Report data-processing sondes-project 2022

Emanuel Mauch

2023-07-07

Introduction

This is a report of the data-processing steps of the Greenland sonde data of 2022. I adapted Moritz' code from his paper (https://doi.org/10.1002/ecy.3371). I have cleaned all the raw datasets, stored them as .txt files (folder "Raw_sonde_2022_3"), merged them with RStudio, processed the merged dataset and plotted some figures of the absolute values. The code was adapted from Moritz' file "make_data.R", using his "method_packages.R". All the files I've used to make this report are in the "project_22" folder that I have uploaded.

Steps done so far

Step 0: Clean raw datafiles in Excel, store them as .txt files

I cleaned all the raw .csv files with Excel (e.g. reassembling split files, fixing date bugs which lead to wrong .txt file conversions etc.). They can be found in the folder "Raw_sonde_2022_3".

Step 1: set up R-script

```
rm(list= ls())

setwd("~/ZIVI_EAWAG/project_22")

source("~/ZIVI_EAWAG/project_22/Moritz_Luehrig_paper_stuff/methods_packages.R")
```

```
## Lade nötiges Paket: pacman
```

```
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
sonde_key = fread( "~/ZIVI_EAWAG/project_22/ponds_sonde_key.txt", header=T)
print(head(sonde_key))
```

```
##    Pond   Sonde Treatment
## 1: B2P3 Sonde06         D
## 2: B2P2 Sonde09         S
## 3: B1P1 Sonde16         S
## 4: B3P3 Sonde10         S
## 5: B3P1 Sonde14         D
## 6: B3P2 Sonde13         D
```

Step 2: merge all raw data files into a single big one

```
path = "~/ZIVI_EAWAG/project_22/Raw_sonde_2022_3"
filenames <- list.files(path=path, pattern=".txt")
files<-data.table(NULL)

system.time(for(i in filenames){
  filepath <- file.path(path,paste(i,sep=""))
  dummy1<-fread(filepath,
                colClasses = "character" ,
                sep = "\t",
                header = FALSE,
                fill=TRUE,
                encoding ="UTF-8")
  rownumb<-dummy1[V1 %like% "(MM/DD/YYYY)", which=TRUE]+1
  colname<-dummy1[V1 %like% "(MM/DD/YYYY)"]
  colname<-gsub(" ", "",(gsub("[[:punct:]]", "",c(lapply(colname,as.character)))))
  colname<-unlist(strsplit(colname,"\t"))
  dummy2<-dummy1[rownumb:.N]
  setnames(dummy2,colname)
  dummy2[,(colname[c(3,5:length(dummy2))]):=lapply(.SD, as.numeric),.SDcols=colname[c(3,5:length(dummy2)
  dummy2$Sonde<-substr(i, 1,7)
  dummy2$Source_file<-i
  files<-rbindlist(list(files, dummy2), fill=TRUE)
})
```

```
##        User      System verstrichen
##        0.92        0.28        2.27
```

```
all <- files

# save intermediate stage 1 (FULL RAW DATA, NO REMOVAL)
fwrite(all, "~/ZIVI_EAWAG/project_22/intermediate_processing_steps/ponds_sonde_data_intermediate_1.txt"
print(head(all,3))
```

```
##    DateMMDDYYYY TimeHHMMSS TimeFractSec   SiteName ChlorophyllRFU
## 1:    6/22/2022   14:02:33            0 greeenland          0.36
## 2:    6/22/2022   14:04:33            0 greeenland         -0.21
## 3:    6/22/2022   14:06:33            0 greeenland         -0.09
##    ChlorophyllugL ConduScm fDOMQSU fDOMRFU nLFConduScm ODOsat ODOlocal ODOmgL
## 1:           6.17      0.0    3.28    0.89         0.0   93.2     94.4   8.49
## 2:           4.35     38.0   -0.65   -0.41        49.6   97.2     98.5  10.16
## 3:           4.75     38.8   -0.21   -0.27        50.7   96.9     98.2  10.13
```

```
##    Salpsu SpConduScm BGAPCRFU BGAPCugL TDSmgL WiperPositionvolt   pH pHmV
## 1:   0.00        0.0    -0.26    -0.26      0             1.196 6.32 36.9
## 2:   0.02       48.8    -1.49    -1.49     32             1.201 6.75 13.6
## 3:   0.02       49.9    -1.41    -1.41     32             1.192 7.07 -3.0
##     TempC BatteryV CablePwrV   Sonde           Source_file
## 1: 19.898     5.08         0 Sonde06 Sonde06_B2P3_2022.txt
## 2: 13.367     5.02         0 Sonde06 Sonde06_B2P3_2022.txt
## 3: 13.380     5.05         0 Sonde06 Sonde06_B2P3_2022.txt
```

Step 3: add exp. design, phases and format time

```
all <- fread("~/ZIVI_EAWAG/project_22/intermediate_processing_steps/ponds_sonde_data_intermediate_1.txt

# format date & time
all$TimeHHMMSS<-substr(all$TimeHHMMSS,1,5) #gsub("\\D+", "", cleaning$TimeHHMMSS)
all$DateMMDDYYYY<-mdy(all$DateMMDDYYYY)
all$Date_time<-ymd_hm(paste(all$DateMMDDYYYY, all$TimeHHMMSS))
```

```
## Warning: 7 failed to parse.
```

```
setnames(all,1:2,c("Date","Time"))

# add fractions of day since start
# all = all[order(Date_time, Sonde)]
all[,Time_seq:=round(((as.numeric(Date_time))-
                        (as.numeric(Date_time)[1]))/86400,7)]

# merge with key
all<-merge(all,sonde_key, by = "Sonde")

# reorder and revalue
setorderv(all, c("Pond", "Sonde", "Date_time"), c(1, 1, 1))

# select columns to keep
all<-all[,c("Pond","Sonde", "Treatment", "Time_seq", "Date_time",
            "ChlorophyllRFU", "ConduScm" , "BGAPCRFU", "ODOmgL", "pH", "fDOMRFU", "SpConduScm", "TempC"

# rename
setnames(all, c("Pond", "Sonde", "Treatment", "Time_seq", "Date_time",
                "Chlorophyll_RFU", "Cond_uScm", "BGAPC_RFU", "ODO_mgL", "pH", "fDOM_RFU", "SpCond_uScm"

# debug: rm duplicate value
all = unique(all, by=c("Pond", "Time_seq"))

fwrite(all, "~/ZIVI_EAWAG/project_22/intermediate_processing_steps/ponds_sonde_data_intermediate_2.txt"
print(head(all,3))
```

```
##    Pond   Sonde Treatment   Time_seq           Date_time Chlorophyll_RFU
## 1: B1P1 Sonde16         S -0.1798611 2022-06-22 09:43:00            0.20
## 2: B1P1 Sonde16         S -0.1784722 2022-06-22 09:45:00            0.15
## 3: B1P1 Sonde16         S -0.1770833 2022-06-22 09:47:00            0.22
```

```
##    Cond_uScm BGAPC_RFU ODO_mgL    pH fDOM_RFU SpCond_uScm Temp_C
## 1:       0.0      1.59   11.15 10.63     0.72         0.1  9.344
## 2:      36.0     -1.28   10.33  6.53     3.52        46.7 13.055
## 3:      36.3     -1.42   10.28  6.79     3.50        47.0 13.073
```

