Sub-weekly Discharge - Alteck 2016, Part I

PAZ

25 octobre 2016

Purpose

This file computes average discharge and total volume discharged per sub-week (sampled and non-sampled).

Imports:

- **Alteck2016Debit__smooth__R.csv**

Generates:

- **WeeklyHydro__R.csv**

which is a file with discharge data summarized by sub-week.

Columns in **WeeklyHydro__R.csv** include:

1. Week-SubWeek ID
2. Average Discharge
3. Volumne discharged
4. Elapsed hours per subweek
5. Marker indicating whether the subweek was sampled or not.

To see the variables in this file see the end of this document.

Required R-packages:

```
# Date-time functions
library(chron)
```

Working directory

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

```
## [1] "D:/Documents/these_pablo/Alteckendorf2016/HydrologicalMonitoring"
```

Import smooth (and sampler merged) discharge data

```
debitAlt = read.csv2("Data/hydroAlteck2016_R.csv", header = TRUE)
head(debitAlt)
```

```
##                  Date      DateCheck.S        DateCheck Q.m3Hrs   Qna
## 1 2016-03-25 00:04:00 25/03/2016 00:04 25/03/2016 00:04   1.192 1.192
## 2 2016-03-25 00:06:00 25/03/2016 00:06 25/03/2016 00:06   1.212 1.212
## 3 2016-03-25 00:08:00 25/03/2016 00:08 25/03/2016 00:08   1.195 1.195
## 4 2016-03-25 00:10:00 25/03/2016 00:10 25/03/2016 00:10   1.219 1.219
## 5 2016-03-25 00:12:00 25/03/2016 00:12 25/03/2016 00:12   1.217 1.217
## 6 2016-03-25 00:14:00 25/03/2016 00:14 25/03/2016 00:14   1.230 1.230
##   Qapprox Qinterp    Q.HW1            Q.HW2    DayMoYr    Vol2min sampleQ
## 1   1.192   1.192 1.248600            1.182 2016-03-25 0.04162000      NA
## 2   1.212   1.212 1.237280 1.15424267729696 2016-03-25 0.04124267      NA
## 3   1.195   1.195 1.232224 1.17062590682503 2016-03-25 0.04107413      NA
## 4   1.219   1.219 1.224779 1.15615409458726 2016-03-25 0.04082597      NA
## 5   1.217   1.217 1.223623 1.17724053690379 2016-03-25 0.04078745      NA
## 6   1.230   1.230 1.222299 1.17698892559366 2016-03-25 0.04074329      NA
##        Type Rain.mm
## 1 Discharge       0
## 2 Discharge       0
## 3 Discharge       0
## 4 Discharge       0
## 5 Discharge       0
## 6 Discharge       0
```

```r
tail(debitAlt)
```

```
##                       Date      DateCheck.S        DateCheck Q.m3Hrs   Qna
## 78754 2016-07-12 10:10:00 12/07/2016 10:10 12/07/2016 10:10   2.224 2.224
## 78755 2016-07-12 10:12:00 12/07/2016 10:12 12/07/2016 10:12   2.260 2.260
## 78756 2016-07-12 10:14:00 12/07/2016 10:14 12/07/2016 10:14   2.223 2.223
## 78757 2016-07-12 10:16:00 12/07/2016 10:16 12/07/2016 10:16   2.101 2.101
## 78758 2016-07-12 10:18:00 12/07/2016 10:18 12/07/2016 10:18   2.059 2.059
## 78759 2016-07-12 10:20:00 12/07/2016 10:20 12/07/2016 10:20   2.007 2.007
##       Qapprox Qinterp    Q.HW1            Q.HW2    DayMoYr    Vol2min
## 78754   2.224   2.224 2.145201 2.09646551274142 2016-07-12 0.07150671
## 78755   2.260   2.260 2.160961 2.17734152373787 2016-07-12 0.07203204
## 78756   2.223   2.223 2.180769 2.21674008593281 2016-07-12 0.07269230
## 78757   2.101   2.101 2.189215 2.18552592249578 2016-07-12 0.07297384
## 78758   2.059   2.059 2.171572 2.07040134001013 2016-07-12 0.07238574
## 78759   2.007   2.007 2.149058 2.02286344936347 2016-07-12 0.07163526
##       sampleQ      Type Rain.mm
## 78754      NA Discharge       0
## 78755      NA Discharge       0
## 78756      NA Discharge       0
## 78757      NA Discharge       0
## 78758      NA Discharge       0
## 78759      NA Discharge       0
```

