Merging Discharge & Sampler Data

PAZ

26 octobre 2016

Purpose

A. This document merges corrected flowmeter data and automatic sampler data.

B. Converts 6 min rainfall to monthly statistics with visualizations comparing discharge.

Used files:

1. **hydroAlteck2016_smooth_R.csv**
2. **prelev_20160713.csv**
3. **sixMinutePluvioAlteck2016.csv**
4. **Data/2minRain.csv** (at the end)

Produced file:

1. **hydroAlteck2016_R.csv** (Used for plotting Sample and Discharge data together).
2. **RainAprJune2016_R.csv** (March and July delted)
3. **WaterDay_R.csv** (Daily and cumulative rainfall - mm) 4- **SamplerFormatted_R.csv** (Cleaned sampler)

Required R-packages:

```
# Plotting functions
library("ggplot2")
library("scales")
library("tidyr")
```

```
## Warning: package 'tidyr' was built under R version 3.3.3
```

```
library("dplyr")
library("reshape")
library("ggrepel")
```

Working directory

```
# setwd("D:/Documents/these_pablo/Alteckendorf2016/R")
# setwd("/Users/DayTightChunks/Documents/PhD/Routput/Alteck/R")
# setwd("D:/Documents/these_pablo/Alteckendorf2016/00_TransparencyFolder/Discharge")
getwd()
```

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

Import imputed discharge and rainfall data

```
# Discharge
dischargeAlteck = read.csv2("Data/hydroAlteck2016_smooth_R.csv")
head(dischargeAlteck)
```

```
##                  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
```

```
dischargeAlteck$Date = as.POSIXct(strptime(dischargeAlteck$DateCheck,
                                           "%d/%m/%Y %H:%M"
                                           , tz="EST")
                                  )

dischargeAlteck$DayMoYr = as.POSIXct(strptime(dischargeAlteck$DateCheck,
                                           "%d/%m/%Y"
                                           , tz="EST")
                                  )

sum(is.na(dischargeAlteck$Date))
```

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

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

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

```
## [1] Date      DateCheck Q.m3Hrs   Qna       Qapprox   Qinterp   Q.HW1
## [8] Q.HW2     DayMoYr
## <0 rows> (or 0-length row.names)
```

Convert discharge Q to volume V

Convert discharge at Δt of 2 min:

$$V(t) = \int_0^{\Delta t} Q(t)dt$$

```
## Convert m3.h -> m3
dischargeAlteck$Vol2min <- dischargeAlteck$Q.HW1*2/60
```

Compare discharge to rainfall

```
# 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")  )

sum(is.na(rain$Date))
```

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

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

#library(dplyr)
# detach("package:plyr")
dischDay <- dischargeAlteck %>%
  group_by(DayMoYr) %>%
  dplyr::summarize(Disch.mm = (sum(Vol2min)/(47*10000))*10^3 ) # Discharge in mm

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

rainDay$CumRain <- cumsum(rainDay$Rain.mm)

waterDay <- merge(rainDay, dischDay, by = "DayMoYr")