Step 4.1: remove outliers

The outlier plots are in the folder "outliers".

```
all <- fread("~/ZIVI_EAWAG/project_22/intermediate_processing_steps/ponds_sonde_data_intermediate_2.txt

save_path = "outliers/"

for(i in  c("Chlorophyll_RFU", "BGAPC_RFU", "ODO_mgL")){
  all[,  (i) := lapply(.SD[,..i],
                       outlier_removal,
                       window=48,
                       threshold=10,
                       plot=T,
                       return="removed",
                       save=T,
                       path=save_path,
                       name=paste0(i,"_",Pond)),
      by = c("Pond")]
}

for(i in c("fDOM_RFU")){
  all[,  (i) := lapply(.SD[,..i],
                       outlier_removal,
                       window=96,
                       threshold=10,
                       plot=F,
                       return="removed",
                       save=T,
                       path=save_path,
                       name=paste0(i,"_",Pond)),
      by = c("Pond")]
}

for(i in c("Temp_C", "SpCond_uScm", "Cond_uScm")){
  all[,  (i) := lapply(.SD[,..i],
                       outlier_removal,
                       window=96,
                       threshold=20,
                       plot=F,
                       return="removed",
                       save=T,
                       path=save_path,
                       name=paste0(i,"_",Pond)),
      by = c("Pond")]
}
```

```
# cut-off "loose ends" on both sides -> all same start and same end of measurements
all <- subset(all, Date_time>"2022-06-23 22:00:00" & Date_time<"2022-06-29 05:00:00")

# add fractions of day since start
# all = all[order(Date_time, Sonde)]
all[,Time_seq:=round(((as.numeric(Date_time))-
                        (as.numeric(Date_time)[1]))/86400,7)]

# ommit date-time (just needed for cutting the dates)
all<-all[,!c("Date_time")]

# save
fwrite(all, "~/ZIVI_EAWAG/project_22/intermediate_processing_steps/ponds_sonde_data_intermediate_3.1.tx
```

Step 4.2: remove anomalies

We inspected the data for anomalies and remove them here - for details see Russo, S., M. Lührig, W. Hao, B. Matthews, and K. Villez. 2020. Active learning for anomaly detection in environmental data. Environmental Modelling & Software 134:104869.

I didn't do this step, since I would need some time to refresh my python skills, but if you think it is worth, I will manage.

Step 5: add light and precipitation data

I do not have this data available.

Step 6: intercept correction

This uses the results form a cross correlation survey to adjust the sondes for "off factory differences, i.e. remove any intercept variation among the data.

I do not have the sonds correlation factors.

Step 7: save

```
## order and remove rownames
all = all[order(Pond, Time_seq), ]

rownames(all)<-NULL

fwrite(all, "~/ZIVI_EAWAG/project_22/data/ponds_sonde_data_all.txt")
```

```
all<-fread("~/ZIVI_EAWAG/project_22/data/ponds_sonde_data_all.txt")
print(head(all,3))
```

```
##    Pond   Sonde Treatment  Time_seq           Date_time Chlorophyll_RFU
## 1: B1P1 Sonde16         S 0.0000000 2022-06-23 20:01:00            0.61
## 2: B1P1 Sonde16         S 0.0013889 2022-06-23 20:03:00            0.04
```

```
## 3: B1P1 Sonde16         S 0.0027778 2022-06-23 20:05:00           0.13
##    Cond_uScm BGAPC_RFU ODO_mgL  pH fDOM_RFU SpCond_uScm Temp_C
## 1:      39.7     -1.45   10.49 7.2     3.25        50.3 14.015
## 2:      39.7     -1.56   10.51 7.2     3.23        50.3 13.996
## 3:      39.7     -1.49   10.50 7.2     3.21        50.3 13.982
```

Step 8: check completeness

```
all <- fread("~/ZIVI_EAWAG/project_22/intermediate_processing_steps/ponds_sonde_data_intermediate_3.1.t
all$Date_time <- as.POSIXct((all$Time_seq*86400), origin="2022-06-23 20:00:00", tz="UTC")

all$Date <-date(all$Date_time)

# N per date
all[, N := uniqueN(Date), by = Sonde]
complete <- setDT(data.frame(unclass(table(all$Date, all$Sonde))), keep.rownames=T)
colnames(complete)[1]<-"Date"
fwrite(complete, "~/ZIVI_EAWAG/project_22/data/sonde_data_completeness.txt", sep="\t")
print(complete)
```

```
##          Date Sonde06 Sonde07 Sonde08 Sonde09 Sonde10 Sonde11 Sonde12 Sonde13
## 1: 2022-06-23     120     120     120     120     120     120     120     120
## 2: 2022-06-24     720     720     720     720     720     720     720     720
## 3: 2022-06-25     720     720     720     720     720     720     720     720
## 4: 2022-06-26     720     720     720     720     720     720     720     720
## 5: 2022-06-27     720     720     720     720     720     720     720     720
## 6: 2022-06-28     720     720     720     720     720     720     720     720
## 7: 2022-06-29      89      89      90      89      90      89      89      90
##    Sonde14 Sonde15 Sonde16
## 1:     120      75     120
## 2:     720     662     720
## 3:     720     656     720
## 4:     720     323     720
## 5:     720     560     720
## 6:     720     491     720
## 7:      90      48      90
```

Step 9: calculate daily averages

```
all<-fread("~/ZIVI_EAWAG/project_22/data/ponds_sonde_data_all.txt")
summary(all)
```

```
##      Pond               Sonde            Treatment            Time_seq
##  Length:40910       Length:40910       Length:40910       Min.   :0.000
##  Class :character   Class :character   Class :character   1st Qu.:1.304
##  Mode  :character   Mode  :character   Mode  :character   Median :2.605
##                                                           Mean   :2.632
##                                                           3rd Qu.:3.960
##                                                           Max.   :5.290
```

```
##
##    Date_time                     Chlorophyll_RFU     Cond_uScm
##  Min.   :2022-06-23 20:01:00.00   Min.   :-0.6300   Min.   : 30.60
##  1st Qu.:2022-06-25 03:19:00.00   1st Qu.:-0.1900   1st Qu.: 37.70
##  Median :2022-06-26 10:32:00.00   Median : 0.2300   Median : 43.30
##  Mean   :2022-06-26 11:11:03.67   Mean   : 0.4643   Mean   : 58.05
##  3rd Qu.:2022-06-27 19:03:00.00   3rd Qu.: 0.8800   3rd Qu.: 71.60
##  Max.   :2022-06-29 02:59:00.00   Max.   : 4.7100   Max.   :134.90
##                                   NA's   :327       NA's   :2224
##    BGAPC_RFU          ODO_mgL            pH           fDOM_RFU
##  Min.   :-2.5200   Min.   : 8.82   Min.   :6.300   Min.   :-0.270
##  1st Qu.:-1.2700   1st Qu.:10.25   1st Qu.:6.660   1st Qu.: 0.340
##  Median :-0.6300   Median :10.59   Median :7.090   Median : 1.230
##  Mean   :-0.8058   Mean   :10.63   Mean   :7.336   Mean   : 2.013
##  3rd Qu.:-0.3000   3rd Qu.:11.01   3rd Qu.:7.360   3rd Qu.: 2.690
##  Max.   : 0.2100   Max.   :12.79   Max.   :9.990   Max.   : 9.790
##  NA's   :29        NA's   :4584                    NA's   :483
##   SpCond_uScm         Temp_C
##  Min.   : 40.50   Min.   :10.28
##  1st Qu.: 50.00   1st Qu.:12.03
##  Median : 57.20   Median :12.86
##  Mean   : 74.67   Mean   :12.95
##  3rd Qu.: 95.80   3rd Qu.:14.06
##  Max.   :165.30   Max.   :15.45
##  NA's   :4298
```

```
all[,Time_seq:=as.integer(Time_seq)]

parameters = c("Chlorophyll_RFU", "Cond_uScm", "BGAPC_RFU", "ODO_mgL", "pH",
               "fDOM_RFU", "SpCond_uScm", "Temp_C")
groups = c("Time_seq", "Pond", "Sonde", "Treatment")

all.summ = all[,lapply(.SD, m_r, 2), by=groups, .SDcols = parameters]

fwrite(all.summ, "~/ZIVI_EAWAG/project_22/data/ponds_sonde_data_daily_avg.txt", sep=",")

table(all.summ$Pond, all.summ$Treatment)
```

```
##
##          D NF S
##   B1P1   0  0 6
##   B2P2   0  0 6
##   B2P3   6  0 0
##   B2P4   0  6 0
##   B3P0   0  6 0
##   B3P1   6  0 0
##   B3P2   6  0 0
##   B3P3   0  0 6
##   B3P4   0  6 0
##   ERL152 0  6 0
##   ERL85  0  6 0
```

```
print(head(all.summ,3))
```

```
##    Time_seq Pond   Sonde Treatment Chlorophyll_RFU Cond_uScm BGAPC_RFU ODO_mgL
## 1:        0 B1P1 Sonde16         S            0.15     39.98     -1.46   10.37
## 2:        1 B1P1 Sonde16         S            0.27     39.72     -1.43   10.38
## 3:        2 B1P1 Sonde16         S            0.36     37.92     -1.37   10.26
##      pH fDOM_RFU SpCond_uScm Temp_C
## 1: 7.16     3.13       50.37  14.20
## 2: 7.20     3.10       50.35  13.94
## 3: 6.99     3.35       49.75  12.49
```