Convert Date column to a readable Date-time object

```r
class(debitAlt$Date)
```

```
## [1] "factor"
```

```r
debitAlt$Date = as.POSIXct(strptime(debitAlt$Date, "%Y-%m-%d %H:%M", tz="EST"))
sum(is.na(debitAlt$Date))
```

```
## [1] 0
```

Choose the variables to report as sub-weekly summaries

1. Discharge (m3/h) variable to calculate sub-weekly averages
2. Rainfall (mm) taken every 2 minutes, will become cummulative for the sub-weekly sample

```
# Remove all unecessary columns
debitAlt = debitAlt[,c("Date", "Q.HW1", "Rain.mm")]
head(debitAlt)
```

```
##                  Date    Q.HW1 Rain.mm
## 1 2016-03-25 00:04:00 1.248600       0
## 2 2016-03-25 00:06:00 1.237280       0
## 3 2016-03-25 00:08:00 1.232224       0
## 4 2016-03-25 00:10:00 1.224779       0
## 5 2016-03-25 00:12:00 1.223623       0
## 6 2016-03-25 00:14:00 1.222299       0
```

Define the sub-week sample dates

During the 2016 campaign some periods where not sampled because the automatic sampler either experienced malfunction or the capacity was reached before intervention. Interventioned took place on a weekly basis.