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

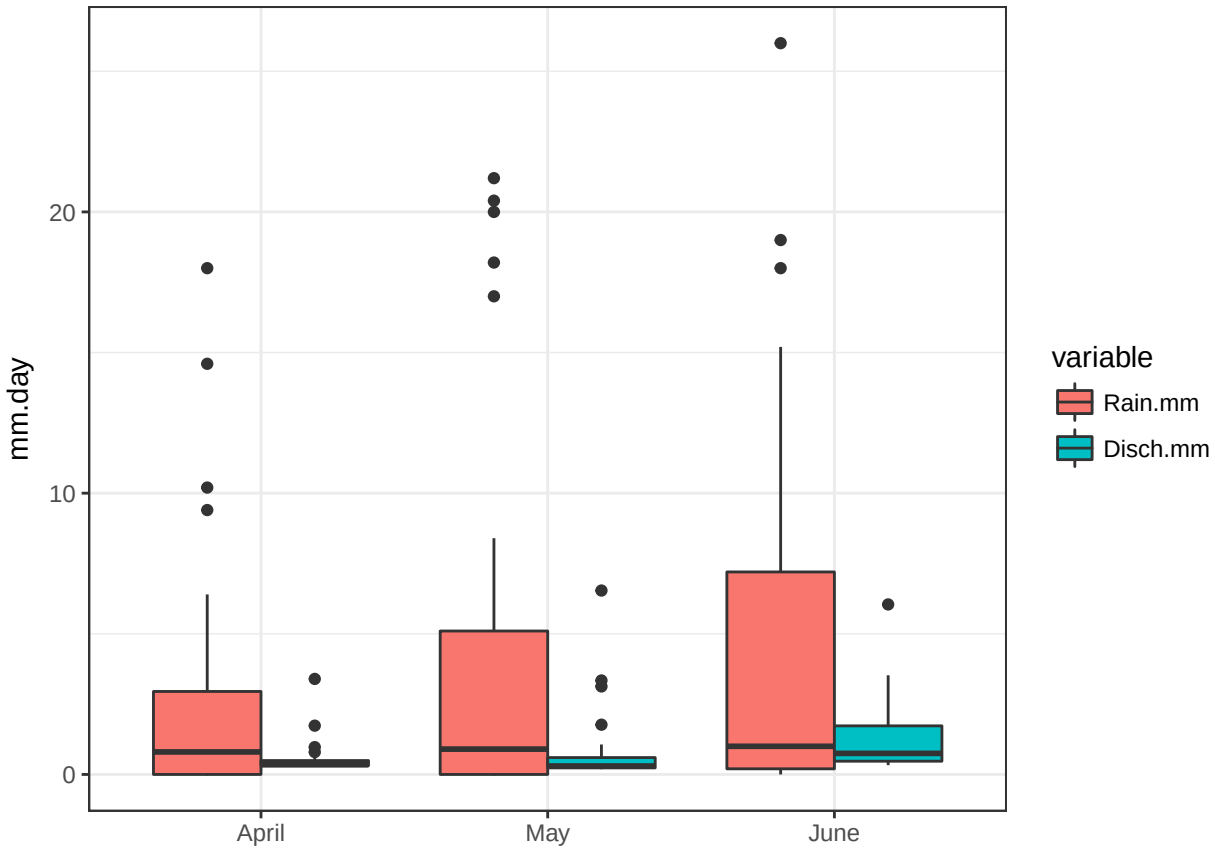
```
write.csv2(waterDay,
           'Data/WaterDay_R.csv', row.names = F)
```

Delete unused months

```
waterDay <- subset(waterDay, Month != "March" & Month != "July")
waterDay$Month <- factor(waterDay$Month, levels = c("April" , "May", "June" ))
levels(waterDay$Month)
```

```
## [1] "April" "May"   "June"
```

```
meltWaterDay <- melt(waterDay, id=c("DayMoYr", "Month"))

split <- strsplit(as.character(meltWaterDay$DayMoYr), "2016-", fixed = TRUE)
meltWaterDay$Day <- sapply(split, "[", 2)

meltWaterDay <- subset(meltWaterDay, variable != 'CumRain.mm')

# Test function
#g_param = 1.5
# g_param = 2.2  #  (Hoaglin et al.,1986; Hoaglin & Iglewicz, 1987)
g_param = 4  #  Just to capture XXX events
is_outlier <- function(x) {
  return(x < quantile(x, 0.25) - g_param * IQR(x) | x > quantile(x, 0.75) + g_param * IQR(x))
}

meltWaterDay2 <- meltWaterDay %>%
  group_by(Month) %>%
  #mutate(outlier = ifelse(is_outlier(value) & variable == "Disch.mm", as.character(Day), NA))
  mutate(outlier = ifelse(is_outlier(value), as.character(Day), NA))

hydroBox <- ggplot(meltWaterDay2) +
  geom_boxplot(data = subset(meltWaterDay2, variable != "CumRain"), aes(x = factor(Month), y = value, f
  theme_bw() +
  ylab("mm.day") +
  theme(axis.title.x = element_blank() ) +
  geom_text_repel(data = subset(meltWaterDay2, variable != "CumRain"), aes(x = factor(Month), y = value
                  size = 3,
                  arrow = arrow(length = unit(0.005, 'npc'), type = "closed"),
                  force = 0.5,
                  point.padding = unit(0.9, 'lines'),
                  max.iter = 2e3,
                  nudge_x = .05,
                  na.rm = TRUE)
hydroBox
```




```
#ggsave(hydroBox, filename = "RainDischBox.png", width = 8, height = 5, units = "in", scale = 1)
```

As Bar Plots

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

dischSumm <- dischDay %>%
  group_by(Month) %>%
  dplyr::summarize(Qtot.measure = sum(Disch.mm),
            Qmedian.measure = median(Disch.mm),
            Qmean.measure = mean(Disch.mm),
            Qmean.SD = sd(Disch.mm))
```

