Doc

Recreational catch of Black sea bass

"Ricky Tabandera"

```
knitr::opts_chunk$set(echo = TRUE)
library("tidyverse")
```

```
## Warning: package 'tidyverse' was built under R version 4.0.4
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.1.0     v dplyr   1.0.5
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
```

```
## Warning: package 'tidyr' was built under R version 4.0.4
```

```
## Warning: package 'dplyr' was built under R version 4.0.4
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
#read in recreational landings data
rec_landings_1950_2019<-readr::read_csv(here::here("data","foss_landings_REC_1950_2019_MA_NE.csv"))
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   Year = col_double(),
##   State = col_character(),
##   `NMFS Name` = col_character(),
##   Pounds = col_number(),
##   Dollars = col_logical(),
##   Collection = col_character(),
##   Confidentiality = col_character()
## )
```

```
rec_landings_1950_2019<-rec_landings_1950_2019 %>% dplyr::rename( "common_name"= "NMFS Name")
rec_landings_1950_2019<-as.data.frame(rec_landings_1950_2019)

#load in the stocks managed in NEFSC
stock_list_all_strata <- read.csv("https://raw.githubusercontent.com/NOAA-EDAB/ECSA/master/data/stock_li
#lowercase all of the names
stock_list_all_strata$common_name <- stringr::str_to_lower(stock_list_all_strata$common_name)
#extract out the unique names and rename column
stock_list<-as_tibble(unique(stock_list_all_strata$common_name),column_name = "common_name")
stock_list<-stock_list %>% dplyr::rename(common_name= value)

#summarize the catches to species and year
rec_by_sp<-rec_landings_1950_2019 %>% group_by(common_name,Year) %>% summarise(total= sum(Pounds), sd =
```

```
## `summarise()` has grouped output by 'common_name'. You can override using the `.groups` argument.
```

```
#lower case the name for easier comparisons
rec_by_sp$common_name<-stringr::str_to_lower(rec_by_sp$common_name)
rec_landings_1950_2019$common_name<-stringr::str_to_lower(rec_landings_1950_2019$common_name)
#split out the records with commas to be re arranged
rec_w_comma<-rec_by_sp[str_detect(rec_by_sp$common_name, "\\,\\s"),]
rec_landings_1950_2019_comma<-rec_landings_1950_2019[str_detect(rec_landings_1950_2019$common_name, "\\
# detect the comma space formatting and separate out each term
rec_w_comma_split<-rec_w_comma %>% separate(col=common_name,sep ="\\,\\s", into = c("first" ,"second",
```

```
## Warning: Expected 3 pieces. Additional pieces discarded in 2 rows [227, 228].
```

```
## Warning: Expected 3 pieces. Missing pieces filled with `NA` in 4777 rows [1, 2,
## 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].
```

```
rec_landings_1950_2019_comma_split<-rec_landings_1950_2019_comma %>% separate(col=common_name,sep ="\\,
```

```
## Warning: Expected 3 pieces. Additional pieces discarded in 2 rows [10287,
## 11688].
```

```
## Warning: Expected 3 pieces. Missing pieces filled with `NA` in 15433 rows [1, 2,
## 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].
```

```
#reorder the tibble to match NEFSC names
rec_w_comma_split<-rec_w_comma_split %>% select( third, second, first,Year, total ,sd)
rec_landings_1950_2019_comma_split<-rec_landings_1950_2019_comma_split %>% select(third, second, first,
########################## renaming rows to align with NEFSC naming

#removing the shark tag from the first name term
