

pH_HOBO_plots 1.1

Emily Roberts

12/17/23

Trim individual files to time deployed and retrieved, then compile and visualize the data. Also compare HOBO pH data to sediment temp data, YSI data, and HOBO conductivity data

Table of contents

0.1	Set up workspace	2
0.1.1	Dennis	2
0.1.2	Ptown	10
0.1.3	Eel Pond	53
0.2	Combine	87
0.3	Calculate tris buffer offset and SW pH from NBS scale	89
0.3.1	Plot tides	90
0.3.2	Final plots	101
0.3.3	Compare with surfclam carbonate chemistry data	103
0.4	Read YSI data	106
0.5	Sediment temp loggers	120
0.6	Conductivity	123
0.7	Flood and ebb tide analysis	151

0.1 Set up workspace

0.1.1 Dennis

```
setwd(wdDen)
filenames <- dir()
filenames
```

```
[1] "1230_East_Dennis 2022-08-05 16_12_44 EDT (Data EDT).csv"
[2] "1230_East_Dennis 2022-08-05 16_12_44 EDT (Data EDT).csv.orig"
[3] "1230_East_Dennis 2022-09-21 09_29_48 EDT (Data EDT).csv"
[4] "1230_East_Dennis 2022-09-21 09_29_48 EDT (Data EDT).csv.orig"
[5] "Den_077 2022-09-14 14_08_26 EDT (Data EDT).csv"
[6] "Den_077 2022-09-14 14_08_26 EDT (Data EDT).csv.orig"
[7] "Den_077 2022-09-21 09_28_37 EDT (Data EDT)(1).csv"
[8] "Den_077 2022-09-21 09_28_37 EDT (Data EDT)(1).csv.orig"
[9] "E Dennis 2022-06-13 08_04_00 EDT (Data EDT).csv"
[10] "E Dennis 2022-06-13 08_04_00 EDT (Data EDT).csv.orig"
[11] "Untitled.ipynb"
[12] "Untitled.ipynb.orig"
```

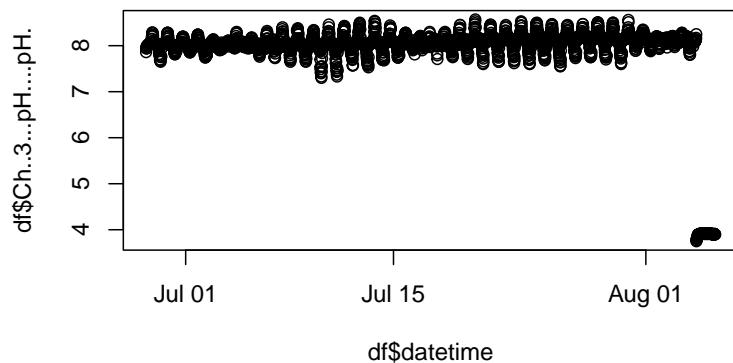
```
setwd(wdDen)
# Late June to early Aug
#(open_file <- filenames[1]) # "1230_East_Dennis 2022-08-05 16_12_44 EDT (Data EDT).csv"
open_file <- "1230_East_Dennis 2022-08-05 16_12_44 EDT (Data EDT).csv"
#Also works: (open_file <- filenames[2]) # "1230_East_Dennis 2022-09-21 09_29_48 EDT (Data EDT).csv"
df <- read.csv(open_file, stringsAsFactors = F)

df <- df[,c(1:5)]
head(df)
```

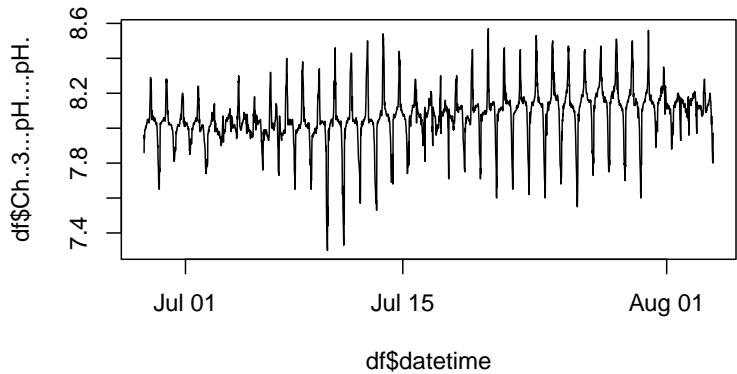
X.	Date.Time..EDT.	Ch..1...Temperature.....C..	Ch..2...Millivolts....mv.
1	06/28/2022 08:00:00	18.79	-76.61
2	06/28/2022 08:15:00	18.87	-82.05
3	06/28/2022 08:30:00	18.82	-81.16
4	06/28/2022 08:45:00	18.87	-80.99
5	06/28/2022 09:00:00	19.08	-83.07
6	06/28/2022 09:15:00	19.25	-83.75

```
Ch..3...pH....pH.  
1          7.86  
2          7.96  
3          7.94  
4          7.94  
5          7.98  
6          7.99
```

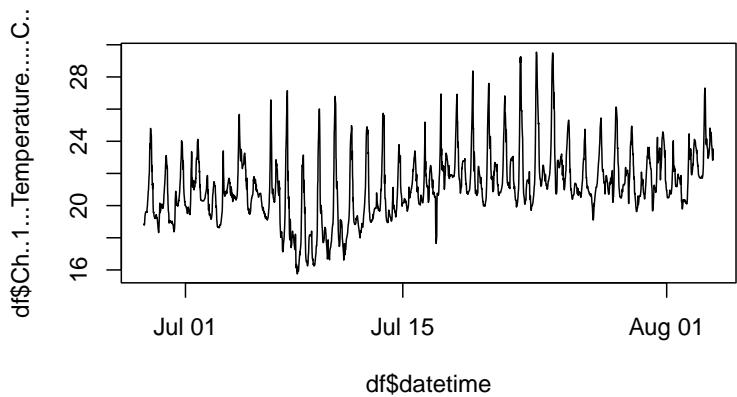
```
names(df) <- c("n", "Date.Time..EDT.", "Ch..1...Temperature.....C..", "mV", "Ch..3...pH....pH." )  
  
df$datetime <- as.POSIXct(strptime(df$Date.Time..EDT., format = "%m/%d/%Y %H:%M:%S"))  
plot(df$datetime, df$Ch..3...pH....pH.)
```



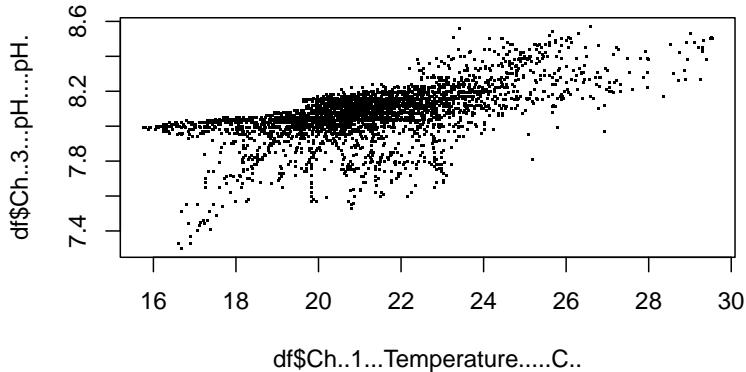
```
df <- df[df$datetime < "2022-08-04 00:08:00 EDT",]  
plot(df$datetime, df$Ch..3...pH....pH., type = "l")
```



```
plot(df$datetime, df$Ch..1....Temperature.....C.., type = "l")
```



```
plot(df$Ch..1....Temperature.....C.., df$Ch..3...pH....pH., pch = '.')
```



```

#Metadata
SN <- 230
tris <- 8.34
tris_temp <- 24.05

data_combined_new <- 0
data_combined_new <- data.frame(
  Site = rep("Den",times = nrow(df)),
  Date = df$datetime,
  Temp = df$Ch..1...Temperature.....C.,
  pH = df$Ch..3...pH....pH.,
  Tris = rep(tris,length(df$datetime)),
  Tris_temp = rep(tris_temp,length(df$datetime)),
  Serial_number = rep(SN,length(df$datetime))
)

data_combined <- rbind(data_combined_new)

# setwd(wdDen)
# # Late June to early Aug
# #(open_file <- filenames[1]) # "1230_East_Dennis 2022-08-05 16_12_44 EDT (Data EDT).csv"
# open_file <- "1230_East_Dennis 2022-08-05 16_12_44 EDT (Data EDT).csv"
# #Also works: (open_file <- filenames[2]) # "1230_East_Dennis 2022-09-21 09_29_48 EDT (Data ED
# df <- read.csv(open_file, stringsAsFactors = F)
# df$datetime <- as.POSIXct(strptime(df$Date.Time..EDT., format = "%m/%d/%Y %H:%M:%S"))
# plot(df$datetime, df$Ch..3...pH....pH.)

