

# Get and summarize AFSC survey data

Kirstin Holsman, Alaska Fisheries Science Center

## Contents

Overview	1
Comparison of Bering Sea survey values	4
Code	4
Appendix 1: R/setup.Rprimary setup script . . . . .	10

**AFSC Survey CPUE data:** [github.com/kholsman/AFSC\\_CPUE](https://github.com/kholsman/AFSC_CPUE) Repo maintained by:

Kirstin Holsman

Alaska Fisheries Science Center

NOAA Fisheries, Seattle WA

**[kirstin.holsman@noaa.gov](mailto:kirstin.holsman@noaa.gov)**

*Last updated: Mar 03, 2023*

## Overview

The below scripts return a list object `cpue_data` saved as a compressed Rdata file with the naming ‘reg.srvy#.spp.cpue\_data.Rdata’ such as “ebs.srvy98.plk.cpue\_data.Rdata”. Each `cpue_data` list contains 8 data.frames:

```
load(paste0("data/out/",qrydate,"/cpue/ebs/ebs.srvy98.plk.cpue_data.Rdata"))
names(cpue_data)
```

There is a folder for each region “ebs”, “goa”, “ai”. For the “ebs” (Bering Sea) there are two sets of `cpue_data`, one that is NEBS+SEBS combined (‘ebs.srvy98.[sp].cpue\_data.Rdata’) and one that is just SEBS survey areas (‘sebs.srvy98.[sp].cpue\_data.Rdata’). For both the Gulf of Alaska (“goa”) and the Bering Sea, mean CPUE (Kg per km<sup>2</sup> or Number per km<sup>2</sup>) for each size bin at each strata was calculated and then multiplied by the STRATA area to get total Biomass and abundance. **Note: Since strata area estimates were not available for the Aleutian Island (“ai”) or slope surveys (“slope”) these AREA was set equal to 1 and the Total Biomass and abundance is actually the sum of mean biomass.**

The data.frames within each `cpue_data` object are:

1. **totalB\_N**: Total biomass (kg) or abundance (# of fish) for the species in each year

2. **mnCPUE\_strata\_yr** : Average survey CPUE (kg per Km<sup>2</sup>) or abundance (# per Km<sup>2</sup>) for the species in each strata and year
3. **total\_bin\_B\_N**: Total biomass (kg) or abundance (# of fish) for each bin (10 mm) for the species in each year
4. **mnCPUE\_strata\_bin\_yr** : Average survey CPUE (kg per Km<sup>2</sup>) or abundance (# per Km<sup>2</sup>) for each size bin for the species in each strata and year
5. **CPUE\_station\_bin\_yr**: Station specific survey CPUE (kg per Km<sup>2</sup>) or abundance (# per Km<sup>2</sup>) for each size bin for the species in each year
6. **CPUE\_station\_yr**: Station specific survey CPUE (kg per Km<sup>2</sup>) or abundance (# per Km<sup>2</sup>) for the species in each year
7. **propByBin**: proportion of biomass in each size bin per species per year
8. **propByStrata**: proportion of biomass in each strata per species per year
9. **propByStrataBin**: proportion of biomass in each bin and strata per species per year

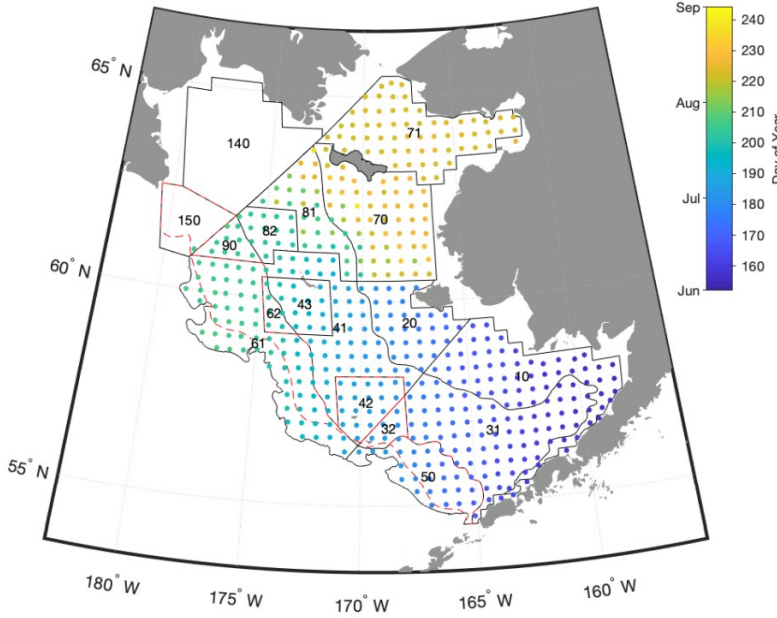


Figure 6: Regional polygons and survey-replication stations associated with the ACLIM indices.

