Observed Data Prep for Model Analysis

PAZ

31/01/2018

Purpose

Generate BEACH calibration data with:

- **groupAlteck2016_R.csv** (Book 04)

Lab parameters and field constants

```
if (MAC) {
  if (WIN){
    path = file.path("C:/Users/DayTimeChunks/Documents/PhD/HydrologicalMonitoring")

  } else {
    # path = file.path("/Users/DayTightChunks/Documents/PhD/HydrologicalMonitoring")
    path = file.path("/Users/DayTightChunks/Documents/PhD/hydrological-monitoring")
    time = read.csv2("/Users/DayTightChunks/Documents/PhD/Models/phd-model-master/Analysis/Data/Time.csv
    time$DayMoYr = as.POSIXct(strptime(time$Date, "%d/%m/%Y", tz="EST"))
  }
} else {
  path = file.path("D:/Documents/these_pablo/Alteckendorf2016/HydrologicalMonitoring")
  time = read.csv2("D:/Documents/these_pablo/Models/BEACH2016/Analysis/Data/Time.csv")
  time$DayMoYr = as.POSIXct(strptime(time$Date, "%d/%m/%Y", tz="EST"))
}
source(file.path(path, "global.R"))
```

Packages

```
# Plotting functions
library("scales")
library("tidyr")
library("dplyr")
library("reshape")
library("zoo") # na.approx()
```

Working directory

```
# setwd("D:/Documents/these_pablo/Alteckendorf2016/R")

# MAC
# setwd("/Users/DayTightChunks/Documents/PhD/Routput/Alteck/R")
```

```
# Mac-WIN
# setwd("C:/Users/DayTightChunks/Documents/Models/pesti-beach16/Analysis/Data")
getwd()
```

```
## [1] "/Users/DayTightChunks/Documents/PhD/Models/phd-model-master/Analysis/Data"
```

Discharge & Response Variables (with markers)

- Ignoring δ in filters (for now)

```
q = read.csv2(file.path(path, "Data/groupAlteck2016_R.csv"))
q$Vol.L = q$Vol2min * 1000

q = q[ , c("Date", "DateCheck", "Q.HW1", "DayMoYr", "Vol.L", "Vol2min", "sampleQ", "Type", "SubWeeks",
names(q)
```

```
##  [1] "Date"      "DateCheck" "Q.HW1"     "DayMoYr"   "Vol.L"
##  [6] "Vol2min"   "sampleQ"   "Type"      "SubWeeks"  "Weeks"
## [11] "WeekNo"
```

```
mark = read.csv(file.path(path, "Data/MarkerResponse_R05.csv"))
mark = mark[, c("WeekSubWeek",
                # "AveDischarge.m3.h", "Volume.m3",  "Sampled.Hrs",
                # "Sampled",
                "Conc.mug.L" , "Conc.SD",
                # "Vol.SPE.L", "Conc.in500uL",
                "OXA_mean", "OXA_SD", "ESA_mean", "ESA_SD",
                "N.x", "diss.d13C", "SD.d13C",
                "MES.mg.L", "MES.sd", "MO.mg.L", "Conc.Solids.mug.gMES", "Conc.Solids.ug.gMES.SD" #,
                #"N.y",  "filt.d13C",  "filt.SD.d13C" #,
                #"DD13C.diss", "DD13C.filt"
                )]
names(mark)
```

```
##  [1] "WeekSubWeek"            "Conc.mug.L"
##  [3] "Conc.SD"                "OXA_mean"
##  [5] "OXA_SD"                 "ESA_mean"
##  [7] "ESA_SD"                 "N.x"
##  [9] "diss.d13C"              "SD.d13C"
## [11] "MES.mg.L"               "MES.sd"
## [13] "MO.mg.L"                "Conc.Solids.mug.gMES"
## [15] "Conc.Solids.ug.gMES.SD"
```

```
# Delete repeated W6 observation, or with NA in week markers
# mark = mark[mark$WeekSubWeek != as.character("W6-3j7") & !is.na(mark$WeekSubWeek), ]



q$Date = as.POSIXct(strptime(q$DateCheck, "%d/%m/%Y %H:%M", tz="EST"))
q$DayMoYr = as.POSIXct(strptime(q$DateCheck, "%d/%m/%Y", tz="EST"))
q$Min = 2.0

CHECKO = F
if (CHECKO){
```

```
  sum(is.na(q$Date))
  naDates = q[is.na(q$Date == TRUE),]

  duplicateAlteck <- q[duplicated(q$DateCheck),]
  head(duplicateAlteck)
}
```