```

```

# df <- df[df$datetime<"2022-08-04 00:00:00 EDT",]
# plot(df$datetime, df$Ch..3...pH....pH., type = "l")
# plot(df$datetime, df$Ch..1...Temperature.....C.., type = "l")
# plot(df$Ch..1...Temperature.....C.., df$Ch..3...pH....pH., pch = '.')
#
# #Metadata
# SN <- 230
# tris <- 8.34
# tris_temp <- 24.05
#
# data_combined_new <- data.frame(
#   Site = rep("Den",times = nrow(df)),
#   Date = df$datetime,
#   Temp = df$Ch..1...Temperature.....C..,
#   pH = df$Ch..3...pH....pH.,
#   Tris = rep(tris,length(df$datetime)),
#   Tris_temp = rep(tris_temp,length(df$datetime)),
#   Serial_number = rep(SN,length(df$datetime))
# )
#
#
# data_combined <- rbind(data_combined_new)

```

```

setwd(wdDen)
# Aug to mid September
#"Den_077 2022-09-14 14_08_26 EDT (Data EDT).csv"
#"Den_077 2022-09-21 09_28_37 EDT (Data EDT)(1).csv" is the same file read out later on
#(open_file <- filenames[3])
open_file <- "Den_077 2022-09-14 14_08_26 EDT (Data EDT).csv"
df <- read.csv(open_file, stringsAsFactors = F)

df <- df[,c(1:5)]
head(df)

```

	X.	Date.Time..EDT.	Ch..1...Temperature.....C..	Ch..2...Millivolts....mv.
1	1	08/04/2022 09:00:00	23.91	180.19
2	2	08/04/2022 09:15:00	24.85	180.54
3	3	08/04/2022 09:30:00	26.18	181.01
4	4	08/04/2022 09:45:00	25.21	-77.55
5	5	08/04/2022 10:00:00	25.16	-78.10

```

6 6 08/04/2022 10:15:00           25.48          -79.74
Ch..3...pH....pH.
1      3.60
2      3.60
3      3.61
4      8.18
5      8.19
6      8.22

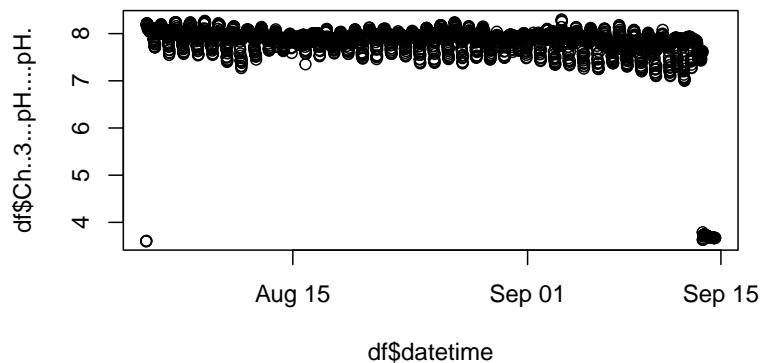
```

```

names(df) <- c("n", "Date.Time..EDT.", "Ch..1...Temperature.....C..", "mV", "Ch..3...pH....pH." )

df$datetime <- as.POSIXct(strptime(df>Date.Time..EDT., format = "%m/%d/%Y %H:%M:%S"))
plot(df$datetime, df$Ch..3...pH....pH.)

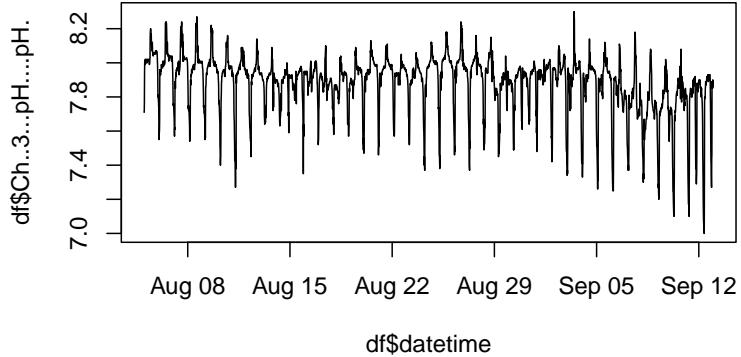
```



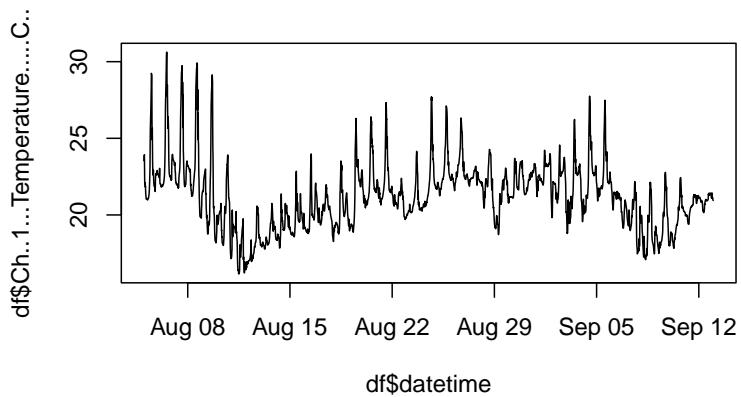
```

df <- df[df$datetime>"2022-08-05 00:00:00 EDT",]
df <- df[df$datetime<"2022-09-13 00:00:00 EDT",]
plot(df$datetime, df$Ch..3...pH....pH., type = "l")

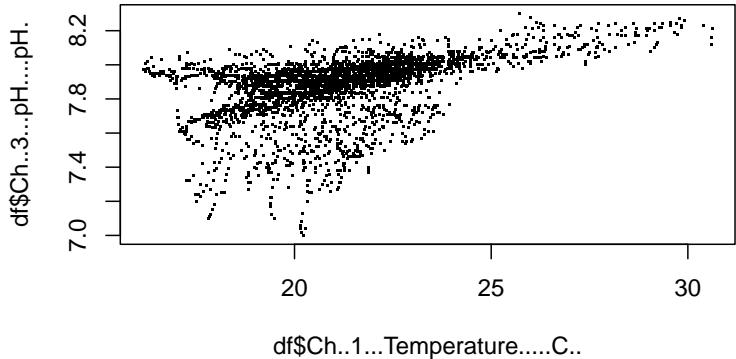
```



```
plot(df$datetime, df$Ch..1....Temperature.....C.., type = "l")
```



```
plot(df$Ch..1....Temperature.....C.., df$Ch..3...pH....pH., pch = '.')
```



```
#Metadata
SN <- 077
tris <- 8.16
tris_temp <- 23.98

data_combined_new <- data.frame(
  Site = rep("Den",times = nrow(df)),
  Date = df$datetime,
  Temp = df$Ch..1...Temperature.....C.,
  pH = df$Ch..3...pH....pH.,
  Tris = rep(tris,length(df$datetime)),
  Tris_temp = rep(tris_temp,length(df$datetime)),
  Serial_number = rep(SN,length(df$datetime))
)

data_combined <- rbind(data_combined,data_combined_new)

head(data_combined_new)
```

	Site	Date	Temp	pH	Tris	Tris_temp	Serial_number
1	Den	2022-08-05 00:15:00	23.54	7.71	8.16	23.98	77
2	Den	2022-08-05 00:30:00	23.79	7.71	8.16	23.98	77
3	Den	2022-08-05 00:45:00	23.93	7.77	8.16	23.98	77
4	Den	2022-08-05 01:00:00	23.04	7.94	8.16	23.98	77
5	Den	2022-08-05 01:15:00	22.15	8.00	8.16	23.98	77

```
6 Den 2022-08-05 01:30:00 22.06 7.99 8.16      23.98      77
```

```
tail(data_combined_new)
```