These are calculated from the RACEBASE data tables for survey results where total CPUE was recorded for the species  $s$  (location\_catch) at each haul, expanded to include stations  $i$  where CPUE=0 (location) and expanded to each size bin  $l$  using the proportional subset of frequency of fish of given length (mm), binned into 10 mm bins ( $l$ ) and predicted weight ( $\hat{W}$ ) for each size bin  $l$  at each station  $i$ :

$$B_{s,y} = CPUE_{s,k,y} \cdot A_k$$

where  $A_k$  is the area of the strata  $k$  in Km<sup>2</sup> and  $CPUE_{s,k,y}$  is the strata specific average CPUE (kg per Km<sup>2</sup> or number per Km<sup>2</sup>) of all stations  $i$  in strata  $k$ :

$$CPUE_{s,k,y} = \frac{1}{n_k} \cdot \sum_{n_k} CPUE_{s,k,y,i}$$

where  $CPUE_{s,k,y,i}$  is the station specific CPUE (saved as the object `cpue_data$CPUE_station_yr`).

To obtain population level estimates of the biomass or abundance of fish by size bin  $l$ , we used a length weight regression to estimate the weight of each size fish  $j$  measured ( $\hat{W}$ ) to calculate the proportion by weight or frequency at each station where

$$\hat{W} = \alpha_s + L_j^{\beta_s}$$

where  $\alpha_s$  and  $\beta_s$  are fit (updated each year) to all available length and weight data from surveys from all years across the EBS, GOA, and AI (rather than each basin separately) and

$$p_{l,i}^w = \frac{N_{l,i} \hat{W}_{l,i}}{\sum N_{l,i} \hat{W}_{l,i}}$$

and

$$p_{l,i}^N = \frac{N_{l,i}}{\sum N_{l,i}}$$

This was then multiplied by the CPUE at each station ( $CPUE_{s,k,y,i}$ ) to obtain a station estimate of CPUE by size bin  $l$ :

$$CPUE_{s,k,y,l,i} = p_{l,i}^N CPUE_{s,k,y,i}$$

Finally, the average strata CPUE ( $mnCPUE_{s,k,y,l}$ ) was calculated as:

$$mnCPUE_{s,k,y,l} = \frac{1}{n_k} \cdot \sum_{n_k} CPUE_{s,k,y,l,i}$$

and strata- and bin-specific biomass (Kg) was calculated the product of mean CPUE (Kg per Km<sup>2</sup>) and strata area (Km<sup>2</sup>):