Step 10: hourly averages

```
all<-fread("~/ZIVI_EAWAG/project_22/data/ponds_sonde_data_all.txt")

parameters = c("Chlorophyll_RFU", "Cond_uScm", "BGAPC_RFU", "ODO_mgL", "pH",
               "fDOM_RFU", "SpCond_uScm", "Temp_C")

all$Date_time <- as.POSIXct((all$Time_seq*86400), origin="2022-06-23 20:00:00", tz="UTC")

all$Time_seq = as.integer(all$Time_seq)
all$Hour <- as.numeric(substr(as.character(all$Date_time),12,13))

groups = c("Time_seq", "Hour","Pond", "Treatment")

all.summ = all[,lapply(.SD, m_r, 2), by=groups, .SDcols = parameters]

fwrite(all.summ, "~/ZIVI_EAWAG/project_22/data/ponds_sonde_data_hourly_avg.txt", sep=",")
print(head(all.summ,3))
```

```
##    Time_seq Hour Pond Treatment Chlorophyll_RFU Cond_uScm BGAPC_RFU ODO_mgL
## 1:        0   20 B1P1         S            0.03     39.71     -1.47   10.44
## 2:        0   21 B1P1         S            0.02     39.70     -1.48   10.45
## 3:        0   22 B1P1         S            0.11     39.70     -1.46   10.43
##      pH fDOM_RFU SpCond_uScm Temp_C
## 1: 7.18     3.20        50.3  14.00
## 2: 7.18     3.20        50.3  14.05
## 3: 7.17     3.22        50.3  14.00
```

Step 11: 10-minute averages

I thought that maybe the hour interval is too large for our time series, so maybe a 10-minute interval could be more appropriate (or just plot¨ the absolute values).

```
all<-fread("~/ZIVI_EAWAG/project_22/data/ponds_sonde_data_all.txt")

parameters = c("Chlorophyll_RFU", "Cond_uScm", "BGAPC_RFU", "ODO_mgL", "pH",
               "fDOM_RFU", "SpCond_uScm", "Temp_C")
```

```
all$Date_time <- as.POSIXct((all$Time_seq*86400), origin="2022-06-23 20:00:00", tz="UTC")

all$Time_seq = as.integer(all$Time_seq)
all$Hour <- as.numeric(substr(as.character(all$Date_time),12,13))
all$Ten_min <- as.numeric(substr(as.character(all$Date_time),15,15))

groups = c("Time_seq", "Hour", "Ten_min", "Pond", "Treatment")

all.summ = all[,lapply(.SD, m_r, 2), by=groups, .SDcols = parameters]

fwrite(all.summ, "~/ZIVI_EAWAG/project_22/data/ponds_sonde_data_ten_min_avg.txt", sep=",")
print(head(all.summ,3))
```

```
##    Time_seq Hour Ten_min Pond Treatment Chlorophyll_RFU Cond_uScm BGAPC_RFU
## 1:        0   20       0 B1P1         S            0.17     39.73     -1.46
## 2:        0   20       1 B1P1         S            0.01     39.73     -1.45
## 3:        0   20       2 B1P1         S            0.00     39.72     -1.50
##    ODO_mgL   pH fDOM_RFU SpCond_uScm Temp_C
## 1:   10.48 7.20     3.22        50.3  13.99
## 2:   10.43 7.18     3.20        50.3  13.99
## 3:   10.42 7.17     3.19        50.3  14.00
```

Step 12: calculate overall averages of individual ponds

```
all<-fread("~/ZIVI_EAWAG/project_22/data/ponds_sonde_data_all.txt")

parameters = c("Chlorophyll_RFU", "Cond_uScm", "BGAPC_RFU", "ODO_mgL", "pH",
               "fDOM_RFU", "SpCond_uScm", "Temp_C")
groups = c("Pond","Treatment", "Sonde")

all.summ = all[,lapply(.SD, m_r, 2), by=groups, .SDcols = parameters]

fwrite(all.summ, "~/ZIVI_EAWAG/project_22/data/ponds_sonde_data_overall_avg.txt", sep=",")

table(all.summ$Pond, all.summ$Treatment)
```

```
##
##          D NF S
##   B1P1   0  0 1
##   B2P2   0  0 1
##   B2P3   1  0 0
##   B2P4   0  1 0
##   B3P0   0  1 0
##   B3P1   1  0 0
##   B3P2   1  0 0
##   B3P3   0  0 1
##   B3P4   0  1 0
##   ERL152 0  1 0
##   ERL85  0  1 0
```

