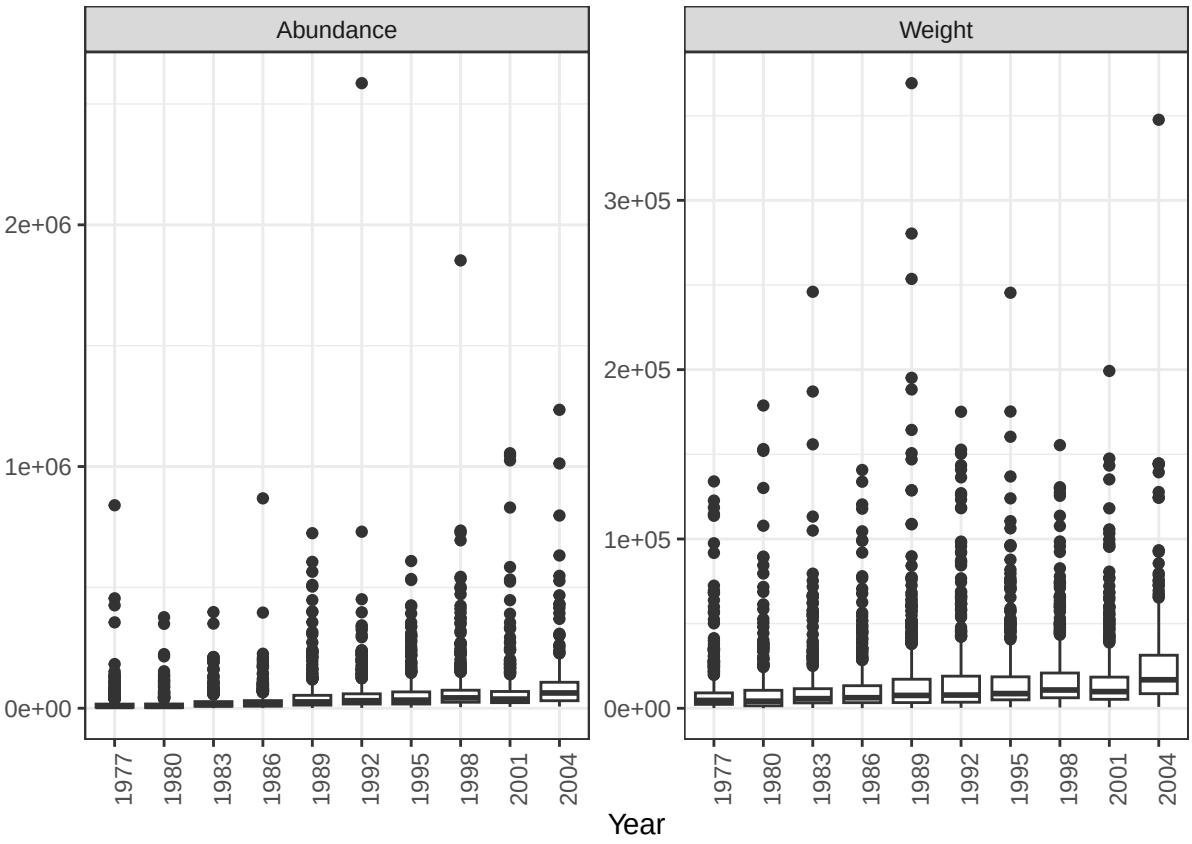
- *num_cpua*, number of individuals (abundance) in $\frac{individuals}{km^2}$
- *num_cpue*, number of individuals (abundance) in $\frac{individuals}{h}$
- *num*, number of individuals (abundance)
- *wgt_cpua*, weight in $\frac{kg}{km^2}$
- *wgt_cpue*, weight in $\frac{kg}{h}$
- *wgt*, weight in kg