$$B_{s,y,k,l} = mnCPUE_{s,k,y,l} \cdot A_k$$

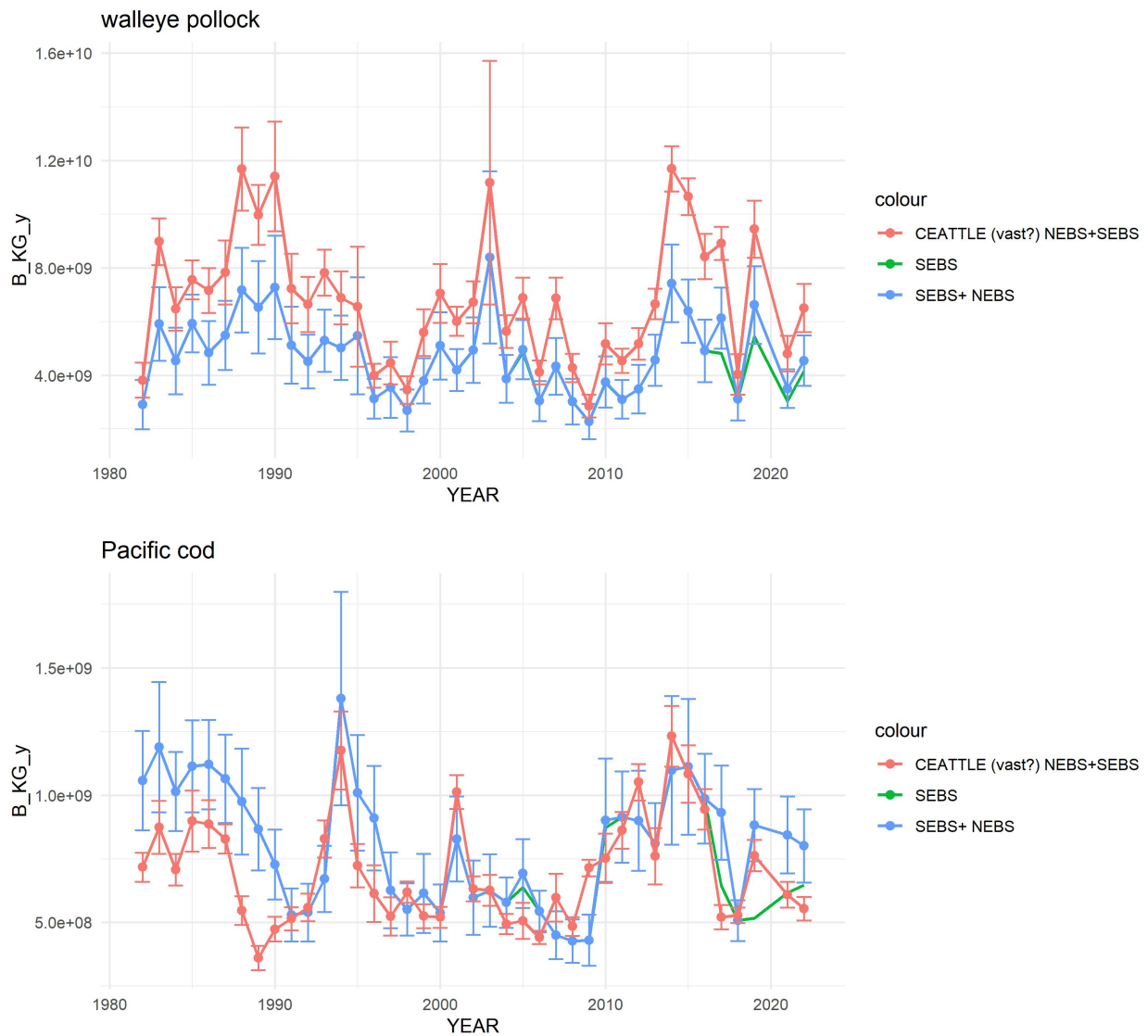
and total annual biomass by bin is the sum of strata-specific biomass for each species:

$$B_{s,y,l} = \sum_{n_k} B_{s,y,k,l}$$

and total annual biomass is:

$$B_{s,y} = \sum_{n_l} \sum_{n_k} B_{s,y,k,l}$$

## Comparison of Bering Sea survey values



## Code

```
# ## Step 0: Set up the R workspace
#
# The first step is to set up the switches for what files to update and create in the file `R/setup.R`.
#
# ## Step 1: Update SQL queries
# This step must be run on a computer that has access to RACEBASE. The code below will generate the bas

# **IMPORTANT:**
#
# * **This step must be connected to the RACEBASE SQL database**
```