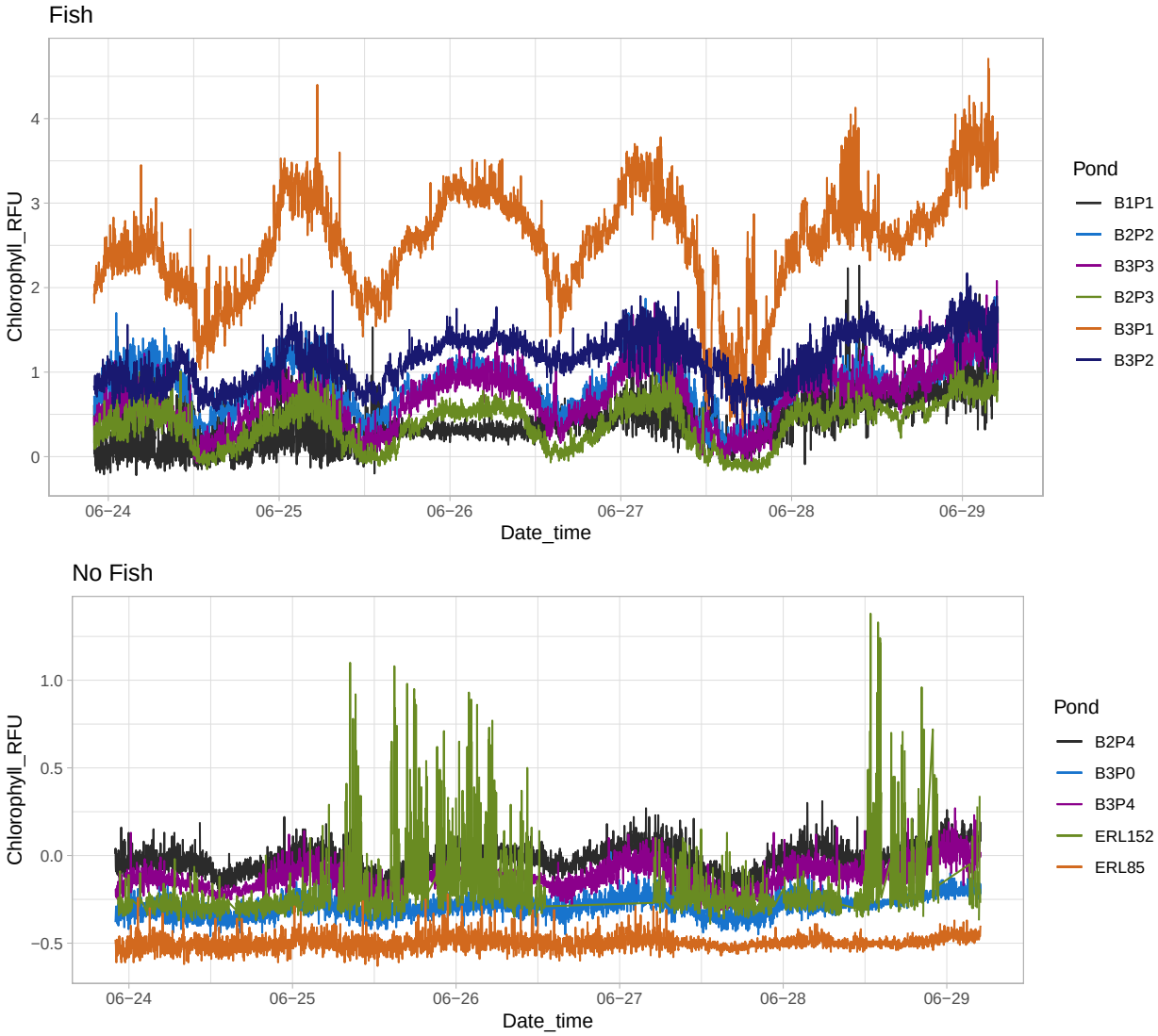
Step 13: control plots

Comment: No dates were cutoff in this plots, the origin of time is the first measurement of the sondes (sonde 16, pond B1P1). However, the data was processed with outlier-removal functions (see step 4.1).

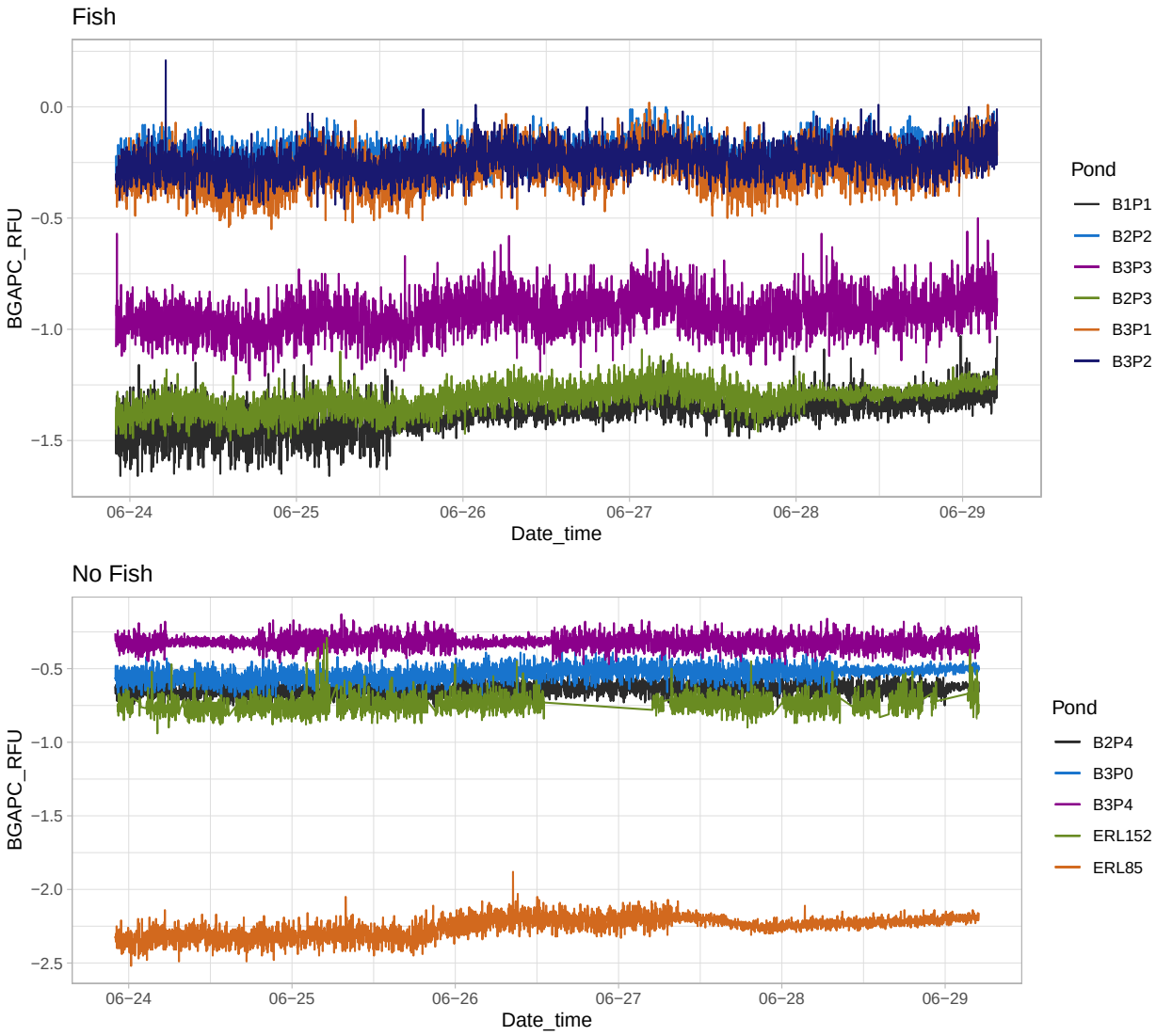
```
all<-fread("~/ZIVI_EAWAG/project_22/data/ponds_sonde_data_all.txt")

all$Date_time <- as.POSIXct((all$Time_seq*86400), origin="2022-06-23 22:00:00", tz="UTC")

table(all$Pond, all$Treatment)
```

```
##
##             D   NF    S
##   B1P1      0    0 3810
##   B2P2      0    0 3809
##   B2P3   3809    0    0
##   B2P4      0 3810    0
##   B3P0      0 3809    0
##   B3P1   3810    0    0
##   B3P2   3810    0    0
##   B3P3      0    0 3810
##   B3P4      0 3809    0
##   ERL152    0 2815    0
##   ERL85     0 3809    0
```