```
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"
                       )
                )
         )

rainSumm <- rainDay %>%
  group_by(Month) %>%
  dplyr:: summarize(Rtot.measure = sum(Rain.mm),
            Rmedian.measure = median(Rain.mm),
            Rmean.measure = mean(Rain.mm),
            Rmean.SD = sd(Rain.mm))

MonthSumm <- merge(rainSumm, dischSumm, by = "Month")
#MonthSumm$Rest.measure <- (MonthSumm$Qtot.measure/MonthSumm$Rtot.measure)*100

MonthSumm <- subset(MonthSumm, Month != "March" & Month != "July")


monthTidy <- MonthSumm %>%
  gather(measure, value, -Month) %>% # Melts data frame
  separate(measure, into = c("Source", "temporary_var")) %>% # parses the sep = "." into...
  spread(temporary_var, value) # Moves molten temporary variable to own column

monthTidy$Type <- ifelse(monthTidy$Source == "Rtot" | monthTidy$Source == "Qtot", "Total", "Summary")
monthTidy <- subset(monthTidy, Source != "Qmedian" & Source != "Rmedian")

monthTidy$Month <- as.factor(monthTidy$Month)
levels(monthTidy$Month)
```

```
## [1] "April" "June"  "May"
```

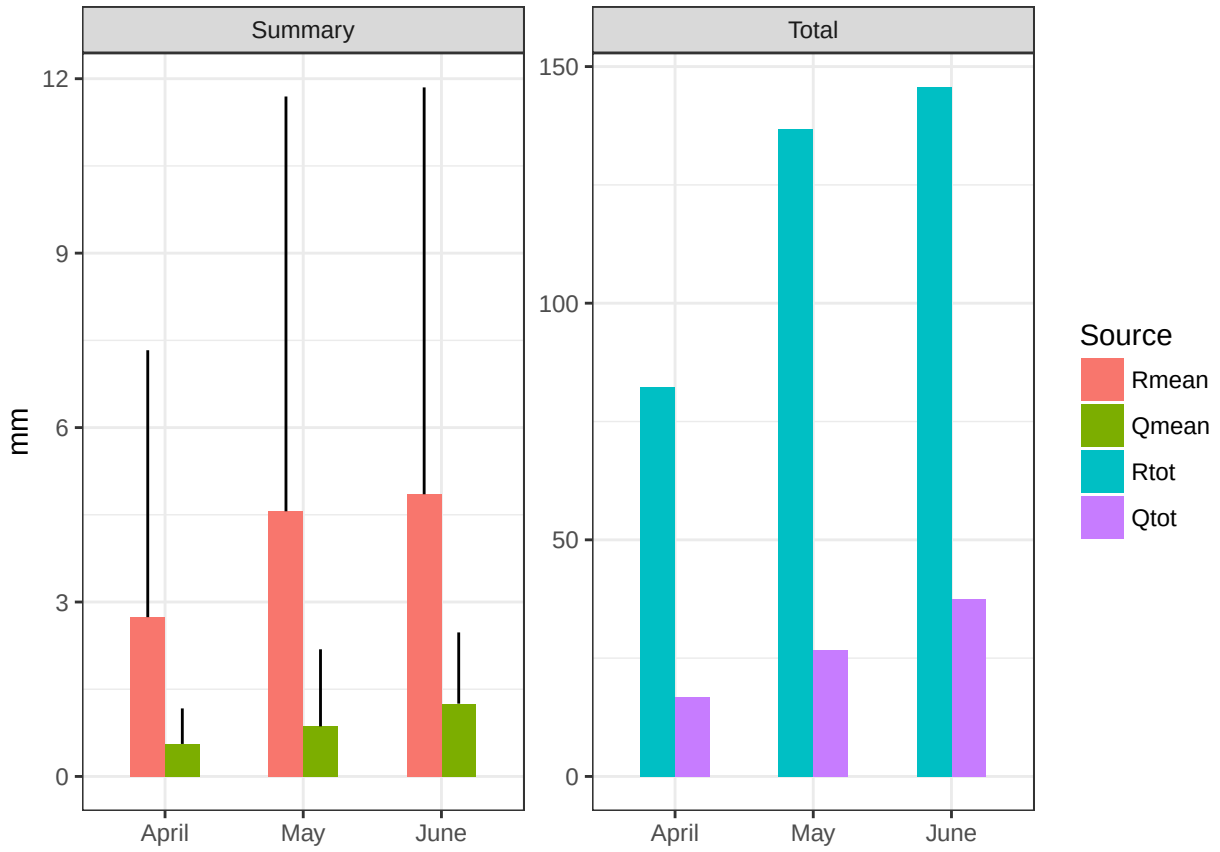
```
monthTidy$Month <- factor(monthTidy$Month, levels = c("April" , "May", "June" ))
monthTidy$Source <- factor(monthTidy$Source, levels = c("Rmean", "Qmean", "Rtot",  "Qtot"))
levels(monthTidy$Source)
```