	Site	Date	Temp	pH	Tris	Tris_temp	Serial_number
3738	Den	2022-09-12 22:30:00	21.10	7.85	8.16	23.98	77
3739	Den	2022-09-12 22:45:00	21.13	7.87	8.16	23.98	77
3740	Den	2022-09-12 23:00:00	21.01	7.90	8.16	23.98	77
3741	Den	2022-09-12 23:15:00	20.98	7.89	8.16	23.98	77
3742	Den	2022-09-12 23:30:00	20.96	7.87	8.16	23.98	77
3743	Den	2022-09-12 23:45:00	20.94	7.86	8.16	23.98	77

0.1.2 Ptown

```
setwd(wdPtown)
filenames <- dir()
filenames
```

```
[1] "20573560_Ptown 2022-06-15 18_08_38 EDT (Data EDT).csv"
[2] "20573560_Ptown 2022-06-15 18_08_38 EDT (Data EDT).csv.orig"
[3] "20573560_Ptown 2022-08-05 17_08_58 EDT (Data EDT).csv"
[4] "20573560_Ptown 2022-08-05 17_08_58 EDT (Data EDT).csv.orig"
[5] "20573560_Ptown 2022-09-29 17_27_52 EDT (Data EDT).csv"
[6] "20573560_Ptown 2022-09-29 17_27_52 EDT (Data EDT).csv.orig"
[7] "20573560_Ptown 2022-11-02 15_02_10 EDT (Data EDT).csv"
[8] "20573560_Ptown 2023-05-10 14_13_07 EDT (Data EDT).csv"
[9] "20573560_Ptown 2023-05-10 14_13_07 EDT (Data EDT).csv.orig"
[10] "21333340_Eel_Pond 2022-11-07 15_11_16 EST (Data EST)_actuallyPtown.csv"
[11] "21333340_Eel_Pond 2022-11-07 15_11_16 EST (Data EST)_actuallyPtown.csv.orig"
[12] "340_Ptown 2023-06-11 14_35_41 EDT (Data EDT)(1).csv"
[13] "Ptown 2022-05-17 14_31_47 EDT (Data EDT) 2.csv"
[14] "Ptown 2022-05-17 14_31_47 EDT (Data EDT) 2.csv.orig"
[15] "Ptown 2022-05-17 14_31_47 EDT (Data EDT).csv"
[16] "Ptown 2022-05-17 14_31_47 EDT (Data EDT).csv.orig"
[17] "Ptown 2022-06-27 15_06_47 EDT (Data EDT).csv"
[18] "Ptown 2022-06-27 15_06_47 EDT (Data EDT).csv.orig"
[19] "Ptown_075 2022-08-29 21_52_54 EDT (Data EDT).csv"
[20] "Ptown_075 2022-08-29 21_52_54 EDT (Data EDT).csv.orig"
```

```
[21] "Ptown_076 2022-12-05 21_20_00 EST (Data EST).csv"
[22] "Ptown_076 2022-12-05 21_20_00 EST (Data EST).csv.orig"
[23] "Ptown_20573560_Ptown 2022-09-29 17_27_52 EDT (Data EDT).csv"
[24] "Ptown 20573560 Ptown 2022-09-29 17_27_52 EDT (Data EDT).csv.orig"
```

```
setwd(wdPtown)
# mid June to August
open_file <- "20573560_Ptown_2022-08-05_17_08_58_EDT_(Data_EDT).csv"
df <- read.csv(open_file, stringsAsFactors = F)
head(df)
```

X.	Date	Time..EDT.	Ch..1...Temperature.....C..	Ch..2...Millivolts....mv		
1	1	06/15/2022 18:09:46	20.91	-84.85		
2	2	06/15/2022 18:24:46	20.25	-89.80		
3	3	06/15/2022 18:39:46	19.46	-70.27		
4	4	06/15/2022 18:54:46	19.22	-72.31		
5	5	06/15/2022 19:09:46	19.25	-71.35		
6	6	06/15/2022 19:24:46	19.48	-85.28		
		Ch..3...pH....pH.	Button.Down	Button.Up	Host.Connected	End.of.File
1		8.36				
2		8.45				
3		8.10				
4		8.14				
5		8.12				
6		8.37				

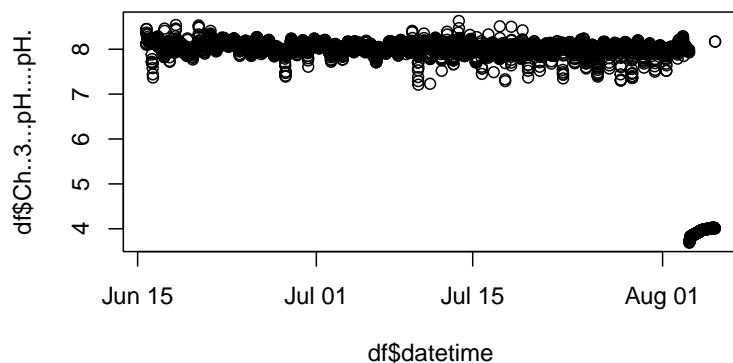
```
df <- df[,c(1:5)]  
head(df)
```

X.	Date	Time..EDT.	Ch..1...Temperature.....C..	Ch..2...Millivolts....mv.
1	1	06/15/2022 18:09:46	20.91	-84.85
2	2	06/15/2022 18:24:46	20.25	-89.80
3	3	06/15/2022 18:39:46	19.46	-70.27
4	4	06/15/2022 18:54:46	19.22	-72.31
5	5	06/15/2022 19:09:46	19.25	-71.35
6	6	06/15/2022 19:24:46	19.48	-85.28
		Ch..3...pH....pH.		
1			8.36	
2			8.45	

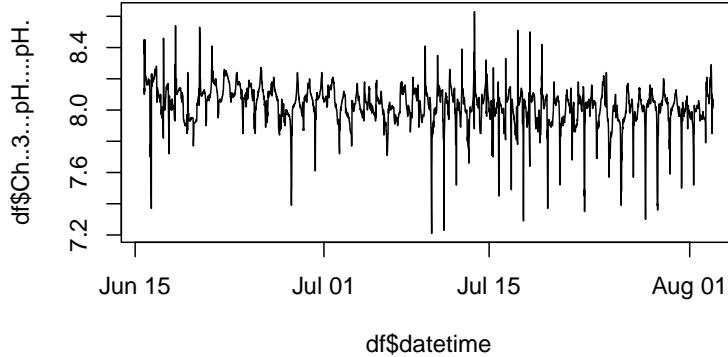
```
3          8.10
4          8.14
5          8.12
6          8.37
```

```
names(df) <- c("n", "Date.Time..EDT.", "Ch..1...Temperature.....C..", "mV", "Ch..3...pH....pH." )

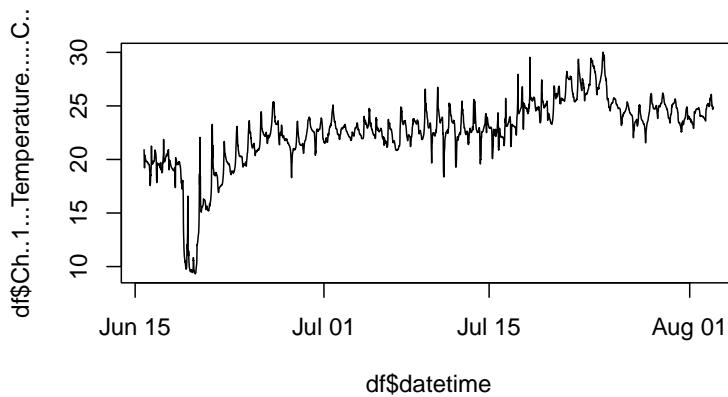
df$datetime <- as.POSIXct(strptime(df>Date.Time..EDT., format = "%m/%d/%Y %H:%M:%S"))
plot(df$datetime, df$Ch..3...pH....pH.)
```