5. Extreme values

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

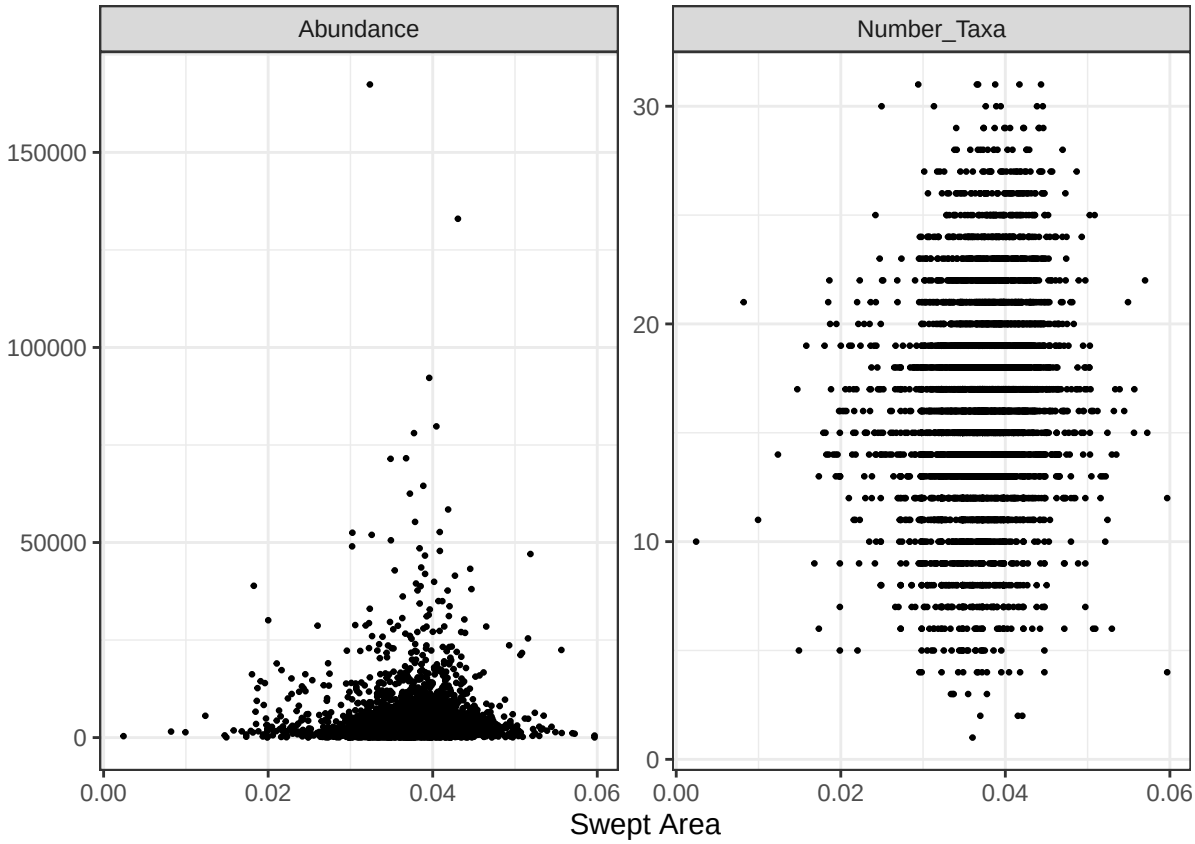
- *num_cpue*, number of individuals (abundance) in $\frac{individuals}{km^2}$
- *wgt_cpue*, weight in $\frac{kg}{km^2}$