```
## [1] "Rmean" "Qmean" "Rtot"  "Qtot"
```

```
ggplot(data = monthTidy,  aes(x=Month, y=measure, fill = Source )) +
  geom_bar(stat = "identity", position = "dodge", width = 0.5) + #, ymin=measure-SD, ymax=measure+SD) +
  facet_wrap(~Type, scale="free") +
  theme_bw() +
  geom_linerange(aes(ymin = measure, ymax = measure+SD),
                 width=.2 , # ) + #,                    # Width of the error bars
                  position=position_dodge(.5)) +
  ylab("mm") +
  theme(axis.title.x = element_blank() )
```

```
## Warning: Ignoring unknown parameters: width
```

```
## Warning: Removed 6 rows containing missing values (geom_linerange).
```




```
  # xlab("Month") #+
  #scale_fill_manual(#values = c("#6a51a3" , "#ec7014", "#807dba", "#fe9929"), # purple-orange
  #                   # values = c("#6a51a3" , "#ec7014", "#d9d9d9", "#fe9929"), # Unknown as grey
  #                   name= element_blank(), #"Mass Balance", # \n
  #                  breaks=c("Qmean", "Qtot" , "Rmean" , "Rtot"),
  #                   labels=c("Qmean", "Qtot" , "Rmean" , "Rtot")) +
  # guides(fill=guide_legend(ncol=2))
```

Import raw sampler data (March 25th to Jul 12th)

```
samplesAlteck = read.csv2("Data/prelev_20160713.csv", header = FALSE)

head(samplesAlteck)
```

```
##                 V1 V2
## 1 25/03/2016 12:04  1
## 2 26/03/2016 08:33  1
## 3 27/03/2016 06:04  1
## 4 28/03/2016 02:52  2
## 5 28/03/2016 22:37  2
## 6 30/03/2016 06:20  1
```

```
samplesAlteck = samplesAlteck[samplesAlteck$V2 != 0, ]
samplesAlteck$Date = as.POSIXct(strptime(samplesAlteck$V1,
                                         "%d/%m/%Y %H:%M",
                                         tz="EST"))
sum(is.na(samplesAlteck$V1))
```

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

```
samplesAlteck = samplesAlteck[,c(3,1:2)]
colnames(samplesAlteck) <- c("Date", "DateCheck", "sampleQ")

sum(is.na(samplesAlteck$Date))
```

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

```
samplesAlteck = samplesAlteck[order(samplesAlteck$Date),]

head(samplesAlteck)
```

```
##                  Date        DateCheck sampleQ
## 1 2016-03-25 12:04:00 25/03/2016 12:04       1
## 2 2016-03-26 08:33:00 26/03/2016 08:33       1
## 3 2016-03-27 06:04:00 27/03/2016 06:04       1
## 4 2016-03-28 02:52:00 28/03/2016 02:52       2
## 5 2016-03-28 22:37:00 28/03/2016 22:37       2
## 6 2016-03-30 06:20:00 30/03/2016 06:20       1
```

```
write.csv2(samplesAlteck,
           'Data/SamplerFormatted_R.csv', row.names = F)
```

Merge the Discharge and the Samples' dataframes

To merge the two data.frames, we need to correct minutes in the sample data, some of which took place during odd minutes.

1. Identify the odd minutes in a temporary data set to discard

```
discard = merge(dischargeAlteck, samplesAlteck, by = "Date", all = T)

# How many missing Discharge values resulting from the merge?
sum(is.na(discard$Date))
```

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

```
sum(is.na(discard$Q.m3Hrs))
```

```
## [1] 190
```

```
naQs = discard[is.na(discard$Q.m3Hrs == TRUE),]

naQs$Date = naQs$Date+60

naQs = naQs[,c("Date", "DateCheck.y")]

head(naQs)
```

```
##                  Date      DateCheck.y
```

```
## 976  2016-03-26 08:34:00 26/03/2016 08:33
## 2839 2016-03-28 22:38:00 28/03/2016 22:37
## 4359 2016-03-31 01:16:00 31/03/2016 01:15
## 4432 2016-03-31 03:40:00 31/03/2016 03:39
## 4451 2016-03-31 04:16:00 31/03/2016 04:15
## 4628 2016-03-31 10:08:00 31/03/2016 10:07
```