rec_w_comma_split<- as.data.frame(rec_w_comma_split)

rec_w_comma_split[str_detect(rec_w_comma_split$first, "shark"),3]<-NA
#changing tibble back to df to simplify the string replacement

rec_landings_1950_2019_comma_split[str_detect(rec_landings_1950_2019_comma_split$first, "shark"),3]<-NA
```

```r
# fixing flounder plaice
rec_w_comma_split[str_detect(rec_w_comma_split$second, "plaice"),3]<-NA
rec_landings_1950_2019_comma_split[str_detect(rec_landings_1950_2019_comma_split$second, "plaice"),3]<-N


#used to check for near matches
#fish_match<-rec_w_comma_whole[str_detect(rec_w_comma_whole$common_name, "herring"),]

rec_w_comma_union<-rec_w_comma_split %>% unite(col = "firstname", c(third, second),sep = " ", remove = T
rec_w_comma_whole<-rec_w_comma_union %>% unite(col = "common_name", c(firstname, first ),sep = " ", remo
rec_landings_1950_2019_comma_u<-rec_landings_1950_2019_comma_split %>% unite(col = "firstname", c(third
rec_landings_1950_2019_comma_w<-rec_landings_1950_2019_comma_u %>% unite(col = "common_name", c(firstnam

#bring back in the records of single name fishes

rec_w_o_comma<-rec_by_sp[!str_detect(rec_by_sp$common_name, "\\,\\s"),]
rec_landings_1950_2019<-as.data.frame(rec_landings_1950_2019)
rec_landings_1950_2019_w_o_comma<-rec_landings_1950_2019[!str_detect(rec_landings_1950_2019$common_name

# rename col names to merge with the corrected dataset
colnames(rec_w_o_comma)<-c("common_name","Year","total","sd")
rec_by_sp_whole<-dplyr::bind_rows(rec_w_o_comma,rec_w_comma_whole)
#common_name ,Year, State, Pounds, Dollars, Collection, Confidentiality
rec_landings_1950_2019_w_o_comma<-rec_landings_1950_2019_w_o_comma %>% dplyr::select("common_name" ,"Yea
rec_landings_1950_2019_whole<-dplyr::bind_rows(rec_landings_1950_2019_w_o_comma,rec_landings_1950_2019_c

#fixing windowpane
rec_by_sp_whole[stringr::str_detect(rec_by_sp_whole$common_name, "window"),1]<-"windowpane flounder"
rec_landings_1950_2019_whole[stringr::str_detect(rec_landings_1950_2019_whole$common_name,"window"),1 ]<
#check how many matches already exist
#there are 42 species in NEFSC stock list
#length(stock_list$common_name)


stock_semi<-semi_join(stock_list, rec_by_sp_whole, by = "common_name")
stock_missing<-anti_join(stock_list, stock_semi, by = "common_name")
#length(stock_missing$common_name)
#the missing stocks
#  1 atlantic hagfish no records
#  2 offshore hake    multiple species, ECSA uses red and silver hake, not sure what to do with other sp
#fish_match<-rec_w_comma_whole[str_detect(rec_w_comma_whole$common_name, "hake"),]
# 3 witch flounder  no records
# 4 monkfish no records
# 5 american lobster no records
# 6 northern shrimp   no records
# 7 northern shortfin squid no records
# 8 longfin inshore squid



#final dataset is
# rec_landings_clean<-rec_by_sp_whole
```

```
rec_landings_1950_2019_clean <-rec_landings_1950_2019_whole %>% mutate(common_name=stringr::str_to_sente
#write.csv(rec_landings_1950_2019_clean, "rec_landings_1950_2019_tidy.csv")
# rm(list =c("rec_by_sp", "rec_landings_1950_2019", "rec_landings_1950_2019_comma", "rec_landings_1950_
```

```
Selected_sp<-rec_landings_1950_2019_clean %>%
  group_by(Year) %>%
  filter(!is.na(Pounds) & common_name==params$species) %>%
  summarise(total=sum(Pounds,na.rm = FALSE))
selected_by_state<-rec_landings_1950_2019_clean %>%
  filter(!is.na(Pounds) & common_name==params$species) %>%
  group_by(State, Year)
```