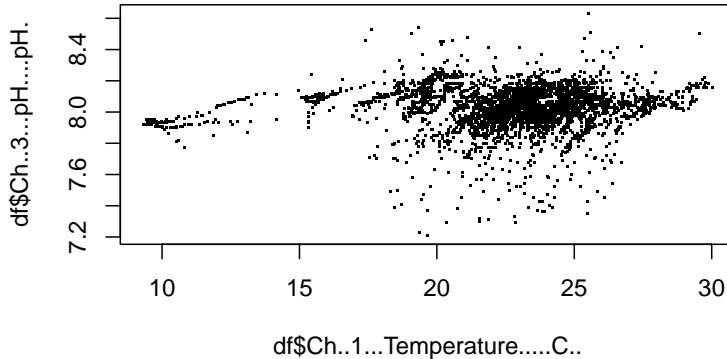
```
df <- df[df$datetime<"2022-08-03 00:08:00 EDT",]
plot(df$datetime, df$Ch..3...pH....pH., type = "l")
```



```
plot(df$datetime, df$Ch..1....Temperature.....C.., type = "l")
```



```
plot(df$Ch..1....Temperature.....C.., df$Ch..3...pH.....pH., pch = '.')
```



```
#Metadata
SN <- 560
tris <- 8.17
tris_temp <- 24.15

data_combined_new <- data.frame(
  Site = rep("Ptown",times = nrow(df)),
  Date = df$datetime,
  Temp = df$Ch..1...Temperature.....C.,
  pH = df$Ch..3...pH....pH.,
  Tris = rep(tris,length(df$datetime)),
  Tris_temp = rep(tris_temp,length(df$datetime)),
  Serial_number = rep(SN,length(df$datetime))
)

data_combined <- rbind(data_combined,data_combined_new)

setwd(wdPtown)
# August
(open_file <- filenames[6])
```

[1] "20573560_Ptown 2022-09-29 17_27_52 EDT (Data EDT).csv.orig"

```

open_file <- "Ptown_075 2022-08-29 21_52_54 EDT (Data EDT).csv"
df <- read.csv(open_file, stringsAsFactors = F)

df <- df[,c(1:5)]
head(df)

  X.      Date.Time..EDT. Ch..1...Temperature.....C.. Ch..2...Millivolts....mV.
1 1 08/02/2022 21:25:37                 NA                 NA
2 2 08/03/2022 08:00:00                23.04              175.90
3 3 08/03/2022 08:15:00                23.59              175.96
4 4 08/03/2022 08:30:00                23.98              176.02
5 5 08/03/2022 08:45:00                25.91              176.32
6 6 08/03/2022 09:00:00                25.48             -91.85
  Ch..3...pH....pH.
1                      NA
2                     3.56
3                     3.56
4                     3.57
5                     3.58
6                     8.28

names(df) <- c("n","Date.Time..EDT.", "Ch..1...Temperature.....C..","mV",
               "Ch..3...pH....pH." )

head(df)

  n      Date.Time..EDT. Ch..1...Temperature.....C..      mV Ch..3...pH....pH.
1 1 08/02/2022 21:25:37                 NA      NA                 NA
2 2 08/03/2022 08:00:00                23.04  175.90                 3.56
3 3 08/03/2022 08:15:00                23.59  175.96                 3.56
4 4 08/03/2022 08:30:00                23.98  176.02                 3.57
5 5 08/03/2022 08:45:00                25.91  176.32                 3.58
6 6 08/03/2022 09:00:00                25.48 -91.85                 8.28

tail(df)

  n      Date.Time..EDT. Ch..1...Temperature.....C..      mV
2552 2552 08/29/2022 21:30:00              24.22  156.68

```

```

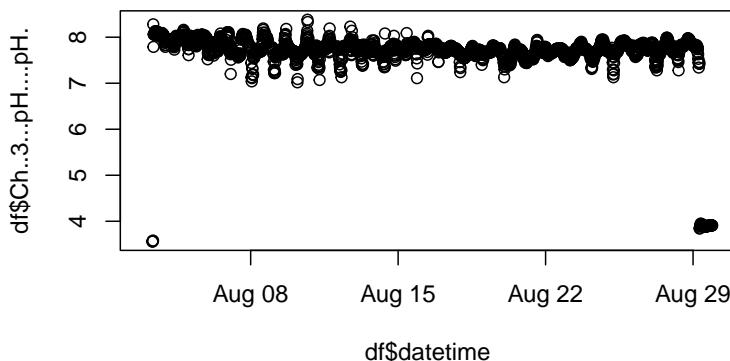
2553 2553 08/29/2022 21:45:00          24.29 156.66
2554 2554 08/29/2022 21:46:15          NA      NA
2555 2555 08/29/2022 21:46:17          NA      NA
2556 2556 08/29/2022 21:51:50          NA      NA
2557 2557 08/29/2022 21:51:51          NA      NA
Ch..3...pH....pH.
2552           3.91
2553           3.91
2554           NA
2555           NA
2556           NA
2557           NA

```

```

df$datetime <- as.POSIXct(strptime(df$date.Time..EDT., format = "%m/%d/%Y %H:%M:%S"))
plot(df$datetime, df$Ch..3...pH....pH.)

```



```

df <- df[df$datetime > "2022-08-04 00:00:00 EDT",]
df <- df[df$datetime < "2022-08-29 00:00:00 EDT",]
#plot(df$datetime, df$Ch..3...pH....pH., type = "l")
#plot(df$datetime, df$Ch..1...Temperature.....C., type = "l")
#plot(df$Ch..1...Temperature.....C., df$Ch..3...pH....pH., pch = '.')
#Metadata
SN <- 075
tris <- 8.14

```

```

tris_temp <- 25.11

data_combined_new <- data.frame(
  Site = rep("Ptown", times = nrow(df)),
  Date = df$datetime,
  Temp = df$Ch..1...Temperature....C.,
  pH = df$Ch..3...pH....pH.,
  Tris = rep(tris, length(df$datetime)),
  Tris_temp = rep(tris_temp, length(df$datetime)),
  Serial_number = rep(SN, length(df$datetime))
)

data_combined <- rbind(data_combined, data_combined_new)

```

SN560 8/28/22

Tris 8.20pH, 24.73C

```

setwd(wdPtown)
# September
# Also is here: "20573560_Ptown 2022-11-02 15_02_10 EDT (Data EDT).csv"
open_file <- "Ptown_20573560_Ptown 2022-09-29 17_27_52 EDT (Data EDT).csv"
df <- read.csv(open_file, stringsAsFactors = F)

df <- df[,c(1:5)]
head(df)

