BEACH Hydro Data Preparation

PAZ

31/01/2018

Purpose

Lab parameters and field constants

```
source("global.R")
```

Packages

```
# Plotting functions
library("scales")
library("tidyr")
library("dplyr")
library("reshape")
```

Working directory

```
# setwd("D:/Documents/these_pablo/Alteckendorf2016/R")
# setwd("/Users/DayTightChunks/Documents/PhD/Routput/Alteck/R")
getwd()
```

```
## [1] "/Users/DayTightChunks/Documents/PhD/HydrologicalMonitoring"
```

Rainfall

```
rain = read.csv2("Data/sixMinutePluvioAlteck2016.csv", header = F)
head(rain)
```

```
##                 V1  V2
## 1 25/03/2016 05:38 0.2
## 2 25/03/2016 05:44 0.0
## 3 25/03/2016 05:50 0.0
## 4 25/03/2016 05:56 0.0
## 5 25/03/2016 06:02 0.0
## 6 25/03/2016 06:08 0.0
```

```
rain$V1 <- as.character(rain$V1)
rain$Date = as.POSIXct(strptime(rain$V1,
                                "%d/%m/%Y %H:%M", tz="EST") )

rain$DayMoYr = as.POSIXct(strptime(rain$V1,
```

```
                                    "%d/%m/%Y", tz="EST")  )

# Check number of NA values
CHECKO = FALSE
if (CHECKO){
  sum(is.na(rain$Date))
  naDates = rain[is.na(rain$Date == TRUE),]
}

rainDay <- rain %>%
  group_by(DayMoYr) %>%
  dplyr::summarize(Rain.mm = sum(V2))
```

Prepare Rainfall Time Series (TSS)

```
rainDay$time = seq.int(nrow(rainDay))
rain_tss = rainDay[,c("time", "Rain.mm")]
#rain_tss = rbind(c("2016-03-25 to 2016-07-11", NA), rain_tss)
write.table(rain_tss, "Data/BEACH/rain_mmday.tss", sep="\t", row.names = F)
```

Analyse Rainfall Monthly Values

```
rainDay$Month <-
  ifelse(rainDay$DayMoYr >= as.POSIXct("2016-03-24 00:30:00", tz = "EST") &
           rainDay$DayMoYr < as.POSIXct("2016-04-01 00:00:00", tz = "EST"), "March",
         ifelse(rainDay$DayMoYr >= as.POSIXct("2016-04-01 00:00:00", tz = "EST") &
                  rainDay$DayMoYr < as.POSIXct("2016-05-01 00:00:00", tz = "EST"), "April",
                ifelse(rainDay$DayMoYr >= as.POSIXct("2016-05-01 00:00:00", tz = "EST") &
                         rainDay$DayMoYr < as.POSIXct("2016-06-01 00:00:00", tz = "EST"), "May",
                       ifelse(rainDay$DayMoYr >= as.POSIXct("2016-06-01 00:00:00", tz = "EST") &
                                rainDay$DayMoYr < as.POSIXct("2016-07-01 00:00:00", tz = "EST"), "June"
                              )
                       )
                )
         )

rainDay$Wet = ifelse(rainDay$Rain.mm > 0, 1, 0)
rainDay$Dry = ifelse(rainDay$Rain.mm == 0, 1, 0)

rainSumm <- rainDay %>%
  group_by(Month) %>%
  dplyr::summarize(WetDays = sum(Wet),
                   DryDays = sum(Dry),
                   MeanP = mean(Rain.mm),
                   StdP = sd(Rain.mm),
                   TotP = sum(Rain.mm))

rainSumm$Prct = rainSumm$WetDays/(rainSumm$WetDays+rainSumm$DryDays)
```

Discharge

```
q = read.csv2("Data/hydroAlteck2016_smooth_R.csv")
head(q)
```

```
##                  Date        DateCheck Q.m3Hrs   Qna Qapprox Qinterp
## 1 2016-03-25 00:04:00 25/03/2016 00:04   1.192 1.192   1.192   1.192
## 2 2016-03-25 00:06:00 25/03/2016 00:06   1.212 1.212   1.212   1.212
## 3 2016-03-25 00:08:00 25/03/2016 00:08   1.195 1.195   1.195   1.195
## 4 2016-03-25 00:10:00 25/03/2016 00:10   1.219 1.219   1.219   1.219
## 5 2016-03-25 00:12:00 25/03/2016 00:12   1.217 1.217   1.217   1.217
## 6 2016-03-25 00:14:00 25/03/2016 00:14   1.230 1.230   1.230   1.230
##      Q.HW1             Q.HW2
## 1 1.248600             1.182
## 2 1.237280 1.15424267729696
## 3 1.232224 1.17062590682503
## 4 1.224779 1.15615409458726
## 5 1.223623 1.17724053690379
## 6 1.222299 1.17698892559366
```

```
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"))

CHECKO = F
if (CHECKO){
  sum(is.na(q$Date))
naDates = q[is.na(q$Date == TRUE),]

duplicateAlteck <- q[duplicated(q$DateCheck),]
head(duplicateAlteck)
}
## Convert m3.h -> m3
q$Vol2min <- q$Q.HW1*2/60

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

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

Prepare Discharge Time Series (TSS)

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

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

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

Analyse Discharge Monthly Values

```
qDay$Month <-
  ifelse(qDay$DayMoYr >= as.POSIXct("2016-03-24 00:30:00", tz = "EST") &
           qDay$DayMoYr < as.POSIXct("2016-04-01 00:00:00", tz = "EST"), "March",
         ifelse(qDay$DayMoYr >= as.POSIXct("2016-04-01 00:00:00", tz = "EST") &
                  qDay$DayMoYr < as.POSIXct("2016-05-01 00:00:00", tz = "EST"), "April",
                ifelse(qDay$DayMoYr >= as.POSIXct("2016-05-01 00:00:00", tz = "EST") &
                         qDay$DayMoYr < as.POSIXct("2016-06-01 00:00:00", tz = "EST"), "May",
                       ifelse(qDay$DayMoYr >= as.POSIXct("2016-06-01 00:00:00", tz = "EST") &
                                qDay$DayMoYr < as.POSIXct("2016-07-01 00:00:00", tz = "EST"), "June", "
                              )
                       )
                )
         )

dischSumm <- qDay %>%
  group_by(Month) %>%
  dplyr::summarize(MeanQmm = mean(Q.mm),
                   SdevQmm = sd(Q.mm),
                   MeanQm3 = mean(Q.m3))
```

EvapoTranspiration (ETP)

Note, these calculations are done in the excel file for now: ZornDaily_Oct2015toJune2017.xls

```
etp = read.csv2("Data/ZornDaily.csv", sep = ",")
head(etp)
```

```
##      POSTE   DATE Rainfall Mean.temperature..24.hourly.values. ETPGRILLE
## 1 67516001 7/1/15        0                                 26.5       7.6
## 2 67516001 7/2/15        0                                 28.6       7.2
## 3 67516001 7/3/15        0                                 29.2         8
## 4 67516001 7/4/15        0                                 29.5       7.3
## 5 67516001 7/5/15        0                                 29.3       8.9
## 6 67516001 7/6/15        0                                 23.6       6.2
```

```
etp$DATE = as.character(etp$DATE)
etp$DayMoYr = as.POSIXct(strptime(etp$DATE, "%m/%d/%y", tz="EST"))
```