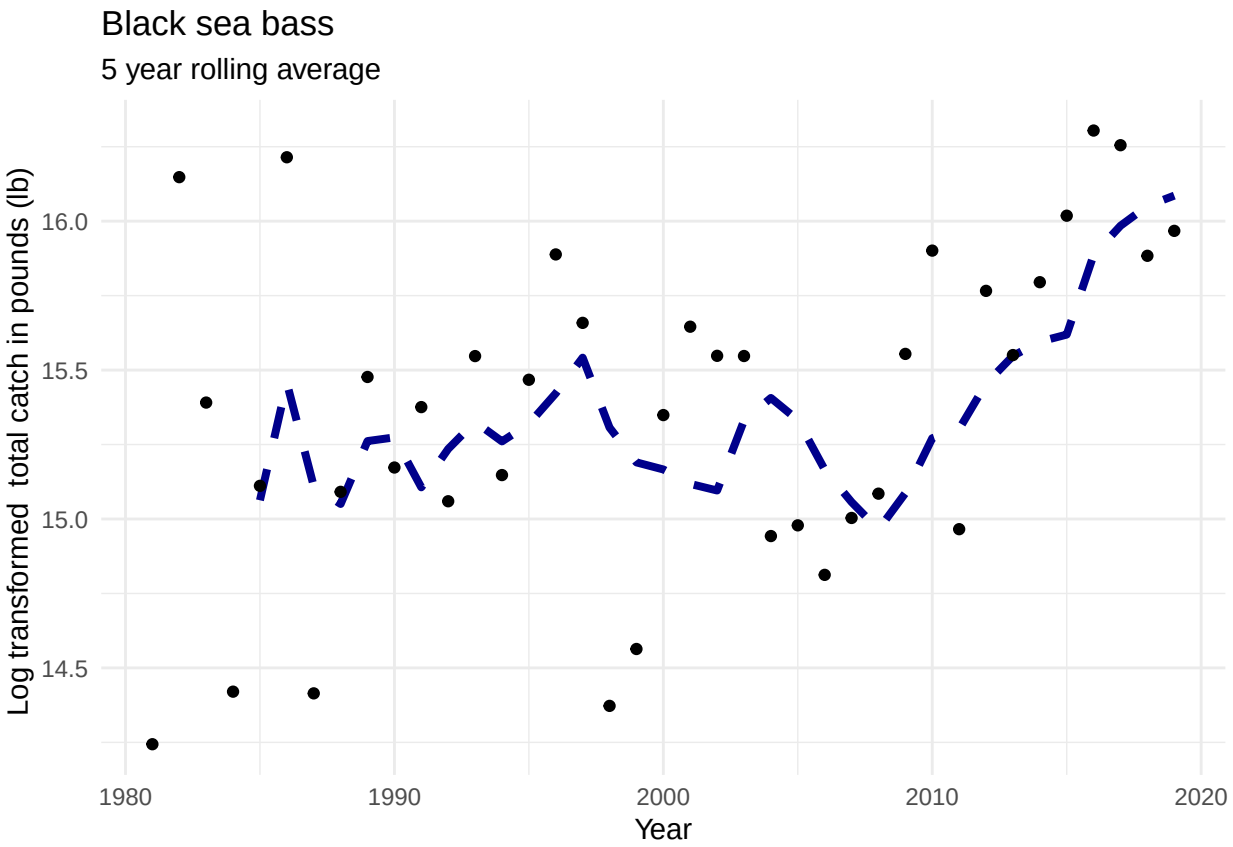
```
sp_data_qual<-TRUE
st_data_qual<-TRUE

#testing if data quality is sufficent for each graph
if(length(which(is.na(Selected_sp$total)))/length(Selected_sp$total)>0.40){ sp_data_qual<-FALSE}else{sp_

if(length(which(is.na(selected_by_state$Pounds)))/length(selected_by_state$Pounds) >0.50){st_data_qual<-
```