```

	X.	Date.Time..EDT.	Ch..1...Temperature....C..	Ch..2...Millivolts....mv.
1	1	08/28/2022 13:50:27	24.65	-181.07
2	2	08/28/2022 14:04:17	NA	NA
3	3	08/28/2022 14:04:18	NA	NA
4	4	08/28/2022 14:05:27	24.68	-82.24
5	5	08/28/2022 14:11:28	NA	NA
6	6	08/28/2022 14:11:29	NA	NA
		Ch..3...pH....pH.		
1		9.93		
2		NA		
3		NA		
4		8.20		

```

5          NA
6          NA

names(df) <- c("n","Date.Time..EDT.", "Ch..1...Temperature.....C..","mV",
              "Ch..3...pH....pH." )

head(df)

      n      Date.Time..EDT. Ch..1...Temperature.....C..      mV Ch..3...pH....pH.
1 1 08/28/2022 13:50:27           24.65 -181.07       9.93
2 2 08/28/2022 14:04:17           NA        NA        NA
3 3 08/28/2022 14:04:18           NA        NA        NA
4 4 08/28/2022 14:05:27           24.68  -82.24      8.20
5 5 08/28/2022 14:11:28           NA        NA        NA
6 6 08/28/2022 14:11:29           NA        NA        NA

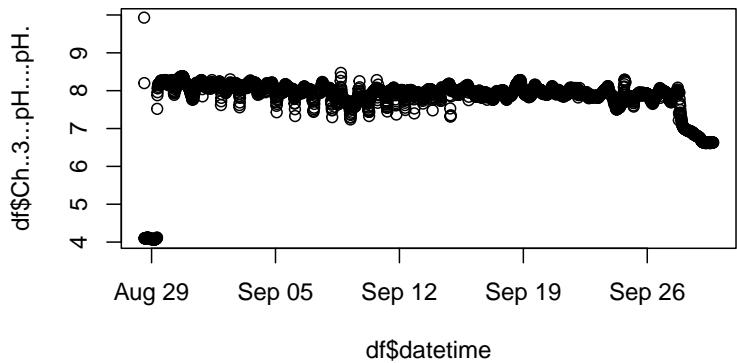
tail(df)

      n      Date.Time..EDT. Ch..1...Temperature.....C..      mV
3092 3092 09/29/2022 16:50:27           24.94  7.03
3093 3093 09/29/2022 17:05:27           24.87  7.09
3094 3094 09/29/2022 17:20:27           24.63  7.13
3095 3095 09/29/2022 17:27:26           NA      NA
3096 3096 09/29/2022 17:27:27           NA      NA
3097 3097 09/29/2022 17:27:28           NA      NA

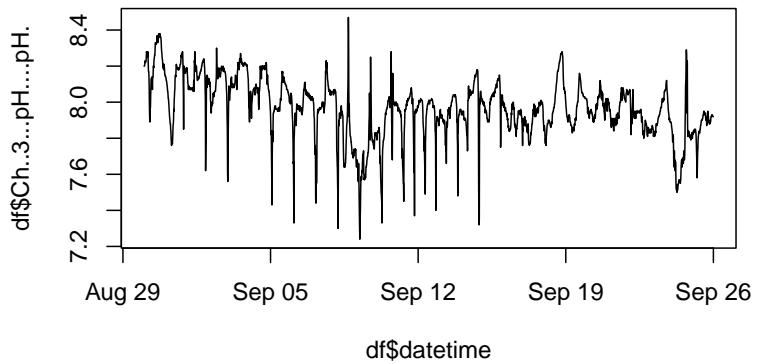
      Ch..3...pH....pH.
3092          6.63
3093          6.63
3094          6.63
3095          NA
3096          NA
3097          NA

df$datetime <- as.POSIXct(strptime(df$Date.Time..EDT., format = "%m/%d/%Y %H:%M:%S"))
plot(df$datetime, df$Ch..3...pH....pH.)

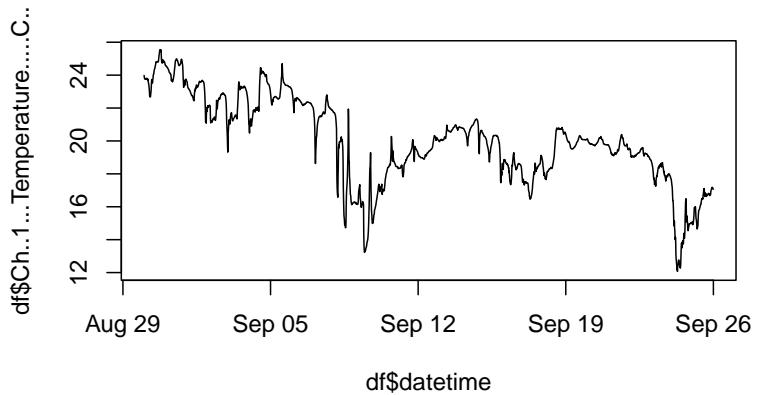
```



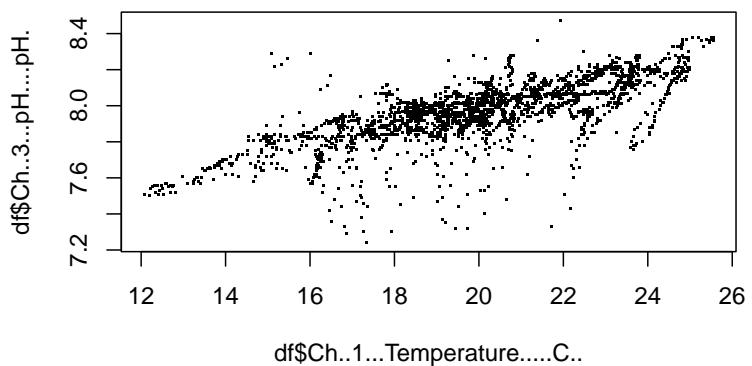
```
df <- df[df$datetime>"2022-08-30 00:00:00 EDT",]  
df <- df[df$datetime<"2022-09-26 00:00:00 EDT",]  
plot(df$datetime, df$Ch...3...pH....pH., type = "l")
```



```
plot(df$datetime, df$Ch..1....Temperature.....C.., type = "l")
```



```
plot(df$Ch..1...Temperature.....C., df$Ch..3...pH....pH., pch = '.')
```



```
#Metadata
SN <- 560
tris <- 8.20
tris_temp <- 24.73

data_combined_new <- data.frame(
  Site = rep("Ptown", times = nrow(df)),
  Date = df$datetime,
```

```

Temp = df$Ch..1...Temperature....C.,
pH = df$Ch..3...pH....pH.,
Tris = rep(tris,length(df$datetime)),
Tris_temp = rep(tris_temp,length(df$datetime)),
Serial_number = rep(SN,length(df$datetime))
)

data_combined <- rbind(data_combined,data_combined_new)

setwd(wdPtown)
# early October
# This sensor seems to be deployed at Eel pond originally, recalibrated without a change in the
# and then moved to Ptown. The segment at Ptown starts late in the day on 9/27/22
filenames

[1] "20573560_Ptown 2022-06-15 18_08_38 EDT (Data EDT).csv"
[2] "20573560_Ptown 2022-06-15 18_08_38 EDT (Data EDT).csv.orig"
[3] "20573560_Ptown 2022-08-05 17_08_58 EDT (Data EDT).csv"
[4] "20573560_Ptown 2022-08-05 17_08_58 EDT (Data EDT).csv.orig"
[5] "20573560_Ptown 2022-09-29 17_27_52 EDT (Data EDT).csv"
[6] "20573560_Ptown 2022-09-29 17_27_52 EDT (Data EDT).csv.orig"
[7] "20573560_Ptown 2022-11-02 15_02_10 EDT (Data EDT).csv"
[8] "20573560_Ptown 2023-05-10 14_13_07 EDT (Data EDT).csv"
[9] "20573560_Ptown 2023-05-10 14_13_07 EDT (Data EDT).csv.orig"
[10] "21333340_Eel_Pond 2022-11-07 15_11_16 EST (Data EST)_actuallyPtown.csv"
[11] "21333340_Eel_Pond 2022-11-07 15_11_16 EST (Data EST)_actuallyPtown.csv.orig"
[12] "340_Ptown 2023-06-11 14_35_41 EDT (Data EDT)(1).csv"
[13] "Ptown 2022-05-17 14_31_47 EDT (Data EDT) 2.csv"
[14] "Ptown 2022-05-17 14_31_47 EDT (Data EDT) 2.csv.orig"
[15] "Ptown 2022-05-17 14_31_47 EDT (Data EDT).csv"
[16] "Ptown 2022-05-17 14_31_47 EDT (Data EDT).csv.orig"
[17] "Ptown 2022-06-27 15_06_47 EDT (Data EDT).csv"
[18] "Ptown 2022-06-27 15_06_47 EDT (Data EDT).csv.orig"
[19] "Ptown_075 2022-08-29 21_52_54 EDT (Data EDT).csv"
[20] "Ptown_075 2022-08-29 21_52_54 EDT (Data EDT).csv.orig"
[21] "Ptown_076 2022-12-05 21_20_00 EST (Data EST).csv"
[22] "Ptown_076 2022-12-05 21_20_00 EST (Data EST).csv.orig"
[23] "Ptown_20573560_Ptown 2022-09-29 17_27_52 EDT (Data EDT).csv"
[24] "Ptown_20573560_Ptown 2022-09-29 17_27_52 EDT (Data EDT).csv.orig"

```

```
open_file <- "21333340_Eel_Pond 2022-11-07 15_11_16 EST (Data EST)_actuallyPtown.csv"
df <- read.csv(open_file, stringsAsFactors = F)
```

```
df <- df[,c(1:5)]
head(df)
```