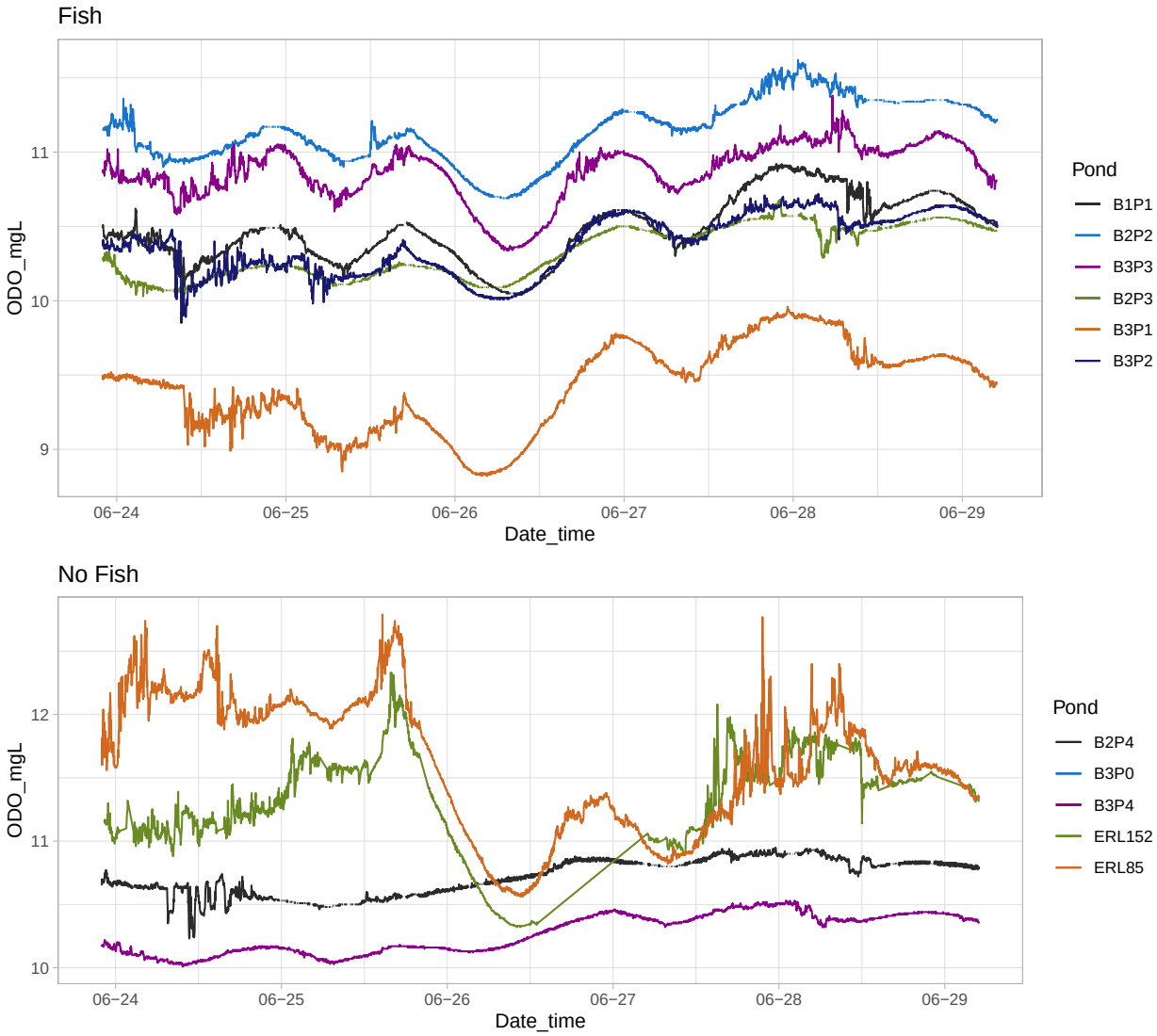
Chlorophyll RFU



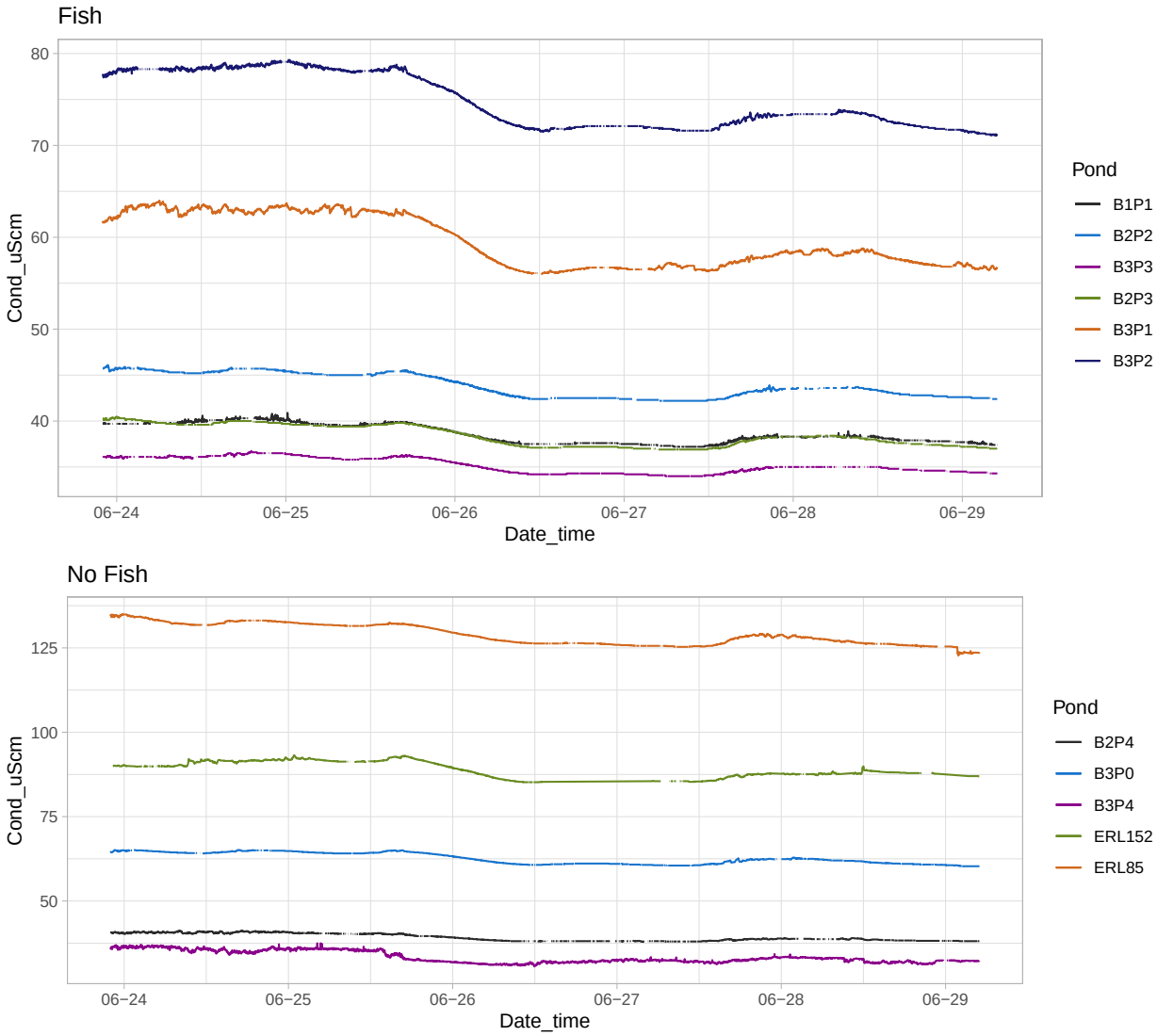
BGAPC RFU


Fish
BGAPC_RFU
0.0
−0.5
−1.0
−1.5
06−24
06−25
06−26
06−27
06−28
06−29
Date_time
Pond
B1P1
B2P2
B3P3
B2P3
B3P1
B3P2
No Fish
BGAPC_RFU
−0.5
−1.0
−1.5
−2.0
−2.5
06−24
06−25
06−26
06−27
06−28
06−29
Date_time
Pond
B2P4
B3P0
B3P4
ERL152
ERL85