1. Extract subsets where sampling was conducted.
2. Extract subsets where sampling was not conducted.

```
# Define the Weekly discharge subsets
W00 <- subset(debitAlt,
              Date < as.POSIXct('2016-03-25 12:04:00' , tz="EST") ) # Not sampled

W01 <- subset(debitAlt,
              Date >= as.POSIXct('2016-03-25 12:04:00' , tz="EST") &
                Date < as.POSIXct('2016-03-28 22:37:00' , tz="EST"))

W02 <- subset(debitAlt,
              Date >= as.POSIXct('2016-03-28 22:37:00' , tz="EST") &
                Date <= as.POSIXct('2016-03-30 12:17:00' , tz="EST")) # Not sampled

W11 <- subset(debitAlt,
              Date >= as.POSIXct('2016-03-30 12:17:00' , tz="EST") &
                Date <= as.POSIXct('2016-03-31 15:35:00' , tz="EST"))
W12 <- subset(debitAlt,
              Date >= as.POSIXct('2016-03-31 15:35:00' , tz="EST") &
                Date <= as.POSIXct('2016-04-01 14:45:00' , tz="EST"))
# W13
W13 <- subset(debitAlt,
              Date >= as.POSIXct('2016-04-01 14:45:00' , tz="EST") &
                Date <= as.POSIXct('2016-04-05 15:07:00' , tz="EST")) # Not sampled

W21 <- subset(debitAlt,
              Date >= as.POSIXct('2016-04-05 15:07:00' , tz="EST") &
                Date <= as.POSIXct('2016-04-06 14:51:00' , tz="EST"))
W22 <- subset(debitAlt,
              Date >= as.POSIXct('2016-04-06 14:51:00' , tz="EST") &
                Date < as.POSIXct('2016-04-08 00:38:00' , tz="EST"))
```

```
# W23 not sampled
W23 <- subset(debitAlt,
              Date >= as.POSIXct('2016-04-08 00:38:00' , tz="EST") &
                Date < as.POSIXct('2016-04-14 13:51:00' , tz="EST"))

W31 <- subset(debitAlt,
              Date >= as.POSIXct('2016-04-14 13:51:00' , tz="EST") &
                Date < as.POSIXct('2016-04-16 18:32:00' , tz="EST"))

W32 <- subset(debitAlt,
              Date >= as.POSIXct('2016-04-16 18:32:00' , tz="EST") &
                Date < as.POSIXct('2016-04-17 09:02:00' , tz="EST"))
###
W32.1 <- subset(debitAlt,
                Date >= as.POSIXct('2016-04-17 09:02:00' , tz="EST") &
                  Date < as.POSIXct('2016-04-18 20:30:00' , tz="EST")) #  Not sampled

W33 <- subset(debitAlt,
              Date >= as.POSIXct('2016-04-18 20:30:00' , tz="EST") &
                Date < as.POSIXct('2016-04-21 09:11:00' , tz="EST"))

W41 <- subset(debitAlt,
              Date >= as.POSIXct('2016-04-21 09:11:00' , tz="EST") &
                Date < as.POSIXct('2016-04-23 06:37:00' , tz="EST"))

W42 <- subset(debitAlt,
              Date >= as.POSIXct('2016-04-23 06:37:00' , tz="EST") &
                Date < as.POSIXct('2016-04-26 11:50:00' , tz="EST")) # Not sampled

W51 <- subset(debitAlt,
              Date >= as.POSIXct('2016-04-26 11:50:00' , tz="EST") &
                Date < as.POSIXct('2016-05-01 10:46:00' , tz="EST"))

W52 <- subset(debitAlt,
              Date >= as.POSIXct('2016-05-01 10:46:00' , tz="EST") &
                Date < as.POSIXct('2016-05-03 12:02:00' , tz="EST"))

##
W53 <- subset(debitAlt,
              Date >= as.POSIXct('2016-05-03 12:02:00' , tz="EST") &
                Date < as.POSIXct('2016-05-03 13:09:00' , tz="EST")) # Not sampled

W61 <- subset(debitAlt,
              Date >= as.POSIXct('2016-05-03 13:09:00' , tz="EST") &
                Date < as.POSIXct('2016-05-10 00:05:00' , tz="EST"))

W62 <- subset(debitAlt,
              Date >= as.POSIXct('2016-05-10 00:05:00' , tz="EST") &
                Date < as.POSIXct('2016-05-12 06:33:00' , tz="EST"))
W63 <- subset(debitAlt,
              Date >= as.POSIXct('2016-05-12 06:33:00' , tz="EST") &
                Date < as.POSIXct('2016-05-12 09:12:00' , tz="EST"))
W64 <- subset(debitAlt,
              Date >= as.POSIXct('2016-05-12 09:12:00' , tz="EST") &
                Date < as.POSIXct('2016-05-12 12:52:00' , tz="EST"))

##
W65 <- subset(debitAlt,
              Date >= as.POSIXct('2016-05-12 12:52:00' , tz="EST") &
                Date < as.POSIXct('2016-05-13 12:05:00' , tz="EST")) # Not sampled

W71 <- subset(debitAlt,
              Date >= as.POSIXct('2016-05-13 12:05:00' , tz="EST") &
                Date < as.POSIXct('2016-05-16 15:11:00' , tz="EST"))
##
W72 <- subset(debitAlt,
              Date >= as.POSIXct('2016-05-16 15:11:00' , tz="EST") &
                Date <= as.POSIXct('2016-05-17 09:16:00' , tz="EST")) # Not sampled

W81 <- subset(debitAlt,