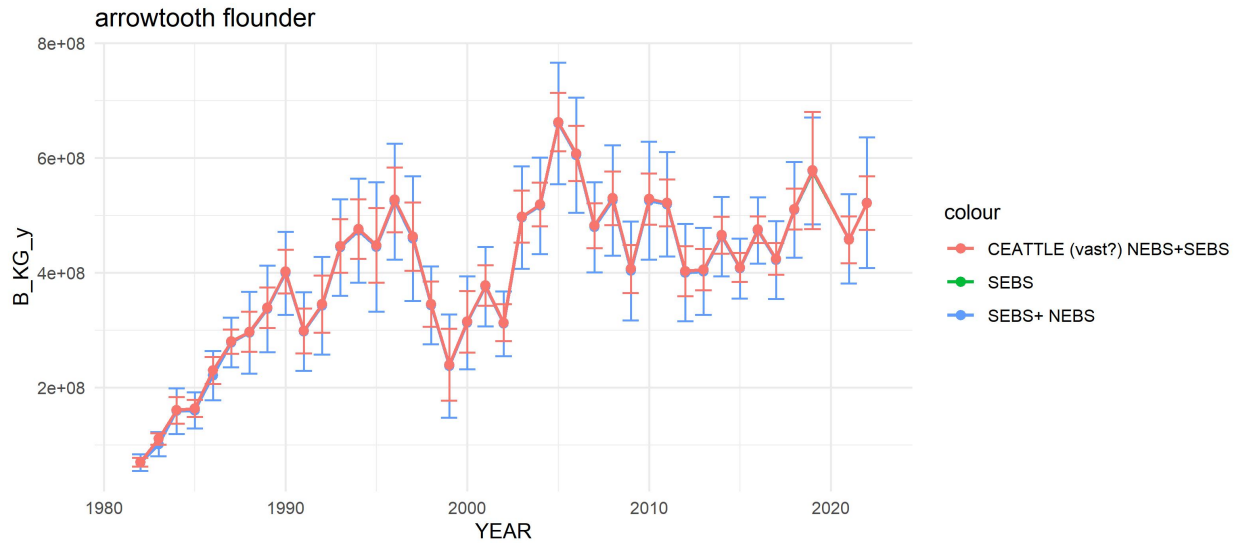


Figure 1: "arrowtooth"

```
#
# * **To change R studio from the default 64 bit to 32 bit go to Tools>Global options and select the
#
# * **The code will connect to the SQL database using your password and username. Remember to update
#
# <!-- ![Header of `setup.R` where `username_path` can be adjusted. This file also is where species, re
#

# get everything set up:
#-----
# rm(list=ls())
# this uses the password saved in R/password.R
suppressMessages(source("R/make.R"))

# update the SQL queries
#-----

source(file.path(code.path, "R/sub_scripts/runRACE_qrys.R"))

# combine sebs and nebs into one region: ebs
if(dir.exists(file.path(data.path, "ebs")))
  system(paste("rm -r", file.path(data.path, "ebs")))
dir.create(file.path(data.path, "ebs"))

# combine files and rename survey area to all of EBS
for(sp in names(splist)){
  if(dir.exists(file.path(data.path, "ebs", sp)))
    system(paste("rm -r", file.path(data.path, "ebs", sp)))
  dir.create(file.path(data.path, "ebs", sp))
  #"length.Rdata"
  load(file.path(data.path, "nebs", sp, "length.Rdata"))
}
```

```

length_nebs <- length;rm(length)
load(file.path(data.path,"sebs",sp,"length.Rdata"))
length_sebs <- length;rm(length)
length<- rbind(length_nebs%>%
  mutate(SURVEY_DEFINITION_ID_aka =SURVEY_DEFINITION_ID,SURVEY_DEFINITION_ID =98),
  length_sebs%>%
  mutate(SURVEY_DEFINITION_ID_aka =SURVEY_DEFINITION_ID,SURVEY_DEFINITION_ID =98))

save(length,file = file.path(data.path,"ebs",sp,"length.Rdata"))
rm(length)

#"location.Rdata"
load(file.path(data.path,"nebs",sp,"location.Rdata"))
location_nebs <- location;rm(location)
load(file.path(data.path,"sebs",sp,"location.Rdata"))
location_sebs <- location;rm(location)
location<- rbind(location_nebs, location_sebs)

save(location,file = file.path(data.path,"ebs",sp,"location.Rdata"))

#"location_catch.Rdata"
load(file.path(data.path,"nebs",sp,"location_catch.Rdata"))
location_catch_nebs <- location_catch;rm(location_catch)
load(file.path(data.path,"sebs",sp,"location_catch.Rdata"))
location_catch_sebs <- location_catch;rm(location_catch)
location_catch <- rbind(location_catch_nebs%>%
  mutate(SURVEY_DEFINITION_ID_aka =SURVEY_DEFINITION_ID,SURVEY_DEFINITION_ID =98),
  location_catch_sebs%>%
  mutate(SURVEY_DEFINITION_ID_aka =SURVEY_DEFINITION_ID,SURVEY_DEFINITION_ID =98))

save(location_catch,file = file.path(data.path,"ebs",sp,"location_catch.Rdata"))

}

#```

## Step 2: Update the LWA regressions

# The default code for RACEBASE uses set LW relationships, however we prefer to update the LW regressions
#
# #```{r updateLWglms, echo=TRUE, eval=FALSE}

# update the LW regressions
#-----

if(update_LWdata){
  source(file.path(code.path,"R/sub_scripts/updateLW.R"))
  # reload with updated data:
  source(file.path(code.path,"R/load_data.R"))
}