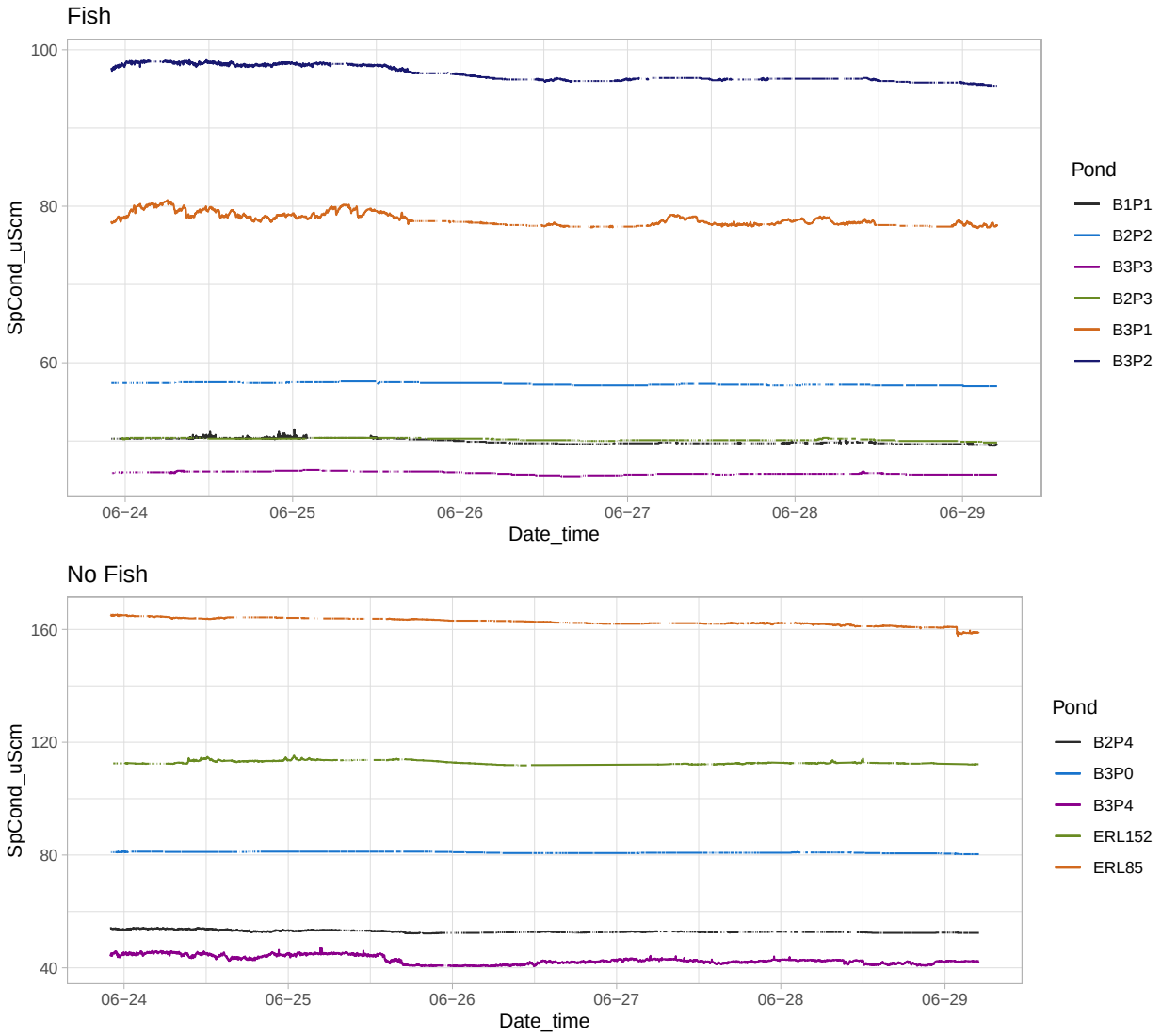
ODO mg/L


Fish
ODO_mgL
11
10
9
06−24
06−25
06−26
06−27
06−28
06−29
Date_time
Pond
B1P1
B2P2
B3P3
B2P3
B3P1
B3P2
No Fish
ODO_mgL
12
11
10
06−24
06−25
06−26
06−27
06−28
06−29
Date_time
Pond
B2P4
B3P0
B3P4
ERL152
ERL85