```
head(dischargeAlteck)
```

```
##                  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    DayMoYr    Vol2min
## 1 1.248600             1.182 2016-03-25 0.04162000
## 2 1.237280 1.15424267729696 2016-03-25 0.04124267
## 3 1.232224 1.17062590682503 2016-03-25 0.04107413
## 4 1.224779 1.15615409458726 2016-03-25 0.04082597
## 5 1.223623 1.17724053690379 2016-03-25 0.04078745
## 6 1.222299 1.17698892559366 2016-03-25 0.04074329
```

2. Add these odd-date markers to the flow-meter data (note that Date column remains as even minutes)

```
# Merge new dates to discharge data
hydroAlteck2016 = merge(dischargeAlteck, naQs, by = c("Date"), all = T)
head(hydroAlteck2016)
```

```
##                  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    DayMoYr    Vol2min DateCheck.y
## 1 1.248600             1.182 2016-03-25 0.04162000        <NA>
## 2 1.237280 1.15424267729696 2016-03-25 0.04124267        <NA>
## 3 1.232224 1.17062590682503 2016-03-25 0.04107413        <NA>
## 4 1.224779 1.15615409458726 2016-03-25 0.04082597        <NA>
## 5 1.223623 1.17724053690379 2016-03-25 0.04078745        <NA>
## 6 1.222299 1.17698892559366 2016-03-25 0.04074329        <NA>
```

```
# Check number of odd-minute dates, should be 0:
sum(is.na(hydroAlteck2016$Q.m3Hrs))
```

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

```
# Fill in the rest  of the Target dates (even)
hydroAlteck2016$DateCheck.S <- ifelse(is.na(hydroAlteck2016$DateCheck.y),
                                      as.character(hydroAlteck2016$DateCheck),
                                      as.character(hydroAlteck2016$DateCheck.y))
hydroAlteck2016$DateCheck.y <- NULL

# Create common column name in samples' target column (i.e. DateCheck.S)
```

```
samplesAlteck <- samplesAlteck[, c("DateCheck", "sampleQ")]
colnames(samplesAlteck) <- c("DateCheck.S", "sampleQ")
head(samplesAlteck)
```

```
##        DateCheck.S sampleQ
## 1 25/03/2016 12:04       1
## 2 26/03/2016 08:33       1
## 3 27/03/2016 06:04       1
## 4 28/03/2016 02:52       2
## 5 28/03/2016 22:37       2
## 6 30/03/2016 06:20       1
```

3. Merging the two tables

```
hydroAlteck2016 = merge(hydroAlteck2016, samplesAlteck, by = c("DateCheck.S"), all = T)

# Checks
sum(is.na(hydroAlteck2016$Date))
```

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

```
anyDuplicated(hydroAlteck2016$Date)
```

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

```
sum(is.na(hydroAlteck2016$Q.m3Hrs))
```

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

```
head(hydroAlteck2016)
```

```
##        DateCheck.S                Date        DateCheck Q.m3Hrs   Qna
## 1 01/04/2016 00:00 2016-04-01 00:00:00 01/04/2016 00:00   17.12 17.12
## 2 01/04/2016 00:02 2016-04-01 00:02:00 01/04/2016 00:02   14.71 14.71
## 3 01/04/2016 00:04 2016-04-01 00:04:00 01/04/2016 00:04   13.82 13.82
## 4 01/04/2016 00:06 2016-04-01 00:06:00 01/04/2016 00:06   14.58 14.58
## 5 01/04/2016 00:08 2016-04-01 00:08:00 01/04/2016 00:08   13.62 13.62
## 6 01/04/2016 00:10 2016-04-01 00:10:00 01/04/2016 00:10   14.48 14.48
##   Qapprox Qinterp    Q.HW1            Q.HW2    DayMoYr   Vol2min sampleQ
## 1   17.12   17.12 14.63129 16.0163558897583 2016-04-01 0.4877096      NA
## 2   14.71   14.71 15.12903 16.9994185259239 2016-04-01 0.5043010      NA
## 3   13.82   13.82 15.04522 14.8463828626439 2016-04-01 0.5015075      NA
## 4   14.58   14.58 14.80018  13.860730304389 2016-04-01 0.4933393      NA
## 5   13.62   13.62 14.75614 14.4885280729906 2016-04-01 0.4918714      NA
## 6   14.48   14.48 14.52891 13.6487756027902 2016-04-01 0.4842972      NA
```