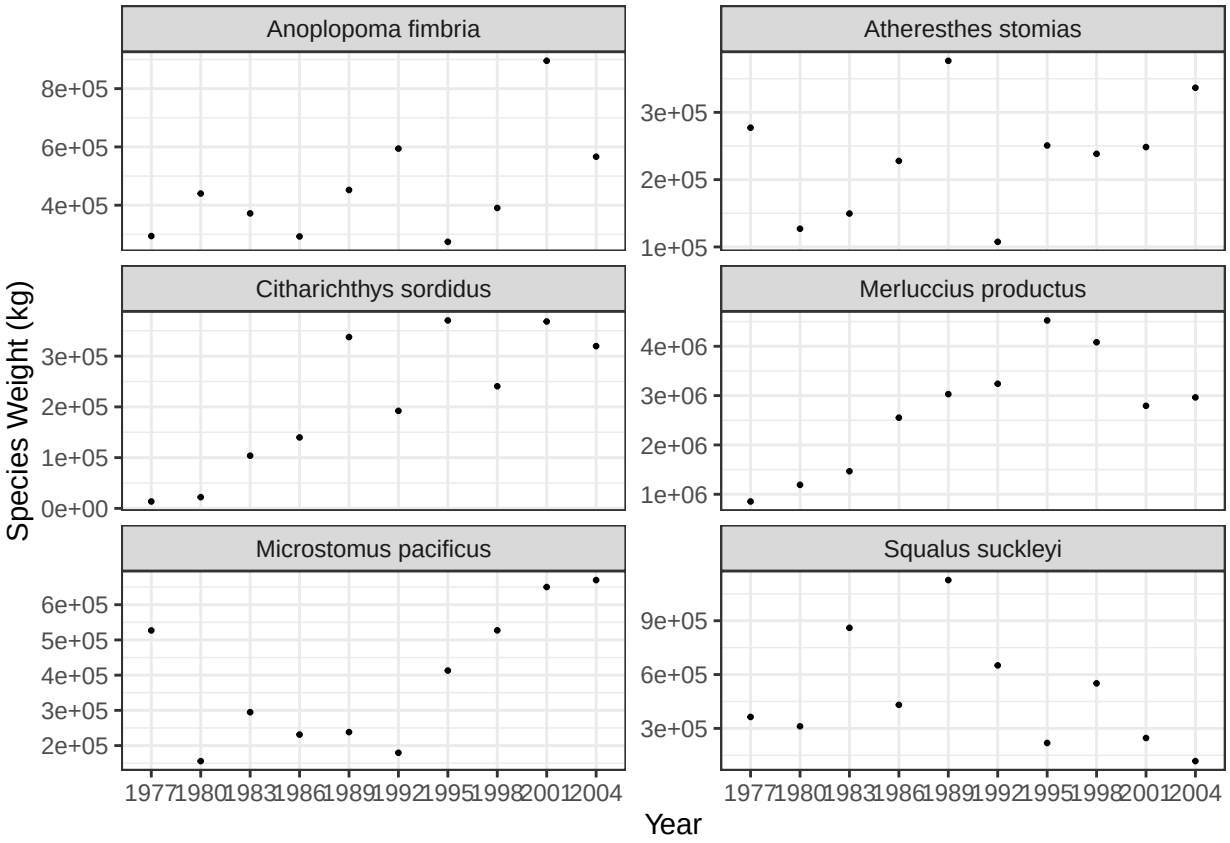
6. Summary of variables against swept area

