LakeMetab_ModelPrep

Kelly Loria

2023-03-04

This is one of X scripts to look at DO output as a timesieries.

```
lapply(c("plyr","dplyr","ggplot2","cowplot","lubridate",
         "tidyverse","data.table","xts","dygraphs",
         "nrlmetab","cowplot"), require, character.only=T)

source("./saved_fxns/k.vachon.R")
source("./saved_fxns/k600.2.kGAS.R")
source("./saved_fxns/getSchmidt.R")
```

Reading in processed data from DO_QAQC.r

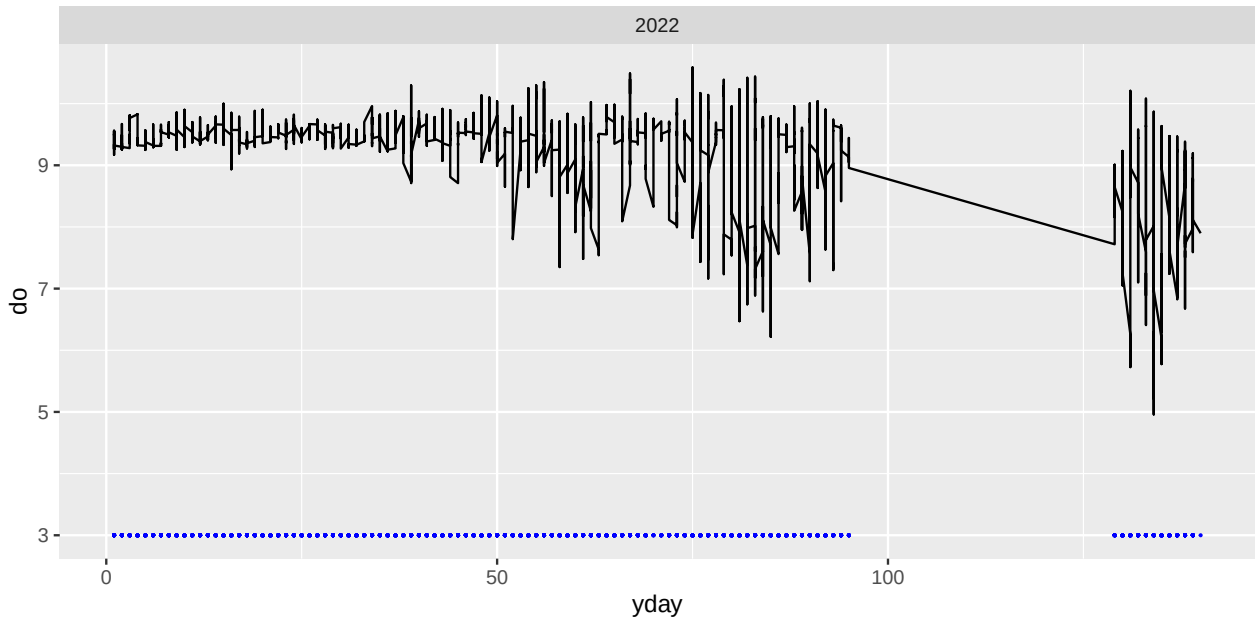
```
## 2021 ##
# BS1_dat <- read.csv("./Step1_LakeMetab_Processed/21BWNS1Inputs.csv", header=T)
# BS2_dat <- read.csv("./Step1_LakeMetab_Processed/21BWNS2Inputs.csv", header=T)
# BS3_dat <- read.csv("./Step1_LakeMetab_Processed/21BWNS3Inputs.csv", header=T)
#
# GS1_dat <- read.csv("./Step1_LakeMetab_Processed/21GBNS1Inputs.csv", header=T)
# GS2_dat <- read.csv("./Step1_LakeMetab_Processed/21GBNS2Inputs.csv", header=T)
# GS3_dat <- read.csv("./Step1_LakeMetab_Processed/21GBNS3Inputs.csv", header=T)

## 2022 ##
BS1_dat <- read.csv("./Step1_LakeMetab_Processed/22BWNS1Inputs.csv", header=T)
BS2_dat <- read.csv("./Step1_LakeMetab_Processed/22BWNS2Inputs.csv", header=T)
BS3_dat <- read.csv("./Step1_LakeMetab_Processed/22BWNS3Inputs.csv", header=T)

GS1_dat <- read.csv("./Step1_LakeMetab_Processed/22GBNS1Inputs.csv", header=T)
GS2_dat <- read.csv("./Step1_LakeMetab_Processed/22GBNS2Inputs.csv", header=T)
GS3_dat <- read.csv("./Step1_LakeMetab_Processed/22GBNS3Inputs.csv", header=T)
```