```
class(hydroAlteck2016$Date)
```

```
## [1] "POSIXct" "POSIXt"
```

```
# Order by date
hydroAlteck2016 = hydroAlteck2016[order(hydroAlteck2016$Date),]
```

Create a "Type"" column to point to Sampling times during plotting

```
hydroAlteck2016$Type = ifelse(is.na(hydroAlteck2016$sampleQ), "Discharge", "Sample")
```

```
head(hydroAlteck2016)
```

```
##            DateCheck.S                Date        DateCheck Q.m3Hrs   Qna
## 60042 25/03/2016 00:04 2016-03-25 00:04:00 25/03/2016 00:04   1.192 1.192
## 60043 25/03/2016 00:06 2016-03-25 00:06:00 25/03/2016 00:06   1.212 1.212
## 60044 25/03/2016 00:08 2016-03-25 00:08:00 25/03/2016 00:08   1.195 1.195
## 60045 25/03/2016 00:10 2016-03-25 00:10:00 25/03/2016 00:10   1.219 1.219
## 60046 25/03/2016 00:12 2016-03-25 00:12:00 25/03/2016 00:12   1.217 1.217
## 60047 25/03/2016 00:14 2016-03-25 00:14:00 25/03/2016 00:14   1.230 1.230
##       Qapprox Qinterp    Q.HW1            Q.HW2    DayMoYr    Vol2min
## 60042   1.192   1.192 1.248600            1.182 2016-03-25 0.04162000
## 60043   1.212   1.212 1.237280 1.15424267729696 2016-03-25 0.04124267
## 60044   1.195   1.195 1.232224 1.17062590682503 2016-03-25 0.04107413
## 60045   1.219   1.219 1.224779 1.15615409458726 2016-03-25 0.04082597
## 60046   1.217   1.217 1.223623 1.17724053690379 2016-03-25 0.04078745
## 60047   1.230   1.230 1.222299 1.17698892559366 2016-03-25 0.04074329
##       sampleQ      Type
## 60042      NA Discharge
## 60043      NA Discharge
## 60044      NA Discharge
## 60045      NA Discharge
## 60046      NA Discharge
## 60047      NA Discharge
```

```
sum(is.na(hydroAlteck2016$Q.HW1))
```

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

Merge rainfall data

1. Merge 2 and 12 min minute pluvio data, handling NA's as 0

```
rain2min =  read.csv2("Data/2minRain.csv", header = T, dec = ".")

rain2min$Rain.mm <- as.numeric(rain2min$Rain.mm)
rain2min$Date <- as.character(rain2min$Date)
rain2min$Date <- as.POSIXct(strptime(rain2min$Date,
                                     "%d/%m/%Y %H:%M",
                                     tz="EST"))

hydroAlteck2016 = merge(hydroAlteck2016, rain2min, by = "Date", all = T)
hydroAlteck2016$Rain.mm <- ifelse(is.na(hydroAlteck2016$Rain.mm), 0, hydroAlteck2016$Rain.mm)

rain12m = read.csv2("Data/12minRain.csv", dec = ".")
rain12m$Date = as.POSIXct(strptime(rain12m$Date,
                                   "%d/%m/%Y %H:%M", tz="EST") )

colnames(rain12m)[2]  <- "Rain12min.mm"
hydroAlteck2016 <-  merge(hydroAlteck2016, rain12m, by = "Date", all = T)
hydroAlteck2016$Rain12min.mm <-
  ifelse(is.na(hydroAlteck2016$Rain12min.mm), 0, hydroAlteck2016$Rain12min.mm)

# Should be zero
sum(is.na(hydroAlteck2016$Rain.mm))
```

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

```
# Check which rainfall minutes do not have Discharge, and if Rain = 0, delete obs.
# naValues <- subset(hydroAlteck2016, is.na(hydroAlteck2016$Q.HW1)) # Only NA's where Rain == 0
hydroAlteck2016 <- subset(hydroAlteck2016, !is.na(hydroAlteck2016$Q.HW1))
```

Saving

```
write.csv2(hydroAlteck2016, "Data/hydroAlteck2016_R.csv", row.names = F)
```