```

```

species_lkup

#``

## Step 3: Get CPUE data from the surveys

#This code is the core script for generating the CPUE_NUMKM2 and CPUE_BIOMKM2 values by size bin, region

#``{r updateCPUE, echo=TRUE, eval=FALSE}

STRATA_AREA%>%filter(REGION=="BS")%>%
  group_by(YEAR)%>%summarise(mnAREA = mean(AREA, na.rm=T),
                             sumAREA = sum(AREA, na.rm=T),
                             cnt = length(unique(STRATUM)))

STRATA_AREA%>%filter(REGION=="GOA")%>%
  group_by(YEAR)%>%summarise(mnAREA = mean(AREA, na.rm=T),
                             sumAREA = sum(AREA, na.rm=T),
                             cnt = length(unique(STRATUM)))

STRATA_AREA%>%filter(REGION=="BS",YEAR==2022)%>%select(STRATUM)

# overwrite the NEBS frame from setup for the next set of code (ebs = sebs+nebs now forward)
srvys <- data.frame(reg=c("ebs","goa","ai","slope"),RGN = c("BS","GOA","AI","SLOPE"), num=c(98,47,52,
# srvys <- data.frame(reg=c("ebs","goa","ai"),RGN = c("BS","GOA","AI"), num=c(98,47,52) )
nreg <- length(srvys$reg)
nspp <- length(species_lkup$sp)

for (r in 1:nreg){
  for(s in 1:nspp){

    if(srvys[r,]$reg == "ebs"){
      # first SEBS only:
      # -----
      STRATA_AREAUSE <- STRATA_AREA%>%filter(REGION==srvys$RGN[r])
      maxyr <- max(STRATA_AREAUSE$YEAR)
      STRATA_AREAUSE <- STRATA_AREAUSE%>%
        filter(YEAR==2022)%>%
        group_by(REGION,STRATUM)%>%
        summarize(AREA = mean(AREA, na.rm=T))%>%ungroup()

      flnm <- paste0("s",srvys[r,]$reg,".srvy",
                     srvys[r,]$num,".",
                     species_lkup[s,]$sp)
      cat("now getting data for: ",flnm,"\n")
      cpue_data <- suppressMessages(
        get_CPUE_DATA(
          datapath = data.path,
          out_dir = file.path(data.out),
          STRATA_AREAIN = STRATA_AREAUSE,
          flnm = flnm,
          species = species_lkup[s,]$SPECIES_CODE,

```

```

    survey      = srvys[r,]$num,
    includeNBS  = FALSE,
    NEBSstrataIN = NEBS_strata ,
    saveit      = T,
    bins        = sp_bins[[ species_lkup[s,]$sp ]]))

rm(cpue_data)

# Now NESB + SEBS
# -----
flnm <- paste0(srvys[r,]$reg, ".srvy",
               srvys[r,]$num, ".",
               species_lkup[s,]$sp)
cat("now getting data for: ", flnm, "\n")
cpue_data <- suppressMessages(
  get_CPUE_DATA(
    datapath    = data.path,
    out_dir     = file.path(data.out),
    STRATA_AREA_IN = STRATA_AREAUSE,
    flnm        = flnm,
    species     = species_lkup[s,]$SPECIES_CODE,
    survey      = srvys[r,]$num,
    includeNBS  = TRUE,
    NEBSstrataIN = NEBS_strata ,
    saveit      = T,
    bins        = sp_bins[[ species_lkup[s,]$sp ]]))
}

if(srvys[r,]$reg == "goa"){
  STRATA_AREAUSE <- STRATA_AREAUSE %>% filter(REGION==srvys$RGN[r])
  maxyr <- max(STRATA_AREAUSE$YEAR)
  STRATA_AREAUSE <- STRATA_AREAUSE %>%
    filter(YEAR==1993) %>%
    group_by(REGION, STRATUM) %>%
    summarize(AREA = mean(AREA, na.rm=T)) %>% ungroup()

  flnm <- paste0(srvys[r,]$reg, ".srvy",
                 srvys[r,]$num, ".",
                 species_lkup[s,]$sp)
  cat("now getting data for: ", flnm, "\n")
  cpue_data <- suppressMessages(
    get_CPUE_DATA(
      datapath    = data.path,
      out_dir     = file.path(data.out),
      STRATA_AREA_IN = STRATA_AREAUSE,
      flnm        = flnm,
      species     = species_lkup[s,]$SPECIES_CODE,
      survey      = srvys[r,]$num,
      includeNBS  = FALSE,
      NEBSstrataIN = NEBS_strata ,
      saveit      = T,
      bins        = sp_bins[[ species_lkup[s,]$sp ]]))
}