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

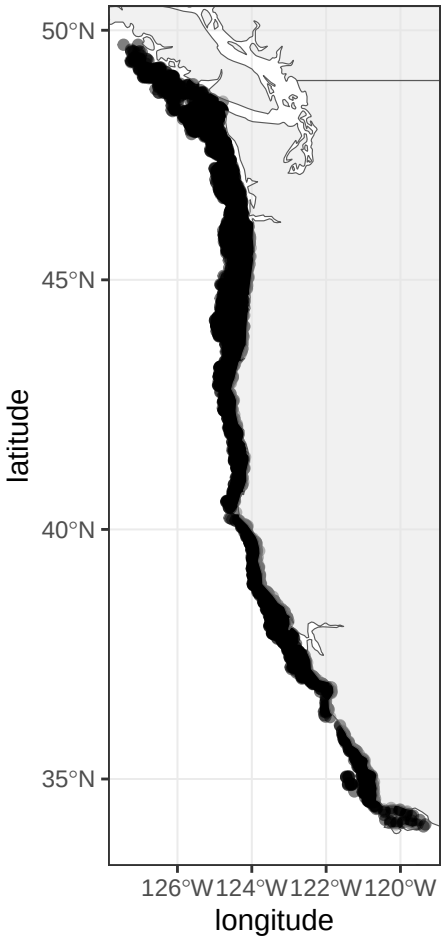


7. Abundance or Weight trends of the six most abundant species



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

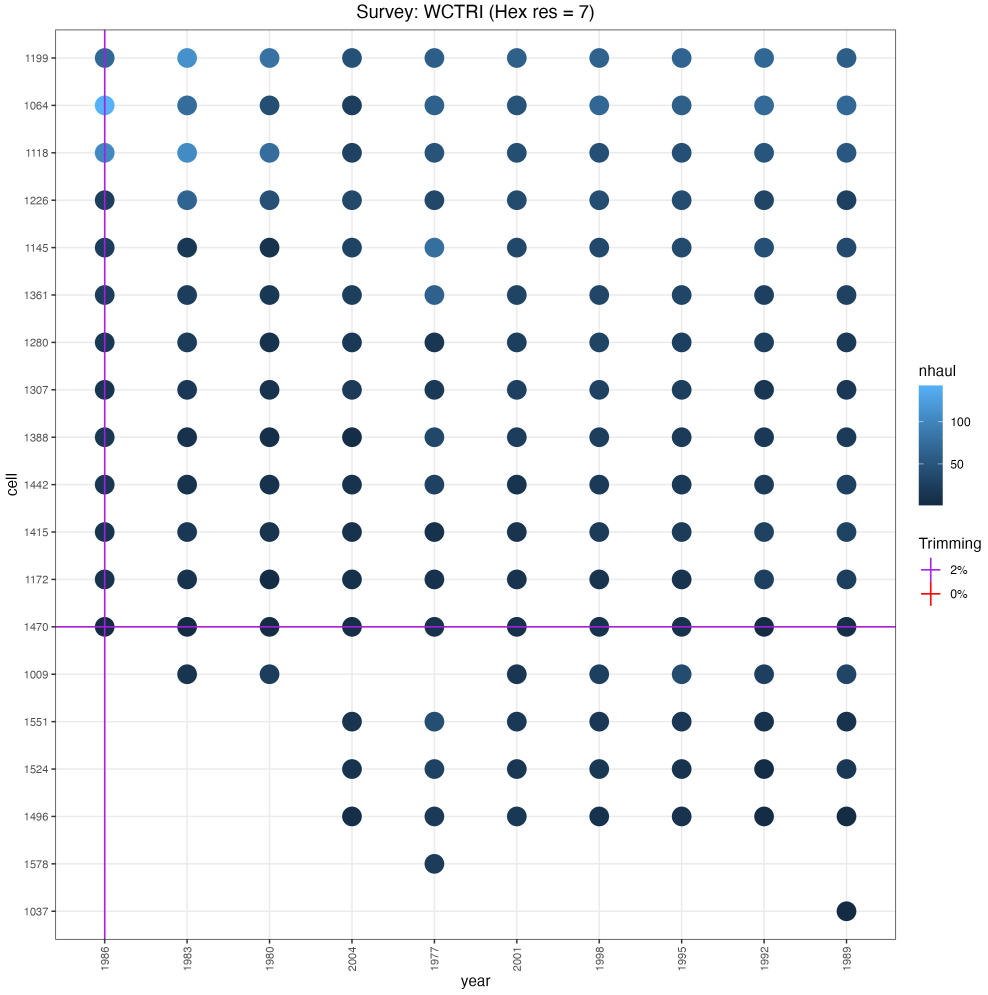
Total number of species	302.0
Percentage of species flagged	11.6