              Date >= as.POSIXct('2016-05-17 09:16:00' , tz="EST") &
                Date <= as.POSIXct('2016-05-23 18:02:00' , tz="EST"))
##
W82 <- subset(debitAlt,
              Date >= as.POSIXct('2016-05-23 18:02:00' , tz="EST") &
                Date <= as.POSIXct('2016-05-24 12:00:00' , tz="EST")) # Not sampled

W91 <- subset(debitAlt,
              Date >= as.POSIXct('2016-05-24 12:00:00' , tz="EST") &
                Date <= as.POSIXct('2016-05-29 12:09:00' , tz="EST"))
W92 <- subset(debitAlt,
              Date >= as.POSIXct('2016-05-29 12:09:00' , tz="EST") &
                Date < as.POSIXct('2016-05-30 05:48:00' , tz="EST"))
W93 <- subset(debitAlt,
              Date >= as.POSIXct('2016-05-30 05:48:00' , tz="EST") &
                Date < as.POSIXct('2016-05-30 12:11:00' , tz="EST"))
W94 <- subset(debitAlt,
              Date >= as.POSIXct('2016-05-30 12:11:00' , tz="EST") &
                Date < as.POSIXct('2016-05-30 17:28:00' , tz="EST"))
##
W95 <- subset(debitAlt,
              Date >= as.POSIXct('2016-05-30 17:28:00' , tz="EST") &
                Date < as.POSIXct('2016-05-31 12:00:00' , tz="EST")) # Not sampled

W101 <- subset(debitAlt,
               Date >= as.POSIXct('2016-05-31 12:00:00' , tz="EST") &
                 Date < as.POSIXct('2016-06-02 12:57:00' , tz="EST"))
W102 <- subset(debitAlt,
               Date >= as.POSIXct('2016-06-02 12:57:00' , tz="EST") &
                 Date < as.POSIXct('2016-06-03 12:05:00' , tz="EST"))
W103 <- subset(debitAlt,
               Date >= as.POSIXct('2016-06-03 12:05:00' , tz="EST") &
                 Date < as.POSIXct('2016-06-04 08:35:00' , tz="EST"))
W104 <- subset(debitAlt,
               Date >= as.POSIXct('2016-06-04 08:35:00' , tz="EST") &
                 Date < as.POSIXct('2016-06-04 11:00:00' , tz="EST"))
```

```r
W105 <- subset(debitAlt,
               Date >= as.POSIXct('2016-06-04 11:00:00' , tz="EST") &
                 Date < as.POSIXct('2016-06-04 15:31:00' , tz="EST"))
#
W106 <- subset(debitAlt,
               Date >= as.POSIXct('2016-06-04 15:31:00' , tz="EST") &
                 Date <= as.POSIXct('2016-06-07 12:00:00' , tz="EST")) # Not sampled

W111 <- subset(debitAlt,
               Date >= as.POSIXct('2016-06-07 12:00:00' , tz="EST") &
                 Date <= as.POSIXct('2016-06-10 05:25:00' , tz="EST"))
W112 <- subset(debitAlt,
               Date > as.POSIXct('2016-06-10 05:25:00' , tz="EST") &
                 Date < as.POSIXct('2016-06-14 12:34:00' , tz="EST"))
W113 <- subset(debitAlt,
               Date >= as.POSIXct('2016-06-14 12:34:00' , tz="EST") &
                 Date < as.POSIXct('2016-06-14 13:06:00' , tz="EST"))
W121 <- subset(debitAlt,
               Date > as.POSIXct('2016-06-14 13:06:00' , tz="EST") &
                 Date < as.POSIXct('2016-06-15 08:14:00' , tz="EST"))
W122 <- subset(debitAlt,
               Date > as.POSIXct('2016-06-15 08:14:00' , tz="EST") &
                 Date < as.POSIXct('2016-06-16 08:21:00' , tz="EST"))
W123 <- subset(debitAlt,
               Date > as.POSIXct('2016-06-16 08:21:00' , tz="EST") &
                 Date < as.POSIXct('2016-06-17 00:49:00' , tz="EST"))

W124 <- subset(debitAlt,
               Date > as.POSIXct('2016-06-17 00:49:00' , tz="EST") &
                 Date <= as.POSIXct('2016-06-17 11:05:00' , tz="EST"))
#
W125 <- subset(debitAlt,
               Date > as.POSIXct('2016-06-17 00:49:00' , tz="EST") &
                 Date < as.POSIXct('2016-06-21 12:00:00' , tz="EST")) # Not sampled

W131 <- subset(debitAlt,
               Date >= as.POSIXct('2016-06-21 12:00:00' , tz="EST") &
                 Date < as.POSIXct('2016-06-24 14:51:00' , tz="EST"))
W132 <- subset(debitAlt,
               Date > as.POSIXct('2016-06-24 14:51:00' , tz="EST") &
                 Date < as.POSIXct('2016-06-25 07:49:00' , tz="EST"))
W133 <- subset(debitAlt,
               Date > as.POSIXct('2016-06-25 07:49:00' , tz="EST") &
                 Date < as.POSIXct('2016-06-28 08:55:00' , tz="EST"))
W141 <- subset(debitAlt,
               Date > as.POSIXct('2016-06-28 08:55:00' , tz="EST") &
                 Date <= as.POSIXct('2016-07-04 14:41:00' , tz="EST"))

W151 <- subset(debitAlt,
               Date > as.POSIXct('2016-07-04 14:41:00' , tz="EST") &
                 Date <= as.POSIXct('2016-07-12 10:20:00' , tz="EST"))
```