```



```

}
if(!srvys[r,]$reg%in%c("ebs","goa")){
  flnm <- paste0(srvys[r,]$reg,".srvy",
                 srvys[r,]$num,".",
                 species_lkup[s,]$sp)
  cat("now getting data for: ",flnm,"\n")
  cpue_data <- suppressMessages(
    get_CPUE_DATA(
      datapath = data.path,
      out_dir  = file.path(data.out),
      STRATA_AREAIN = NULL,
      flnm      = flnm,
      species   = species_lkup[s,]$SPECIES_CODE,
      survey    = srvys[r,]$num,
      includeNBS = FALSE,
      NEBSstrataIN = NEBS_strata ,
      saveit    = T,
      bins      = sp_bins[[ species_lkup[s,]$sp ]]))
}

## check the data :
if(1==10){
  tt <- cpue_data%>%
    group_by(YEAR,REGION,STATIONID,SN)%>%
    filter(BIN ==400)%>%
    summarize(cnt =length(STATIONID))
  max(tt$cnt) #Should be 1
  #this looks to be a duplicate sampling...
  #mis-entry or code error ?
  cpue_data%>%filter(YEAR==1988,STATIONID=="J-13")
}
rm(cpue_data)

}
}

#
# ...
#
# *The cpue files are now saved in the directory `r file.path(data.out,"../")`*
#
# ```{r viewcpue_data, echo=TRUE, eval=FALSE}
#   # this uses the password saved in R/password.R
#   # suppressMessages(source("R/make.R"))
load(file.path(data.out,"cpue/ebs/ebs.srvy98.pcod.cpue_data.Rdata"))

names(cpue_data)

library(dplyr)

checkit <-function(x){
  if(round(max(x ),1)!=1) {
    warning("ERROR! propB > 1 ")
  }
}

```

```

    print(x)}
}

#double check the results
cnt_ByStrataBin <- cpue_data$propByStrataBin%>%
  select("REGION", "YEAR", "STRATUM", BIN, BIN_mm,
         SPECIES_CODE, CN, SN, sp, num,
         "propB_yk1", "propN_yk1")%>%
  group_by(YEAR, REGION, SN, CN)%>%
  summarise(sum_propB_yk1=sum(propB_yk1, na.rm=T),
            sum_propN_yk1=sum(propN_yk1, na.rm=T))

cnt_ByStrata <- cpue_data$propByStrata%>%
  select("REGION", "YEAR", "STRATUM",
         SPECIES_CODE, CN, SN, sp, num,
         "propB_yk", "propN_yk")%>%
  group_by(YEAR, REGION, SN, CN)%>%
  summarise(sum_propB_yk=sum(propB_yk, na.rm=T),
            sum_propN_yk=sum(propN_yk, na.rm=T))

cnt_ByBin <- cpue_data$propByBin%>%
  select("REGION", "YEAR", BIN, BIN_mm, num,
         SPECIES_CODE, CN, SN, sp,
         "propB_y1", "propN_y1")%>%
  group_by(YEAR, REGION, SN, CN)%>%
  summarise(sum_propB_y1=sum(propB_y1, na.rm=T),
            sum_propN_y1=sum(propN_y1, na.rm=T))

checkit(cnt_ByStrataBin$sum_propB_yk1)
checkit(cnt_ByStrata$sum_propB_yk)
checkit(cnt_ByBin$sum_propB_y1)

#

```

## Appendix 1: R/setup.Rprimary setup script