Zooplankton Data Input for LCM

Brian Mahardja

2023-04-12

Original script and dataset packages was written by Sam Bashevkin and can be found here: <https://github.com/sbashevkin/FLOATDrought>

# Load packages

require(conflicted)  
require(MASS)  
require(dplyr)  
require(zooper)  
require(lubridate)  
require(readr)  
require(tidyr)  
require(ggplot2)  
require(sf)  
require(readxl)  
require(stringr)  
require(mgcv)  
require(purrr)  
require(deltamapr)  
require(scales)  
library(imputeTS)  
  
conflict\_prefer("filter", "dplyr")  
conflict\_prefer("select", "dplyr")

# Load data

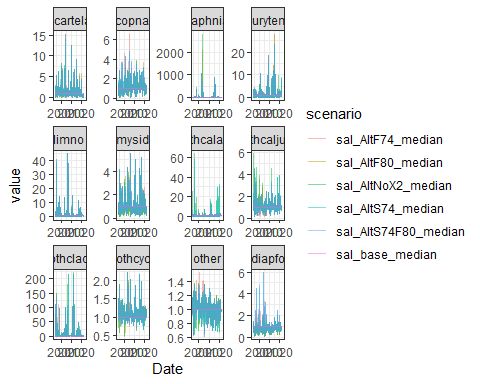
Load zooplankton model output and strata volume from Will Smith

model\_data<-read.csv(file.path("Output","zoop sal adjustments.csv"))  
  
IBMR\_data\_format <- read.csv(file.path("Data","prey\_taxa\_inputs\_IBMR.csv"))

# Impute infinity and N/A with linear interpolation

When 0s were introduced for baseline predictions, it resulted in infinite values for the scalars. We replaced these infinities with linear interpolations from the time series of model predictions for a specific alternative, taxon, and subregion. This step yielded no more infinite scalar values.

## Impute data using linear interpolation  
  
# Change inf and NaN to NA  
model\_data[sapply(model\_data, is.infinite)] <- NA  
model\_data[sapply(model\_data, is.nan)] <- NA  
  
  
# Split data into list  
zoop\_model\_data\_arranged\_median <- model\_data %>%   
 arrange(year, month) %>%  
 select("region","year","month","IBMR",ends\_with("median")) %>%  
 gather("scenario","value",ends\_with("median"))  
  
zoop\_model\_data\_median\_split <- split(zoop\_model\_data\_arranged\_median , list(zoop\_model\_data\_arranged\_median$region,zoop\_model\_data\_arranged\_median$IBMR))  
  
# Impute data using linear interpolation  
for(i in seq\_along(zoop\_model\_data\_median\_split)){   
 zoop\_model\_data\_median\_split[[i]]$value <- na\_interpolation(zoop\_model\_data\_median\_split[[i]]$value)  
}  
  
# Rejoin into a single data frame  
zoop\_model\_data\_median\_imputed<-bind\_rows(zoop\_model\_data\_median\_split) %>%  
 mutate(Date=as.Date(paste(year,month,"01",sep="-")))  
  
ggplot(data=zoop\_model\_data\_median\_imputed %>% filter(region=="NW Suisun")) + geom\_line(aes(x=Date,y=value,color=scenario),alpha=0.5) + theme\_bw() + facet\_wrap(~IBMR, scales = "free")

 # Use pdiapfor prediction for pdiapjuv

# Remove mysid since it is not used  
zoop\_model\_data\_median\_imputed <- zoop\_model\_data\_median\_imputed %>% filter(IBMR!="mysid")  
str(zoop\_model\_data\_median\_imputed)  
  
# Create pdiapjuv dataframe  
pdiapjuv\_data <- zoop\_model\_data\_median\_imputed %>% filter(IBMR=="pdiapfor") %>% mutate(IBMR=as.character("pdiapjuv"))  
  
# Add pdiapjuv back into the main prediction data  
zoop\_model\_data\_median\_imputed <- bind\_rows(zoop\_model\_data\_median\_imputed,pdiapjuv\_data)

# Format data to allow for easy input into IBMR and export

# Get the strata and taxa code in case they are necessary  
IBMR\_code <- IBMR\_data\_format %>% rename(year=Year,month=Month,region=Region,IBMR=Taxa) %>%  
 select(year,month,region,IBMR,Strata\_code,Taxa\_code) %>% distinct()  
  
# Spread zoop scalar data  
zoop\_model\_data\_median\_imputed\_spread <- spread(zoop\_model\_data\_median\_imputed,scenario,value)  
  
# Add strata and taxa code  
zoop\_model\_data\_median\_imputed\_spread <- zoop\_model\_data\_median\_imputed\_spread %>% left\_join(IBMR\_code)  
  
# Remove years that are not present in IBMR input  
zoop\_model\_data\_median\_imputed\_spread <- zoop\_model\_data\_median\_imputed\_spread %>% filter(!is.na(Strata\_code))  
  
# Export data  
write.csv(zoop\_model\_data\_median\_imputed\_spread,file.path("Output",paste("zoop\_scalar\_output\_",Sys.Date(),".csv",sep="")))