Calculate sampled & non-sampled hrs.

1. Create a list made up of each subset data.frame

```
# Create a list of matrices

weeks = list(W01,
             W11,
             W12,
             W21, W22,
             W31, W32, W33,
             W41,
             W51, W52,
             W61, W62, W63, W64,
             W71,
             W81,
             W91, W92, W93, W94,
             W101, W102, W103, W104, W105,
             W111, W112, W113,
             W121, W122, W123, W124,
             W131, W132, W133,
             W141,
             W151)

weeksNS = list(W00, W02, W13, W21, W32.1, W42, W53, W65, W72, W82, W95, W106, W125)
```

2. Create a function to calculate the time difference between the last and first date-entry of each subset.

```
hoursInWeek = function(x){
  as.numeric(difftime(x[nrow(x),1], x[1,1], units = "hours"), units = "hours")
}
```

Here "x" is a matrix (i.e. data.frame object) with number of rows = nrow. The *difftime* function uses the last index-value (t2) - first index-value (t1)

The *difftime* syntax: matrix[last_row, first_column], matrix[first_row, first_column] -> t2 - t1

3. Get total hours for each subset (data.frame) with the *sapply* function

```
sampledHrsList = sapply(weeks, hoursInWeek)
nonSampledHrsList = sapply(weeksNS, hoursInWeek)

nonSampledHrsList
```

```
##  [1]  11.96667  37.63333  96.33333  23.70000  35.43333  77.16667   1.10000
##  [8]  23.20000  18.06667  17.96667  18.50000  68.46667 107.13333
```

Here, the function *sapply* returns a list of outputs based on the created function we pass to it. The syntax: sapply(object, function to apply on each object).

4. Get volume discharged.

Same as above, we create first a function and then use *sapply*. Note that the discharge data in $[m^3h^{-1}]$ has an interval of 2 minutes. So we multiply each entry of the discharge column by 2 and convert to hrs.

```
# m^3/h * 2 min * 1h/60min = m3
volMinute = function(x) {
  (x[,2] * 2/60)
}
```

```
volMinuteList = sapply(weeks, volMinute)
volMinuteListNS = sapply(weeksNS, volMinute)
```

5. Get total rainfall per sub-week

```
rainTot = function(x) {
  sum(x[,3])
}

rainList = sapply(weeks, rainTot)
rainListNS = sapply(weeksNS, rainTot)
```

Check results

```
# first matrix in the weeks list (note: matix has two rows)
# weeks[[1]]

# access the first matrix in the list, the second element in THAT list.
volMinuteList[[1]][[2]]
```

```
## [1] 0.03755018
```

```
# Cumm rain per sub-sample
rainList
```

```
##  [1]  7.6 16.8  6.0  5.4  0.8 20.0 12.4  0.0  0.0 30.4  9.2  0.4 24.4 10.8
## [15]  2.0  9.4 15.6 11.8 18.4 13.2  0.6  5.8  9.6 10.6  6.6  4.0 10.4 27.4
## [29]  0.4  6.2  2.4 18.6  0.6  0.4 24.0  2.6  1.0  7.2
```

6. Get total volume for each subset (i.e. each subweek)

```
total = function(x){
  sum(x)
}

volumesList = sapply(volMinuteList, total)
volumesListNS = sapply(volMinuteListNS, total)

# Check volumnes for the non-sample list:
volumesListNS
```

```
##  [1]   14.41714   48.34827  877.37700  346.15602 1786.10736  486.76267
##  [7]   13.33031 2627.31247  160.76996   78.12190  887.97251 3395.52590
## [13] 3553.06430
```