	X. Date.Time..EST.EDT.	Ch..1...Temperature.....C..	Ch..2...Millivolts....mV.
1	08/28/2022 14:03:02	24.65	-178.86
2	08/28/2022 14:12:50	NA	NA
3	08/28/2022 14:12:51	NA	NA
4	08/28/2022 14:12:53	NA	NA
5	08/28/2022 14:18:02	24.63	152.60
6	08/28/2022 14:33:02	25.09	156.57
	Ch..3...pH....pH.		
1	9.93		
2	NA		
3	NA		
4	NA		
5	4.12		
6	4.06		

```
names(df) <- c("n","Date.Time..EST.", "Ch..1...Temperature.....C..","mV",
               "Ch..3...pH....pH." )
```

```
head(df)
```

n	Date.Time..EST.	Ch..1...Temperature.....C..	mV	Ch..3...pH....pH.
1	08/28/2022 14:03:02	24.65	-178.86	9.93
2	08/28/2022 14:12:50	NA	NA	NA
3	08/28/2022 14:12:51	NA	NA	NA
4	08/28/2022 14:12:53	NA	NA	NA
5	08/28/2022 14:18:02	24.63	152.60	4.12
6	08/28/2022 14:33:02	25.09	156.57	4.06

```
tail(df,80)
```

n	Date.Time..EST.	Ch..1...Temperature.....C..	mV
6768	6768 11/06/2022 20:18:02	16.08	-64.69

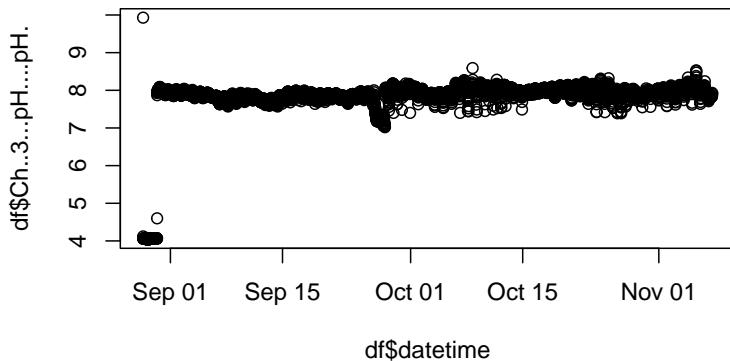
6769	6769	11/06/2022	20:33:02	16.06	-64.55
6770	6770	11/06/2022	20:48:02	16.08	-64.52
6771	6771	11/06/2022	21:03:02	16.08	-64.36
6772	6772	11/06/2022	21:18:02	16.06	-64.25
6773	6773	11/06/2022	21:33:02	16.03	-64.21
6774	6774	11/06/2022	21:48:02	16.01	-64.00
6775	6775	11/06/2022	22:03:02	15.99	-63.85
6776	6776	11/06/2022	22:18:02	15.99	-64.30
6777	6777	11/06/2022	22:33:02	15.99	-63.78
6778	6778	11/06/2022	22:48:02	16.01	-63.22
6779	6779	11/06/2022	23:03:02	16.01	-62.71
6780	6780	11/06/2022	23:18:02	16.01	-62.49
6781	6781	11/06/2022	23:33:02	16.01	-62.24
6782	6782	11/06/2022	23:48:02	16.03	-61.89
6783	6783	11/07/2022	00:03:02	16.06	-61.66
6784	6784	11/07/2022	00:18:02	16.06	-61.64
6785	6785	11/07/2022	00:33:02	16.06	-61.46
6786	6786	11/07/2022	00:48:02	16.06	-61.74
6787	6787	11/07/2022	01:03:02	16.06	-61.97
6788	6788	11/07/2022	01:18:02	16.03	-62.08
6789	6789	11/07/2022	01:33:02	16.01	-62.28
6790	6790	11/07/2022	01:48:02	15.99	-62.03
6791	6791	11/07/2022	02:03:02	15.99	-61.81
6792	6792	11/07/2022	02:18:02	15.96	-60.53
6793	6793	11/07/2022	02:33:02	15.94	-59.64
6794	6794	11/07/2022	02:48:02	15.91	-58.67
6795	6795	11/07/2022	03:03:02	15.87	-58.45
6796	6796	11/07/2022	03:18:02	15.84	-58.30
6797	6797	11/07/2022	03:33:02	15.82	-57.08
6798	6798	11/07/2022	03:48:02	15.80	-55.08
6799	6799	11/07/2022	04:03:02	15.75	-52.36
6800	6800	11/07/2022	04:18:02	15.72	-49.95
6801	6801	11/07/2022	04:33:02	15.75	-51.99
6802	6802	11/07/2022	04:48:02	15.77	-53.28
6803	6803	11/07/2022	05:03:02	15.75	-53.67
6804	6804	11/07/2022	05:18:02	15.80	-54.91
6805	6805	11/07/2022	05:33:02	15.84	-55.33
6806	6806	11/07/2022	05:48:02	15.87	-55.11
6807	6807	11/07/2022	06:03:02	15.89	-55.20
6808	6808	11/07/2022	06:18:02	15.89	-55.42
6809	6809	11/07/2022	06:33:02	15.91	-55.92

6810	6810	11/07/2022	06:48:02		15.91	-55.91
6811	6811	11/07/2022	07:03:02		15.94	-55.97
6812	6812	11/07/2022	07:18:02		15.94	-55.74
6813	6813	11/07/2022	07:33:02		15.96	-55.56
6814	6814	11/07/2022	07:48:02		15.96	-55.61
6815	6815	11/07/2022	08:03:02		15.99	-55.64
6816	6816	11/07/2022	08:18:02		16.01	-56.36
6817	6817	11/07/2022	08:33:02		16.03	-55.81
6818	6818	11/07/2022	08:48:02		16.06	-56.49
6819	6819	11/07/2022	09:03:02		16.06	-56.92
6820	6820	11/07/2022	09:18:02		16.06	-57.55
6821	6821	11/07/2022	09:33:02		16.08	-56.56
6822	6822	11/07/2022	09:48:02		16.11	-57.86
6823	6823	11/07/2022	10:03:02		16.11	-57.19
6824	6824	11/07/2022	10:18:02		16.11	-56.30
6825	6825	11/07/2022	10:33:02		16.13	-56.70
6826	6826	11/07/2022	10:48:02		16.15	-57.33
6827	6827	11/07/2022	11:03:02		16.18	-58.39
6828	6828	11/07/2022	11:18:02		16.22	-57.25
6829	6829	11/07/2022	11:33:02		16.25	-57.61
6830	6830	11/07/2022	11:48:02		16.25	-58.63
6831	6831	11/07/2022	12:03:02		16.22	-59.11
6832	6832	11/07/2022	12:18:02		16.22	-59.38
6833	6833	11/07/2022	12:33:02		16.22	-58.94
6834	6834	11/07/2022	12:48:02		16.27	-58.97
6835	6835	11/07/2022	13:03:02		16.30	-57.60
6836	6836	11/07/2022	13:18:02		16.32	-57.11
6837	6837	11/07/2022	13:33:02		16.32	-56.69
6838	6838	11/07/2022	13:48:02		16.34	-56.89
6839	6839	11/07/2022	14:03:02		16.42	-57.97
6840	6840	11/07/2022	14:18:02		16.53	-58.31
6841	6841	11/07/2022	14:33:02		16.82	-58.80
6842	6842	11/07/2022	14:48:02		17.08	-61.44
6843	6843	11/07/2022	15:03:02		17.08	-62.41
6844	6844	11/07/2022	15:10:04		NA	NA
6845	6845	11/07/2022	15:10:05		NA	NA
6846	6846	11/07/2022	15:10:09		NA	NA
6847	6847	11/07/2022	15:10:40		NA	NA
Ch..3...pH....pH.						
6768			7.97			
6769			7.97			

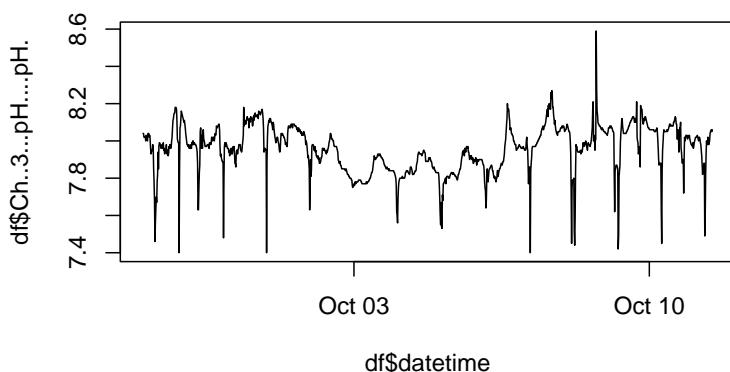
6770	7.97
6771	7.96
6772	7.96
6773	7.96
6774	7.96
6775	7.96
6776	7.96
6777	7.95
6778	7.94
6779	7.94
6780	7.93
6781	7.93
6782	7.92
6783	7.92
6784	7.92
6785	7.91
6786	7.92
6787	7.92
6788	7.92
6789	7.93
6790	7.92
6791	7.92
6792	7.90
6793	7.88
6794	7.86
6795	7.86
6796	7.86
6797	7.83
6798	7.80
6799	7.75
6800	7.71
6801	7.74
6802	7.77
6803	7.77
6804	7.80
6805	7.80
6806	7.80
6807	7.80
6808	7.80
6809	7.81
6810	7.81