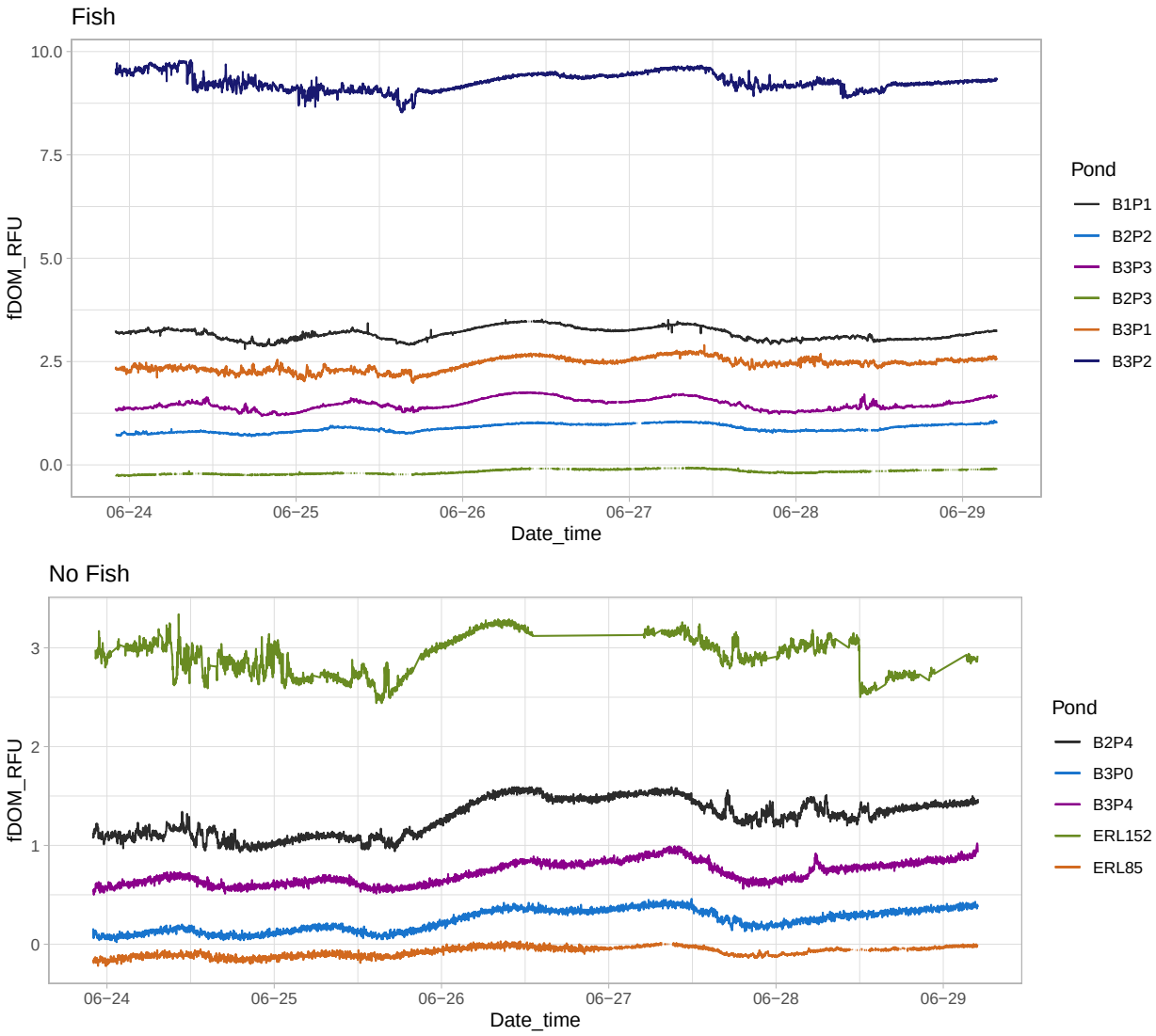
Cond uS/cm


Fish
Cond_uScm
80
70
60
50
40
06−24
06−25
06−26
06−27
06−28
06−29
Date_time
Pond
B1P1
B2P2
B3P3
B2P3
B3P1
B3P2
No Fish
Cond_uScm
125
100
75
50
06−24
06−25
06−26
06−27
06−28
06−29
Date_time
Pond
B2P4
B3P0
B3P4
ERL152
ERL85

Speficic Cond uS/cm



fDOM RFU



14: Appendix

Version and packages used to generate this report:

```
## 2023-07-21 14:08:48.772279 Europe/Zurich

## R version 4.3.1 (2023-06-16 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19045)
##
## Matrix products: default
##
##
## locale:
```

```
## [1] LC_COLLATE=German_Switzerland.utf8  LC_CTYPE=German_Switzerland.utf8
## [3] LC_MONETARY=German_Switzerland.utf8 LC_NUMERIC=C
## [5] LC_TIME=German_Switzerland.utf8
##
## time zone: Europe/Zurich
## tzcode source: internal
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] GGally_2.1.2      zoo_1.8-12        viridis_0.6.3     viridisLite_0.4.2
##  [5] forcats_1.0.0     stringr_1.5.0     dplyr_1.1.2       purrr_1.0.1
##  [9] readr_2.1.4       tidyr_1.3.0       tibble_3.2.1      ggplot2_3.4.2
## [13] tidyverse_2.0.0   lubridate_1.9.2   data.table_1.14.8 cowplot_1.1.1
## [17] bit64_4.0.5       bit_4.0.5         pacman_0.5.1
##
## loaded via a namespace (and not attached):
##  [1] utf8_1.2.3         generics_0.1.3     stringi_1.7.12     lattice_0.21-8
##  [5] hms_1.1.3          digest_0.6.32      magrittr_2.0.3     RColorBrewer_1.1-3
##  [9] evaluate_0.21      grid_4.3.1         timechange_0.2.0   fastmap_1.1.1
## [13] plyr_1.8.8         reshape_0.8.9      gridExtra_2.3      fansi_1.0.4
## [17] scales_1.2.1       cli_3.6.1          rlang_1.1.1        munsell_0.5.0
## [21] withr_2.5.0        yaml_2.3.7         tools_4.3.1        tzdb_0.4.0
## [25] colorspace_2.1-0   vctrs_0.6.3        R6_2.5.1           lifecycle_1.0.3
## [29] pkgconfig_2.0.3    pillar_1.9.0       gtable_0.3.3       Rcpp_1.0.10
## [33] glue_1.6.2         highr_0.10         xfun_0.39          tidyselect_1.2.0
## [37] rstudioapi_0.14    knitr_1.43         farver_2.1.1       htmltools_0.5.5
## [41] labeling_0.4.2     rmarkdown_2.23     compiler_4.3.1
```