Function to get gas exchange estimates

```
## Estimate gas exchange
data_K_estimate <- function(dat){

  # convert datetime to posixct
  dat$datetime <- as.POSIXct(as.character(dat$datetime), format="%Y-%m-%d %H:%M:%S")

  # subset dates
  data <- dat %>%
   # filter(year == 2021 & yday >= 182 & yday <=273)  %>%  #
    drop_na()

  # assume complete mix at 3 meters
```

```
  data$z<- c(3)

  ## rename certain columns, do.obs to do, wtr to wtemp
  colnames(data)[which(colnames(data) == "do.obs")] <- "do"
  colnames(data)[which(colnames(data) == "wtr")] <- "wtemp"

  # further clean data to avoid zmix issues
  data <- data %>%
    dplyr::group_by(year,yday) %>%
    mutate(obs = sum(!is.na(do))) %>% #identify and filter records that have < 23 hrs of data
    dplyr::ungroup() %>%
    mutate(z = ifelse(z<=0.5,.5,z))%>% #can't have zero depth zmix
    mutate(z = ifelse(z>=3,3,z)) #in littoral zone depth zmix can not be deeper than the littoral depth

  # determine data frequency obs/day should be 24
  freq <- nrlmetab::calc.freq(data$datetime)

  # estimate gas exchange
  data <- data %>%
    mutate(k600 = k.vachon.base(wnd = wspeed,lake.area = 496000)) %>% #estimate K in m/day
    mutate(kgas = k600.2.kGAS.base(k600 = k600,temperature = wtemp,gas = "O2")) %>%  #m/d
    mutate(k = (kgas/freq)/z) %>% #convert gas to T^-1
    select(-kgas,-k600,-obs)

  # Assume no DO exchange with the Atmosphere. All DO change is related to metabolism
  data_k <- data %>%
    mutate(k = ifelse(z<3,0,k))

  return(data_k)

}
```

```
# Blackwood
BS1_K <- data_K_estimate(dat = BS1_dat)
BS2_K <- data_K_estimate(dat = BS2_dat)
BS3_K <- data_K_estimate(dat = BS3_dat)

# Glenbrook
GS1_K <- data_K_estimate(dat = GS1_dat)
GS2_K <- data_K_estimate(dat = GS2_dat)
GS3_K <- data_K_estimate(dat = GS3_dat)

# quick visualization:
ggplot(data=GS1_K,aes(x=yday,y=do)) + geom_line() + facet_wrap(vars(year),scales="free_x") +
  geom_point(aes(x=yday,y=z),col="blue",size=0.2)
```