6811	7.81
6812	7.81
6813	7.81
6814	7.81
6815	7.81
6816	7.82
6817	7.81
6818	7.82
6819	7.83
6820	7.84
6821	7.82
6822	7.85
6823	7.84
6824	7.82
6825	7.83
6826	7.84
6827	7.86
6828	7.84
6829	7.84
6830	7.86
6831	7.87
6832	7.87
6833	7.87
6834	7.87
6835	7.84
6836	7.83
6837	7.83
6838	7.83
6839	7.85
6840	7.85
6841	7.86
6842	7.91
6843	7.93
6844	NA
6845	NA
6846	NA
6847	NA

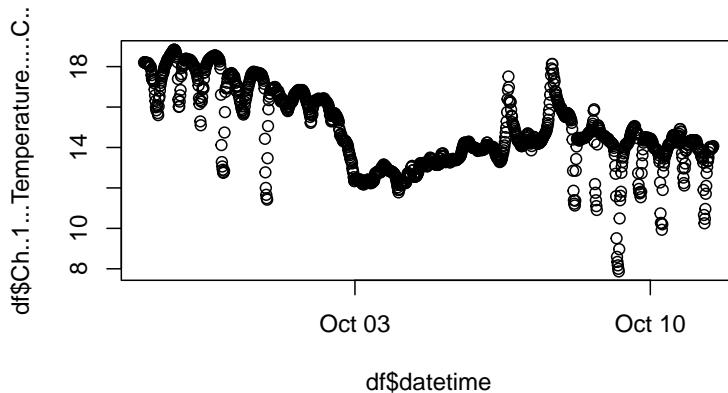
```
df$datetime <- as.POSIXct(strptime(df$Date.Time..EST., format = "%m/%d/%Y %H:%M:%S")) #Switch +  
plot(df$datetime, df$Ch..3...pH....pH.)
```



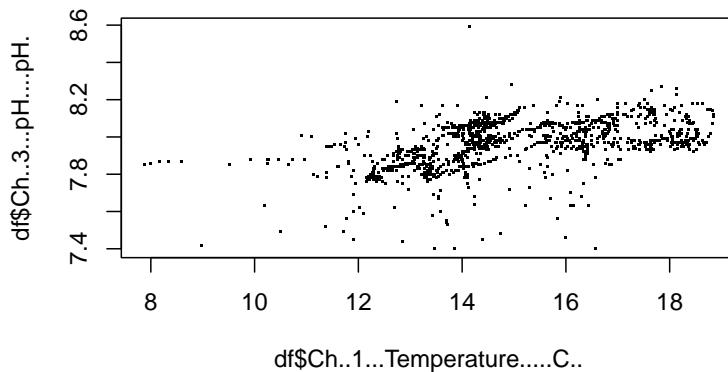
```
# df <- df[df$datetime>"2022-09-25 00:00:00 EDT",]
# df <- df[df$datetime<"2022-09-29 13:00:00 EDT",]
df <- df[df$datetime>"2022-09-28 00:00:00 EDT",]
df <- df[df$datetime<"2022-10-11 12:00:00 EDT",]
#df <- df[df$datetime<"2022-10-12 00:00:00 EDT",]
#df <- df[df$datetime<"2022-11-07 13:00:00 EDT",]
plot(df$datetime, df$Ch..3....pH....pH., type = "l")
```



```
plot(df$datetime, df$Ch..1....Temperature.....C.., type = "p")
```



```
plot(df$Ch..1...Temperature.....C., df$Ch..3...pH....pH., pch = '.')
```



```
#Metadata
SN <- 340
tris <- NA
tris_temp <- NA

data_combined_new <- data.frame(
  Site = rep("Ptown", times = nrow(df)),
  Date = df$datetime,
```

```

Temp = df$Ch..1...Temperature....C.,
pH = df$Ch..3...pH....pH.,
Tris = rep(tris,length(df$datetime)),
Tris_temp = rep(tris_temp,length(df$datetime)),
Serial_number = rep(SN,length(df$datetime))
)

data_combined <- rbind(data_combined,data_combined_new)

setwd(wdPtown)
# late October
# This sensor seems to be deployed at Eel pond originally, recalibrated without a change in the
# and then moved to Ptown. The segment at Ptown starts late in the day on 9/27/22
filenames

[1] "20573560_Ptown 2022-06-15 18_08_38 EDT (Data EDT).csv"
[2] "20573560_Ptown 2022-06-15 18_08_38 EDT (Data EDT).csv.orig"
[3] "20573560_Ptown 2022-08-05 17_08_58 EDT (Data EDT).csv"
[4] "20573560_Ptown 2022-08-05 17_08_58 EDT (Data EDT).csv.orig"
[5] "20573560_Ptown 2022-09-29 17_27_52 EDT (Data EDT).csv"
[6] "20573560_Ptown 2022-09-29 17_27_52 EDT (Data EDT).csv.orig"
[7] "20573560_Ptown 2022-11-02 15_02_10 EDT (Data EDT).csv"
[8] "20573560_Ptown 2023-05-10 14_13_07 EDT (Data EDT).csv"
[9] "20573560_Ptown 2023-05-10 14_13_07 EDT (Data EDT).csv.orig"
[10] "21333340_Eel_Pond 2022-11-07 15_11_16 EST (Data EST)_actuallyPtown.csv"
[11] "21333340_Eel_Pond 2022-11-07 15_11_16 EST (Data EST)_actuallyPtown.csv.orig"
[12] "340_Ptown 2023-06-11 14_35_41 EDT (Data EDT)(1).csv"
[13] "Ptown 2022-05-17 14_31_47 EDT (Data EDT) 2.csv"
[14] "Ptown 2022-05-17 14_31_47 EDT (Data EDT) 2.csv.orig"
[15] "Ptown 2022-05-17 14_31_47 EDT (Data EDT).csv"
[16] "Ptown 2022-05-17 14_31_47 EDT (Data EDT).csv.orig"
[17] "Ptown 2022-06-27 15_06_47 EDT (Data EDT).csv"
[18] "Ptown 2022-06-27 15_06_47 EDT (Data EDT).csv.orig"
[19] "Ptown_075 2022-08-29 21_52_54 EDT (Data EDT).csv"
[20] "Ptown_075 2022-08-29 21_52_54 EDT (Data EDT).csv.orig"
[21] "Ptown_076 2022-12-05 21_20_00 EST (Data EST).csv"
[22] "Ptown_076 2022-12-05 21_20_00 EST (Data EST).csv.orig"
[23] "Ptown_20573560_Ptown 2022-09-29 17_27_52 EDT (Data EDT).csv"
[24] "Ptown_20573560_Ptown 2022-09-29 17_27_52 EDT (Data EDT).csv.orig"

```

```

open_file <- "21333340_Eel_Pond 2022-11-07 15_11_16 EST (Data EST)_actuallyPtown.csv"
df <- read.csv(open_file, stringsAsFactors = F)

df <- df[,c(1:5)]
head(df)

X. Date.Time..EST.EDT. Ch..1...Temperature.....C.. Ch..2...Millivolts....mV.
1 1 08/28/2022 14:03:02 24.65 -178.86
2 2 08/28/2022 14:12:50 NA NA
3 3 08/28/2022 14:12:51 NA NA
4 4 08/28/2022 14:12:53 NA NA
5 5 08/28/2022 14:18:02 24.63 152.60
6 6 08/28/2022 14:33:02 25.09 156.57

Ch..3...pH....pH.
1 9.93
2 NA
3 NA
4 NA
5 4.12
6 4.06

names(df) <- c("n","Date.Time..EST.", "Ch..1...Temperature.....C..","mV",
               "Ch..3...pH....pH." )

head(df)

n      Date.Time..EST. Ch..1...Temperature.....C..      mV Ch..3...pH....pH.
1 1 08/28/2022 14:03:02 24.65 -178.86 9.93
2 2 08/28/2022 14:12:50 NA NA NA
3 3 08/28/2022 14:12:51 NA NA NA
4 4 08/28/2022 14:12:53 NA NA NA
5 5 08/28/2022 14:18:02 24.63 152.60 4.12
6 6 08/28/2022 14:33:02 25.09 156.57 4.06

tail(df,80)

n      Date.Time..EST. Ch..1...Temperature.....C..      mV