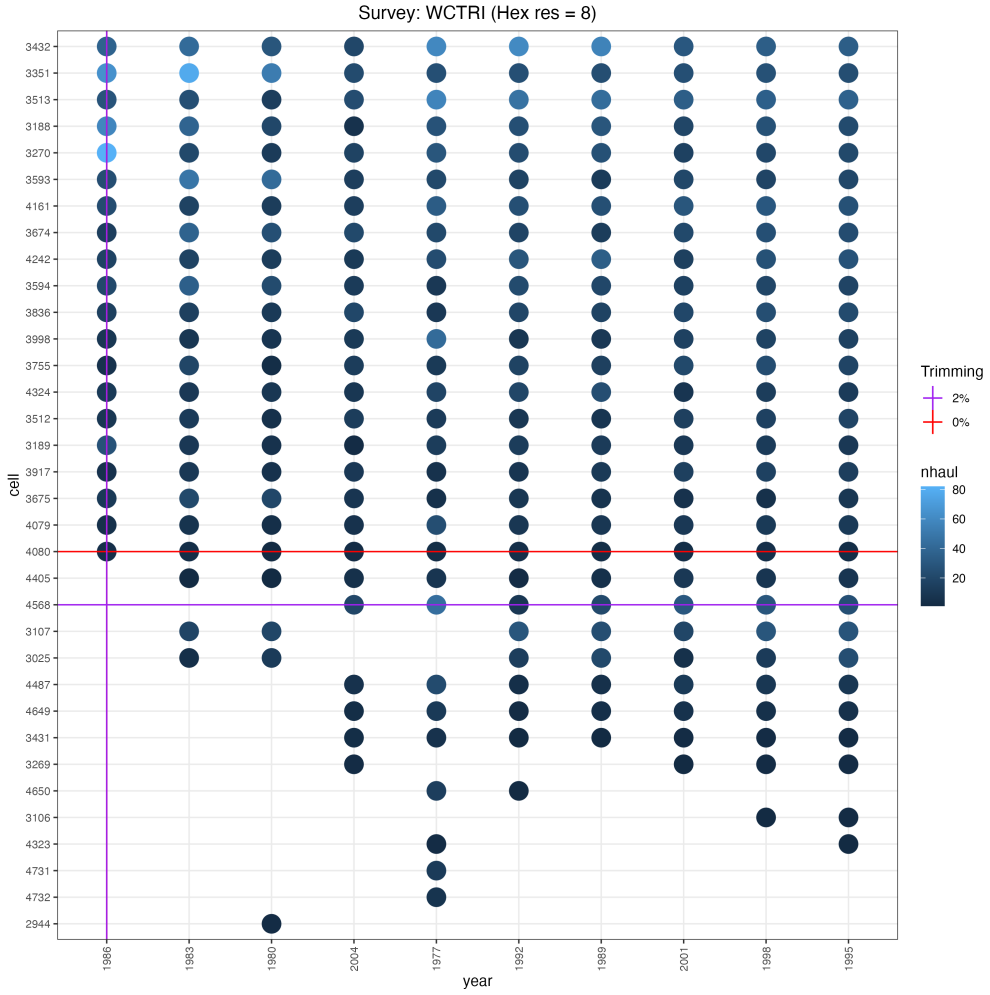
10. Spatio-temporal standardization

a. Standardization method 1

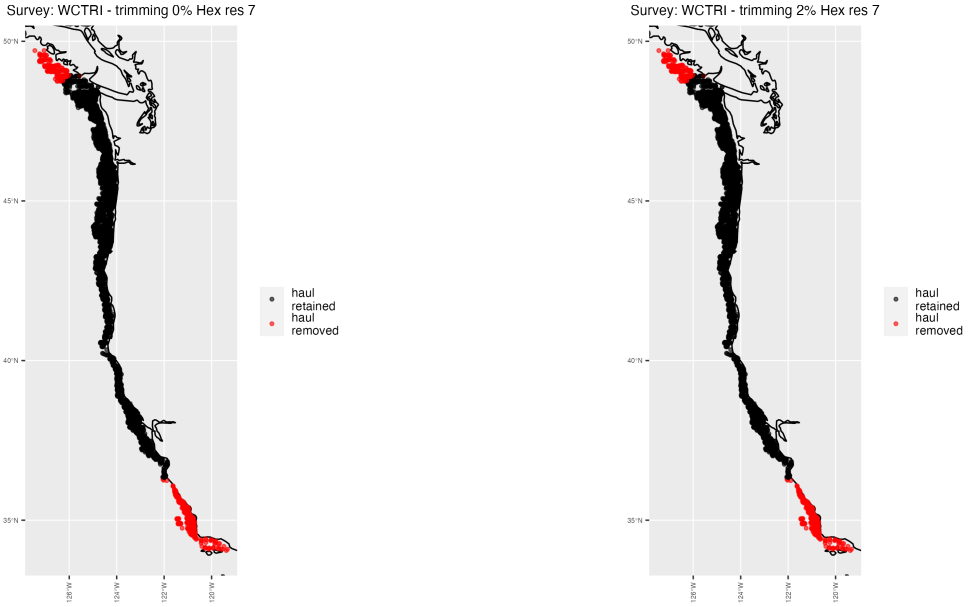
This standardization method was adapted from https://github.com/zoekitchel/trawl_spatial_turnover/blob/master/data_prep_code/species/explore_NorthSea_trimming.Rmd
It was run for hex resolution 7 and 8.

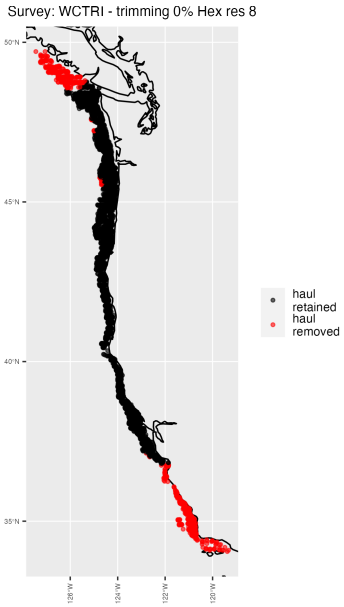
Plot of number of cells x years with overlaid flagging options