7. Create an average discharge and an average rainfall intensity list for each entry result above

```
AveDischarge = volumesList/sampledHrsList
AveDischargeNS = volumesListNS/nonSampledHrsList

RainIntensity = rainList/sampledHrsList
RainIntensityNS = rainListNS/nonSampledHrsList
```

8. Put all data together into a new data.frame structure.

```
# Sampled data frame

ID = c("W0-1", "W1-1", "W1-2",
       "W2-1", "W2-2",
       "W3-1", "W3-2", "W3-3", "W4-1",
       "W5-1", "W5-2",
       "W6-1", "W6-2", "W6-3", "W6-4",
       "W7-1",
       "W8-1",
       "W9-1", "W9-2", "W9-3", "W9-4",
       "W10-1", "W10-2", "W10-3", "W10-4", "W10-5",
       "W11-1", "W11-2", "W11-3",
       "W12-1", "W12-2", "W12-3", "W12-4",
       "W13-1", "W13-2", "W13-3",
       "W14-1",
       "W15-1")

Sampled = rep("Sampled", length(ID))


WeeklyDischarge = as.data.frame(ID)
WeeklyDischarge$AveDischarge.m3.h = AveDischarge
WeeklyDischarge$Volume.m3 = volumesList
WeeklyDischarge$Sampled.Hrs = sampledHrsList
WeeklyDischarge$Sampled = Sampled
WeeklyDischarge$CumRain.mm = rainList
WeeklyDischarge$RainInt.mmhr = RainIntensity

# Not sampled data frame

IDns = c("W0-0x", "W0-2x", "W1-3x", "W2-3x", "W3-2.1x", "W4-2x",
         "W5-3x", "W6-5x", "W7-2x", "W8-2x",
         "W9-5x", "W10-6x", "W12-5x")

SampledNS = rep("Not Sampled", length(IDns))

WeeklyDischargeNS = as.data.frame(IDns)
WeeklyDischargeNS$AveDischarge.m3.h = AveDischargeNS
WeeklyDischargeNS$Volume.m3 = volumesListNS
WeeklyDischargeNS$Sampled.Hrs = nonSampledHrsList
WeeklyDischargeNS$Sampled = SampledNS
WeeklyDischargeNS$CumRain.mm = rainListNS
WeeklyDischargeNS$RainInt.mmhr = RainIntensityNS

colnames(WeeklyDischargeNS)[1] <- "ID"

# Bind both data frames

WeeklyHydro = rbind(WeeklyDischarge, WeeklyDischargeNS)
WeeklyHydro$ID = as.character(WeeklyHydro$ID)
WeeklyHydro = with(WeeklyHydro, WeeklyHydro[order(ID),])

head(WeeklyHydro)
```

```
##       ID AveDischarge.m3.h Volume.m3 Sampled.Hrs     Sampled CumRain.mm
## 39 W0-0x          1.204775  14.41714    11.96667 Not Sampled        2.8
## 1   W0-1          1.213511 100.15508    82.53333     Sampled        7.6
## 40 W0-2x          1.284719  48.34827    37.63333 Not Sampled        7.6
## 2   W1-1         14.316647 390.36726    27.26667     Sampled       16.8
## 3   W1-2         15.529299 359.24445    23.13333     Sampled        6.0
## 41 W1-3x          9.107720 877.37700    96.33333 Not Sampled        9.4
##    RainInt.mmhr
## 39   0.23398329
## 1    0.09208401
## 40   0.20194863
## 2    0.61613692
## 3    0.25936599
## 41   0.09757785
```

Save files

```
VolNS = sum(WeeklyDischargeNS$Volume.m3)
VolS = sum(WeeklyDischarge$Volume.m3)
print("% Not sampled: ")
```

```
## [1] "% Not sampled: "
```

```
(VolNS/(VolNS+VolS))*100
```

```
## [1] 35.60444
```

```
write.csv2(WeeklyHydro,
           'Data/WeeklyHydro_R.csv', row.names = F)
```