Code used to generate control plots:

```
# Chloropyll_RFU
p1 = ggplot(subset(all, Pond=="B1P1")) + theme_light() +
  geom_line(aes(y=Chlorophyll_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B2P2"), aes(y=Chlorophyll_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P3"), aes(y=Chlorophyll_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B2P3"), aes(y=Chlorophyll_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P1"), aes(y=Chlorophyll_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P2"), aes(y=Chlorophyll_RFU, x=Date_time, color = Pond)) +
  scale_x_datetime(date_breaks = "1 day", date_labels = "%m-%d") +
  scale_color_manual(breaks=c('B1P1', 'B2P2', 'B3P3', "B2P3", "B3P1", "B3P2"),
                     values=c('B1P1'="gray17", 'B2P2'='dodgerblue3', 'B3P3'='magenta4',
                              "B2P3"="olivedrab4", "B3P1"="chocolate", "B3P2"="midnightblue")) +
  ggtitle("Fish")

p2 = ggplot(subset(all, Pond=="B2P4")) + theme_light() +
  geom_line(aes(y=Chlorophyll_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P0"), aes(y=Chlorophyll_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P4"), aes(y=Chlorophyll_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="ERL152"), aes(y=Chlorophyll_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="ERL85"), aes(y=Chlorophyll_RFU, x=Date_time, color = Pond)) +
```

```
  scale_x_datetime(date_breaks = "1 day", date_labels = "%m-%d") +
  scale_color_manual(breaks=c('B2P4', 'B3P0', 'B3P4', "ERL152", "ERL85"),
                     values=c('B2P4'="gray17", 'B3P0'='dodgerblue3', 'B3P4'='magenta4',
                              "ERL152"="olivedrab4", "ERL85"="chocolate")) +
  ggtitle("No Fish")

plot_grid(p1, p2, nrow = 2, rel_heights =  c(0.229,0.2,0.2,0.32))

# ODO mg/L
p3 = ggplot(subset(all, Pond=="B1P1")) + theme_light() +
  geom_line(aes(y=ODO_mgL, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B2P2"), aes(y=ODO_mgL, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P3"), aes(y=ODO_mgL, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B2P3"), aes(y=ODO_mgL, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P1"), aes(y=ODO_mgL, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P2"), aes(y=ODO_mgL, x=Date_time, color = Pond)) +
  scale_x_datetime(date_breaks = "1 day", date_labels = "%m-%d") +
  scale_color_manual(breaks=c('B1P1', 'B2P2', 'B3P3', "B2P3", "B3P1", "B3P2"),
                     values=c('B1P1'="gray17", 'B2P2'='dodgerblue3', 'B3P3'='magenta4',
                              "B2P3"="olivedrab4", "B3P1"="chocolate", "B3P2"="midnightblue")) +
  ggtitle("Fish")

p4 = ggplot(subset(all, Pond=="B2P4")) + theme_light() +
  geom_line(aes(y=ODO_mgL, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P0"), aes(y=ODO_mgL, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P4"), aes(y=ODO_mgL, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="ERL152"), aes(y=ODO_mgL, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="ERL85"), aes(y=ODO_mgL, x=Date_time, color = Pond)) +
  scale_x_datetime(date_breaks = "1 day", date_labels = "%m-%d") +
  scale_color_manual(breaks=c('B2P4', 'B3P0', 'B3P4', "ERL152", "ERL85"),
                     values=c('B2P4'="gray17", 'B3P0'='dodgerblue3', 'B3P4'='magenta4',
                              "ERL152"="olivedrab4", "ERL85"="chocolate")) +
  ggtitle("No Fish")

plot_grid(p3, p4, nrow = 2, rel_heights =  c(0.229,0.2,0.2,0.32))

# BGAPC RFU
p5 = ggplot(subset(all, Pond=="B1P1")) + theme_light() +
  geom_line(aes(y=BGAPC_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B2P2"), aes(y=BGAPC_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P3"), aes(y=BGAPC_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B2P3"), aes(y=BGAPC_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P1"), aes(y=BGAPC_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P2"), aes(y=BGAPC_RFU, x=Date_time, color = Pond)) +
  scale_x_datetime(date_breaks = "1 day", date_labels = "%m-%d") +
  scale_color_manual(breaks=c('B1P1', 'B2P2', 'B3P3', "B2P3", "B3P1", "B3P2"),
                     values=c('B1P1'="gray17", 'B2P2'='dodgerblue3', 'B3P3'='magenta4',
                              "B2P3"="olivedrab4", "B3P1"="chocolate", "B3P2"="midnightblue")) +
  ggtitle("Fish")

p6 = ggplot(subset(all, Pond=="B2P4")) + theme_light() +
  geom_line(aes(y=BGAPC_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P0"), aes(y=BGAPC_RFU, x=Date_time, color = Pond)) +
```

```
  geom_line(data = subset(all, Pond=="B3P4"), aes(y=BGAPC_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="ERL152"), aes(y=BGAPC_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="ERL85"), aes(y=BGAPC_RFU, x=Date_time, color = Pond)) +
  scale_x_datetime(date_breaks = "1 day", date_labels = "%m-%d") +
  scale_color_manual(breaks=c('B2P4', 'B3P0', 'B3P4', "ERL152", "ERL85"),
                     values=c('B2P4'="gray17", 'B3P0'='dodgerblue3', 'B3P4'='magenta4',
                              "ERL152"="olivedrab4", "ERL85"="chocolate")) +
  ggtitle("No Fish")

plot_grid(p5, p6, nrow = 2, rel_heights =  c(0.229,0.2,0.2,0.32))



# Cond uS/cm
p7 = ggplot(subset(all, Pond=="B1P1")) + theme_light() +
  geom_line(aes(y=Cond_uScm, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B2P2"), aes(y=Cond_uScm, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P3"), aes(y=Cond_uScm, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B2P3"), aes(y=Cond_uScm, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P1"), aes(y=Cond_uScm, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P2"), aes(y=Cond_uScm, x=Date_time, color = Pond)) +
  scale_x_datetime(date_breaks = "1 day", date_labels = "%m-%d") +
  scale_color_manual(breaks=c('B1P1', 'B2P2', 'B3P3', "B2P3", "B3P1", "B3P2"),
                     values=c('B1P1'="gray17", 'B2P2'='dodgerblue3', 'B3P3'='magenta4',
                              "B2P3"="olivedrab4", "B3P1"="chocolate", "B3P2"="midnightblue")) +
  ggtitle("Fish")

p8 = ggplot(subset(all, Pond=="B2P4")) + theme_light() +
  geom_line(aes(y=Cond_uScm, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P0"), aes(y=Cond_uScm, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P4"), aes(y=Cond_uScm, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="ERL152"), aes(y=Cond_uScm, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="ERL85"), aes(y=Cond_uScm, x=Date_time, color = Pond)) +
  scale_x_datetime(date_breaks = "1 day", date_labels = "%m-%d") +
  scale_color_manual(breaks=c('B2P4', 'B3P0', 'B3P4', "ERL152", "ERL85"),
                     values=c('B2P4'="gray17", 'B3P0'='dodgerblue3', 'B3P4'='magenta4',
                              "ERL152"="olivedrab4", "ERL85"="chocolate")) +
  ggtitle("No Fish")

plot_grid(p7, p8, nrow = 2, rel_heights =  c(0.229,0.2,0.2,0.32))

# Speficic Cond uS/cm
p9 = ggplot(subset(all, Pond=="B1P1")) + theme_light() +
  geom_line(aes(y=SpCond_uScm, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B2P2"), aes(y=SpCond_uScm, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P3"), aes(y=SpCond_uScm, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B2P3"), aes(y=SpCond_uScm, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P1"), aes(y=SpCond_uScm, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P2"), aes(y=SpCond_uScm, x=Date_time, color = Pond)) +
  scale_x_datetime(date_breaks = "1 day", date_labels = "%m-%d") +
  scale_color_manual(breaks=c('B1P1', 'B2P2', 'B3P3', "B2P3", "B3P1", "B3P2"),
                     values=c('B1P1'="gray17", 'B2P2'='dodgerblue3', 'B3P3'='magenta4',
                              "B2P3"="olivedrab4", "B3P1"="chocolate", "B3P2"="midnightblue")) +
```

```
  ggtitle("Fish")

p10 = ggplot(subset(all, Pond=="B2P4")) + theme_light() +
  geom_line(aes(y=SpCond_uScm, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P0"), aes(y=SpCond_uScm, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P4"), aes(y=SpCond_uScm, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="ERL152"), aes(y=SpCond_uScm, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="ERL85"), aes(y=SpCond_uScm, x=Date_time, color = Pond)) +
  scale_x_datetime(date_breaks = "1 day", date_labels = "%m-%d") +
  scale_color_manual(breaks=c('B2P4', 'B3P0', 'B3P4', "ERL152", "ERL85"),
                     values=c('B2P4'="gray17", 'B3P0'='dodgerblue3', 'B3P4'='magenta4',
                              "ERL152"="olivedrab4", "ERL85"="chocolate")) +
  ggtitle("No Fish")

plot_grid(p9, p10, nrow = 2, rel_heights =  c(0.229,0.2,0.2,0.32))

# fDOM RFU
p11 = ggplot(subset(all, Pond=="B1P1")) + theme_light() +
  geom_line(aes(y=fDOM_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B2P2"), aes(y=fDOM_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P3"), aes(y=fDOM_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B2P3"), aes(y=fDOM_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P1"), aes(y=fDOM_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P2"), aes(y=fDOM_RFU, x=Date_time, color = Pond)) +
  scale_x_datetime(date_breaks = "1 day", date_labels = "%m-%d") +
  scale_color_manual(breaks=c('B1P1', 'B2P2', 'B3P3', "B2P3", "B3P1", "B3P2"),
                     values=c('B1P1'="gray17", 'B2P2'='dodgerblue3', 'B3P3'='magenta4',
                              "B2P3"="olivedrab4", "B3P1"="chocolate", "B3P2"="midnightblue")) +
  ggtitle("Fish")

p12 = ggplot(subset(all, Pond=="B2P4")) + theme_light() +
  geom_line(aes(y=fDOM_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P0"), aes(y=fDOM_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="B3P4"), aes(y=fDOM_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="ERL152"), aes(y=fDOM_RFU, x=Date_time, color = Pond)) +
  geom_line(data = subset(all, Pond=="ERL85"), aes(y=fDOM_RFU, x=Date_time, color = Pond)) +
  scale_x_datetime(date_breaks = "1 day", date_labels = "%m-%d") +
  scale_color_manual(breaks=c('B2P4', 'B3P0', 'B3P4', "ERL152", "ERL85"),
                     values=c('B2P4'="gray17", 'B3P0'='dodgerblue3', 'B3P4'='magenta4',
                              "ERL152"="olivedrab4", "ERL85"="chocolate")) +
  ggtitle("No Fish")

plot_grid(p11, p12, nrow = 2, rel_heights =  c(0.229,0.2,0.2,0.32))
```