Prepare Volume Discharged Time Series (TSS)

```
qDay <- q %>%
  group_by(DayMoYr) %>%
  dplyr::summarize(Volday.L = sum(Vol.L))

qDay$VolTot.m3 = round(qDay$Volday.L/10^3, 3)

qTime = merge(time, qDay, by = "DayMoYr", all = T)

qTime_cal = subset(qTime, !is.na(VolTot.m3))
qTime_cal = qTime_cal[, c("Jdays", "VolTot.m3")]
names(qTime_cal) = c("Jdays", "Qm3")

mean(qTime$VolTot.m3, na.rm = T)
```

```
## [1] 259.9841
```

```
sd(qTime$VolTot.m3, na.rm = T)
```

```
## [1] 453.5581
```

```
Volm3_tss = qTime[,c("Jdays", "VolTot.m3")]

Volm3_tss$VolTot.m3 = ifelse(is.na(Volm3_tss$VolTot.m3), -1.0, Volm3_tss$VolTot.m3)

if (F) {
  write.table(Volm3_tss, "BEACH_R/q_obs_m3day.tss", sep="\t", row.names = F, col.names = F)
  write.table(qTime_cal, "BEACH_R/q_out_cal.tss", sep="\t", row.names = F, col.names = T) # m3day
}

if (F) {

  ## Convert m3.h -> m3
  qDay <- q %>%
    group_by(DayMoYr) %>%
    dplyr::summarize(Q.m3 = sum(Vol2min))

  qDay$Q.mm = (qDay$Q.m3/catchment_area)*10^3

  qDay$time = seq.int(nrow(qDay))

  # Qm3/day
  DischQm3_tss = qDay[,c("time", "Q.m3")]
  write.table(DischQm3_tss, "BEACH_R/disch_m3day.tss", sep="\t", row.names = F, col.names = F)
```

```
  # Qmm/day
  DischQmm_tss = qDay[,c("time", "Q.mm")]
  write.table(DischQmm_tss, "BEACH_R/disch_mmday.tss", sep="\t", row.names = F)


}
```

New observed outlet calculations

New calculation of observed outlet samples, introducing a weight to the sample based on the volume discharge associated to the sub-sample.

Steps:

- Compute the total volume of each day associated to each sub-sample

This is needed to obtain the proportion of volume contributing to each sub sample on any given day.

- Compute the sum of the total volumes above, i.e. the total discharged volume assocaited to the sub sample.

We will use this to obtain a weight on each sub-sample.

- Compute the weigh (i.e. fraction of the discharged volume per sub sample)

```
# Step 1
q2 <- q %>%
    group_by(DayMoYr, SubWeeks) %>%
    dplyr::summarize(TotVol.L = sum(Vol.L),
                     SmpHrs = sum(Min)/60)
# Step 2
bal = q %>%
  group_by(SubWeeks) %>% # Sum of total volumes by sub-sample
  dplyr::summarize(VolBalSmp = sum(Vol.L))

q2 = merge(q2, bal, by = "SubWeeks", all = T)

# Step 3
q2$weigh = q2$TotVol.L/q2$VolBalSmp

qm2 = merge(q2, mark, by.x = "SubWeeks", by.y = "WeekSubWeek", all = T)
qm2 = subset(qm2, !is.na(Conc.mug.L))

# Step 4 - Filter out samples with less than 90% of the day sampled (> 21.5 hrs).
qm2_90 = subset(qm2, SmpHrs > 21.5)

# Step 5  - Get the loads of each day
qm2_90$smloads.g = ((qm2_90$Conc.mug.L)/10**6) * qm2_90$TotVol.L

# Check all duplicated days with data
allDup2 = qm2_90 %>%
  group_by(DayMoYr) %>%
  filter(n()>1 )  # NO data, no duplicates


# Step 6 megre with Jdays
```

```
qmDaily = merge(time, qm2_90, by = "DayMoYr", all = T)


loads_g_cal = qmDaily[, c("Jdays", "smloads.g", "weigh")]
conc_out_cal = qmDaily[, c("Jdays", "Conc.mug.L", "weigh")]
names(conc_out_cal) = c("Jdays", "ug.L", "weigh")
d13c_out_cal = qmDaily[, c("Jdays", "diss.d13C", "weigh")]
names(d13c_out_cal) = c("Jdays", "d13C", "weigh")

if (F){
  if (F) {
  # write.csv(qmBlk, "qmBlk_R.csv", row.names = F) # , sep = ";", dec = ".")
  write.table(loads_g_cal, "BEACH_R/lds_out_cal.tss", sep="\t", row.names = F, col.names = T)
  write.table(conc_out_cal, "BEACH_R/conc_out_cal.tss", sep="\t", row.names = F, col.names = T)
  write.table(d13c_out_cal, "BEACH_R/d13c_out_cal.tss", sep="\t", row.names = F, col.names = T)
}
}
```