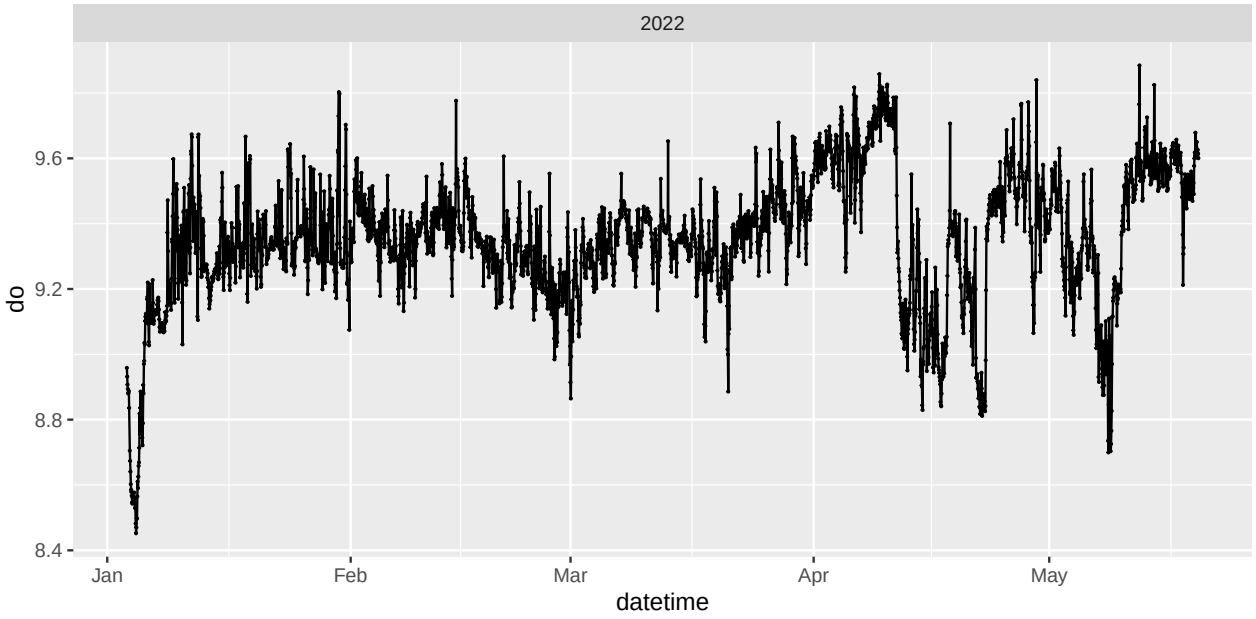


Prepare for data analysis

```
prep_data_for_LM <- function(data){

  data$hour <- lubridate::hour(data$datetime)

  sonde_prep <- data %>%
    arrange(year, yday, hour) %>%
    # for each year, create identifier for uninterrupted stretches of observations
    group_by(year) %>%
    mutate(i = ifelse(is.na(do)==T, 1, 0),
           j = c(1,abs(diff(i)))) %>%
    filter(is.na(do)==F) %>%
    mutate(series = cumsum(j)) %>%
    ungroup() %>%
    # create unique index for each series
    # remove series with fewer than 24 observations
    mutate(unique_series = year + series/length(unique(series))) %>%
    group_by(unique_series) %>%
    mutate(series_length = length(unique_series)) %>%
    ungroup() %>%
    # recreate series index and make unique index for days
    # create index for observations (for joining later)
    # replace 0 par_int with smallest non-zero value
    mutate(unique_series = as.factor(unique_series) %>% as.numeric(),
           unique_day = paste(year, yday) %>% as.factor() %>% as.numeric(),
           index = 1:length(do),
           par_int = ifelse(par_int==0,0.00001, par_int)) %>%
    select(-i, -j)

  # return missing observations for check
  sonde_check <- data %>%
    tidyr::expand(year,yday,hour) %>%
    full_join(sonde_prep) %>%
```

```
    arrange(year,yday)

  sonde_check<- na.omit(sonde_check)
  range(sonde_check$datetime)

  return(sonde_check)

}
```

```
BS1_prep <- prep_data_for_LM(BS1_K)
BS2_prep <- prep_data_for_LM(BS2_K)
BS3_prep <- prep_data_for_LM(BS3_K)

GS1_prep <- prep_data_for_LM(GS1_K)
GS2_prep <- prep_data_for_LM(GS2_K)
GS3_prep <- prep_data_for_LM(GS3_K)

## visualize
ggplot(BS1_prep,aes(x=datetime,y=do)) + geom_point(size=0.2) +
  geom_line() + facet_wrap(vars(year),scales="free_x")
```



Function to save data:

```
save.dat.fxn <- function(dat, site){
  write.csv(dat, paste("/Users/kellyloria/Desktop/LakeTahoeNS/Sondefiles/sonde_prep_",site,"_2022.csv",s
}
```

```
# save.dat.fxn(BS1_prep, "BWNS1")
# save.dat.fxn(BS2_prep, "BWNS2")
# save.dat.fxn(BS3_prep, "BWNS3")
#
# save.dat.fxn(GS1_prep, "GBNS1")
# save.dat.fxn(GS2_prep, "GBNS2")
# save.dat.fxn(GS3_prep, "GBNS3")
```

Prepare R files

```
prep_R_for_LM <- function(sonde_check, freq, site){

  # define variables in environment
  o2_freq = freq;
  o2_obs = 1000*sonde_check$do # convert to mg m^-3
  o2_eq = 1000*sonde_check$do_eq # convert to mg m^-3
  light = sonde_check$par_int
  temp = sonde_check$wtemp
  wspeed = sonde_check$wspeed
  # sch_conv = sonde_prep$sch_conv
  map_days = sonde_check$unique_day
  k = sonde_check$k

  days_per_year = array(c({sonde_check %>%
      group_by(year) %>%
      summarize(value = length(unique(unique_day)))}$value), dim = 1) #,dim = 1

  obs_per_series = array(c({sonde_check %>%
      group_by(unique_series) %>%
      summarize(value = length(unique_series))}$value))

  obs_per_day = array(c({sonde_check %>%
      group_by(unique_day) %>%
      summarize(value = length(unique_day))}$value))
  z = sonde_check$z
  n_obs = length(o2_obs)
  n_series = length(obs_per_series)
  n_days = sum(days_per_year)
  n_years = length(days_per_year)

  # export as .R
  stan_rdump(c("o2_freq","o2_obs","o2_eq","light","temp","wspeed","map_days","obs_per_series","days_per
               "obs_per_day", "z","k","n_obs","n_series","n_days","n_years"),
             file=paste("/Users/kellyloria/Desktop/LakeTahoeNS/Sondefiles/",site,"_2022_sonde_list.R",s

}
```

```
# prep_R_for_LM(sonde_check = BS1_prep, freq = 24, site = "BWNS1")
# prep_R_for_LM(sonde_check = BS2_prep, freq = 24, site = "BWNS2")
# prep_R_for_LM(sonde_check = BS3_prep, freq = 24, site = "BWNS3")
#
# prep_R_for_LM(sonde_check = GS1_prep, freq = 24, site = "GBNS1")
# prep_R_for_LM(sonde_check = GS2_prep, freq = 24, site = "GBNS2")
# prep_R_for_LM(sonde_check = GS3_prep, freq = 24, site = "GBNS3")
```