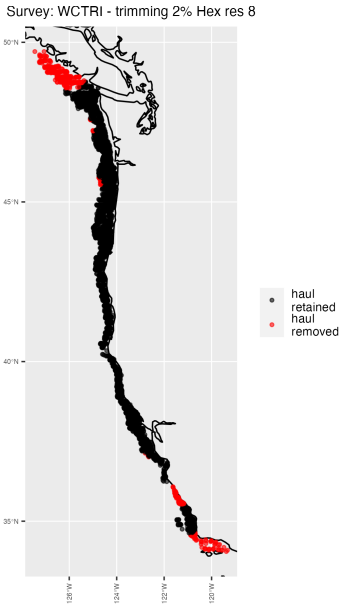




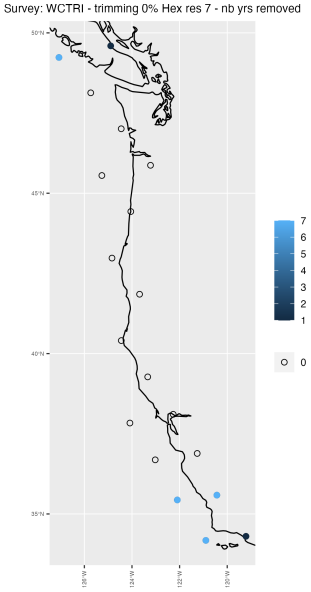
Map of hauls retained and removed per flagging method and threshold

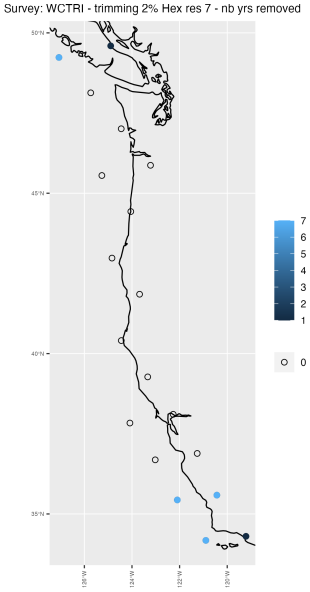


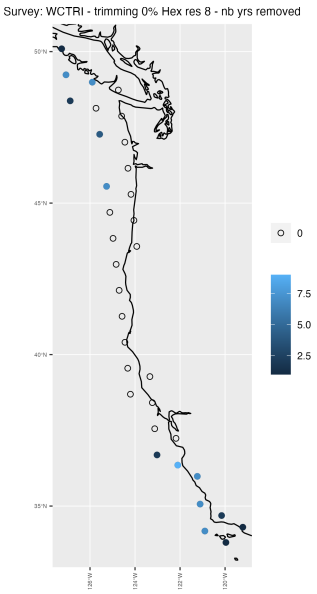


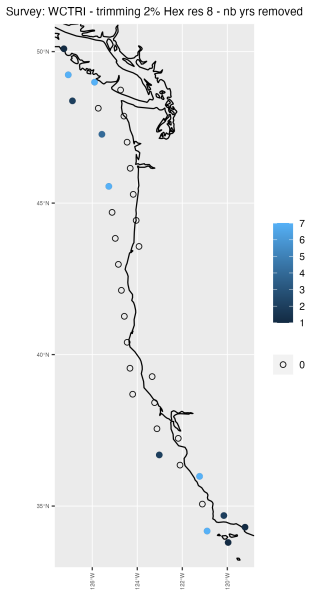


Map of numbers of years removed per grid cell and flagging method/threshold





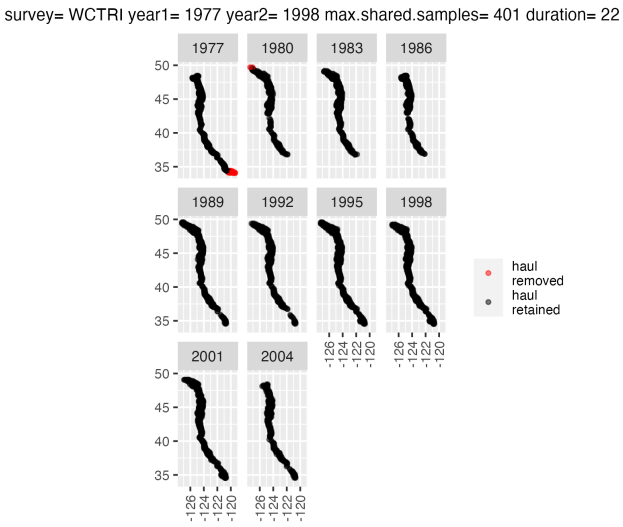




b. Standardization method 2

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	490.0	490.0	651.0	426.0	383.0
percentage of hauls removed	10.8	10.8	14.3	9.4	0.5