Recreational catches time series

This data was sourced from FOSS - Fisheries One Stop Shop and is inclusive of 1981 to 2019. The entire data set contains 475 species. Figures produced reflect the coverage of the data. With stocks that have high coverage, a running average is calculated. In low coverage stocks, missing values are excluded and a simple time series is produced.

```
Selected_sp%>%
  ggplot2::ggplot(aes(x=Year, y=log(total)))+
  tidyquant::geom_ma( n=5, lwd=1.4)+
  geom_point()+
  ggtitle(label = params$species ,subtitle = "5 year rolling average")+
  xlim(1981,2019)+
  xlab(label= "Year")+
  ylab(label = "Log transformed  total catch in pounds (lb) ")+
  theme_minimal()
```

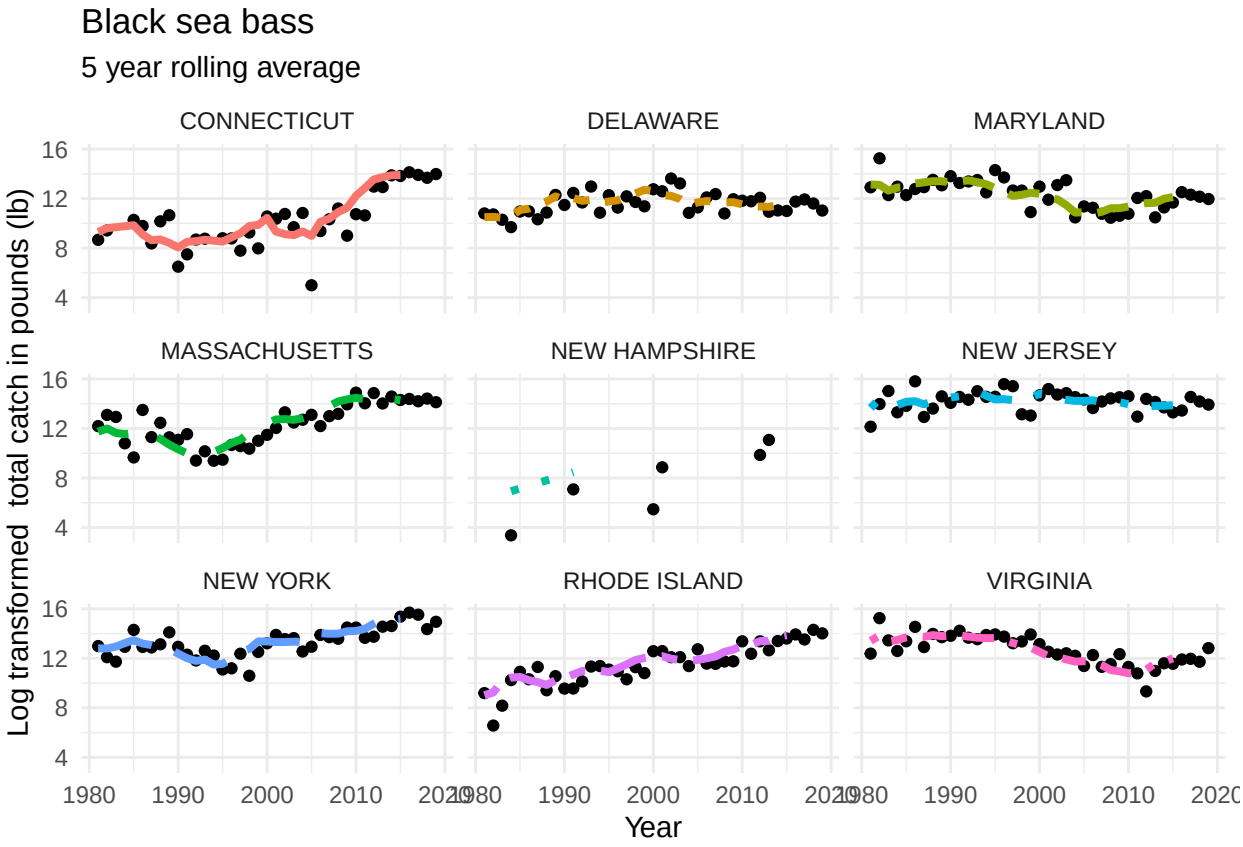
```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```




```
Selected_sp%>%na.omit()%>%
  ggplot2::ggplot(aes(x=Year, y=log(total)))+
  geom_point(size=1.3)+
  geom_line()+
  ggtitle(label = params$species ,subtitle = "Data coverage is poor")+
  xlim(1981,2019)+
  xlab(label= "Year")+
  ylab(label = "Log transformed  total catch in pounds (lb)")+
  theme_minimal()
```

Recreational landings of Black sea bass across states in the mid-Atlantic and north-east regions

```
selected_by_state %>%
  ggplot2::ggplot(aes(x=Year, y=log(Pounds)))+
  geom_point()+
  tidyquant::geom_ma(aes(color = State, lty= State),n=5, lwd=1.4,na.rm = TRUE,show.legend = FALSE)+
  facet_wrap(vars(State))+
  ggtitle(label = params$species ,subtitle = "5 year rolling average")+
  xlim(1981,2019)+
  xlab(label= "Year")+
  ylab(label = "Log transformed  total catch in pounds (lb) ")+
  theme_minimal()
```




```
selected_by_state%>%
  ggplot2::ggplot(aes(x=Year, y=log(Pounds)))+
  geom_point(aes(color=State),show.legend = FALSE)+
  geom_line(aes(color=State,lty=State),lwd=1.3,show.legend = FALSE)+
  facet_wrap(vars(State))+
  xlim(1981,2019)+
  ggtitle(label = params$species ,subtitle = "Data coverage is poor")+
  xlab(label= "Year")+
  ylab(label = "Log transformed  total catch in pounds (lb) ")+
  theme_minimal()
```