```

6768	6768	11/06/2022	20:18:02	16.08	-64.69
6769	6769	11/06/2022	20:33:02	16.06	-64.55
6770	6770	11/06/2022	20:48:02	16.08	-64.52
6771	6771	11/06/2022	21:03:02	16.08	-64.36
6772	6772	11/06/2022	21:18:02	16.06	-64.25
6773	6773	11/06/2022	21:33:02	16.03	-64.21
6774	6774	11/06/2022	21:48:02	16.01	-64.00
6775	6775	11/06/2022	22:03:02	15.99	-63.85
6776	6776	11/06/2022	22:18:02	15.99	-64.30
6777	6777	11/06/2022	22:33:02	15.99	-63.78
6778	6778	11/06/2022	22:48:02	16.01	-63.22
6779	6779	11/06/2022	23:03:02	16.01	-62.71
6780	6780	11/06/2022	23:18:02	16.01	-62.49
6781	6781	11/06/2022	23:33:02	16.01	-62.24
6782	6782	11/06/2022	23:48:02	16.03	-61.89
6783	6783	11/07/2022	00:03:02	16.06	-61.66
6784	6784	11/07/2022	00:18:02	16.06	-61.64
6785	6785	11/07/2022	00:33:02	16.06	-61.46
6786	6786	11/07/2022	00:48:02	16.06	-61.74
6787	6787	11/07/2022	01:03:02	16.06	-61.97
6788	6788	11/07/2022	01:18:02	16.03	-62.08
6789	6789	11/07/2022	01:33:02	16.01	-62.28
6790	6790	11/07/2022	01:48:02	15.99	-62.03
6791	6791	11/07/2022	02:03:02	15.99	-61.81
6792	6792	11/07/2022	02:18:02	15.96	-60.53
6793	6793	11/07/2022	02:33:02	15.94	-59.64
6794	6794	11/07/2022	02:48:02	15.91	-58.67
6795	6795	11/07/2022	03:03:02	15.87	-58.45
6796	6796	11/07/2022	03:18:02	15.84	-58.30
6797	6797	11/07/2022	03:33:02	15.82	-57.08
6798	6798	11/07/2022	03:48:02	15.80	-55.08
6799	6799	11/07/2022	04:03:02	15.75	-52.36
6800	6800	11/07/2022	04:18:02	15.72	-49.95
6801	6801	11/07/2022	04:33:02	15.75	-51.99
6802	6802	11/07/2022	04:48:02	15.77	-53.28
6803	6803	11/07/2022	05:03:02	15.75	-53.67
6804	6804	11/07/2022	05:18:02	15.80	-54.91
6805	6805	11/07/2022	05:33:02	15.84	-55.33
6806	6806	11/07/2022	05:48:02	15.87	-55.11
6807	6807	11/07/2022	06:03:02	15.89	-55.20
6808	6808	11/07/2022	06:18:02	15.89	-55.42

6809	6809	11/07/2022	06:33:02		15.91	-55.92
6810	6810	11/07/2022	06:48:02		15.91	-55.91
6811	6811	11/07/2022	07:03:02		15.94	-55.97
6812	6812	11/07/2022	07:18:02		15.94	-55.74
6813	6813	11/07/2022	07:33:02		15.96	-55.56
6814	6814	11/07/2022	07:48:02		15.96	-55.61
6815	6815	11/07/2022	08:03:02		15.99	-55.64
6816	6816	11/07/2022	08:18:02		16.01	-56.36
6817	6817	11/07/2022	08:33:02		16.03	-55.81
6818	6818	11/07/2022	08:48:02		16.06	-56.49
6819	6819	11/07/2022	09:03:02		16.06	-56.92
6820	6820	11/07/2022	09:18:02		16.06	-57.55
6821	6821	11/07/2022	09:33:02		16.08	-56.56
6822	6822	11/07/2022	09:48:02		16.11	-57.86
6823	6823	11/07/2022	10:03:02		16.11	-57.19
6824	6824	11/07/2022	10:18:02		16.11	-56.30
6825	6825	11/07/2022	10:33:02		16.13	-56.70
6826	6826	11/07/2022	10:48:02		16.15	-57.33
6827	6827	11/07/2022	11:03:02		16.18	-58.39
6828	6828	11/07/2022	11:18:02		16.22	-57.25
6829	6829	11/07/2022	11:33:02		16.25	-57.61
6830	6830	11/07/2022	11:48:02		16.25	-58.63
6831	6831	11/07/2022	12:03:02		16.22	-59.11
6832	6832	11/07/2022	12:18:02		16.22	-59.38
6833	6833	11/07/2022	12:33:02		16.22	-58.94
6834	6834	11/07/2022	12:48:02		16.27	-58.97
6835	6835	11/07/2022	13:03:02		16.30	-57.60
6836	6836	11/07/2022	13:18:02		16.32	-57.11
6837	6837	11/07/2022	13:33:02		16.32	-56.69
6838	6838	11/07/2022	13:48:02		16.34	-56.89
6839	6839	11/07/2022	14:03:02		16.42	-57.97
6840	6840	11/07/2022	14:18:02		16.53	-58.31
6841	6841	11/07/2022	14:33:02		16.82	-58.80
6842	6842	11/07/2022	14:48:02		17.08	-61.44
6843	6843	11/07/2022	15:03:02		17.08	-62.41
6844	6844	11/07/2022	15:10:04		NA	NA
6845	6845	11/07/2022	15:10:05		NA	NA
6846	6846	11/07/2022	15:10:09		NA	NA
6847	6847	11/07/2022	15:10:40		NA	NA

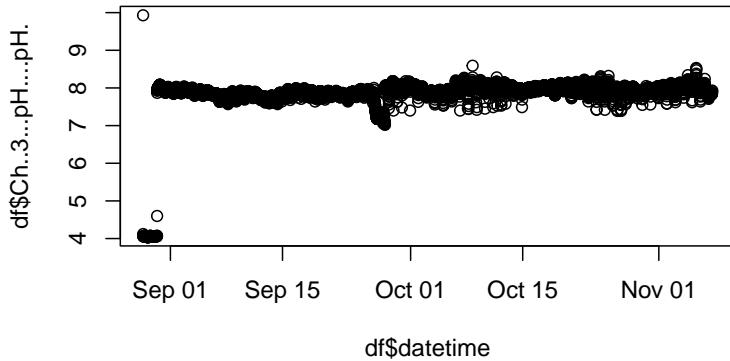
Ch..3...pH....pH.

6768 7.97

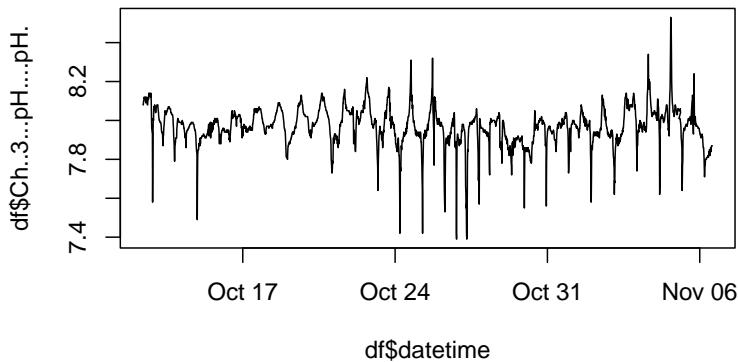
6769	7.97
6770	7.97
6771	7.96
6772	7.96
6773	7.96
6774	7.96
6775	7.96
6776	7.96
6777	7.95
6778	7.94
6779	7.94
6780	7.93
6781	7.93
6782	7.92
6783	7.92
6784	7.92
6785	7.91
6786	7.92
6787	7.92
6788	7.92
6789	7.93
6790	7.92
6791	7.92
6792	7.90
6793	7.88
6794	7.86
6795	7.86
6796	7.86
6797	7.83
6798	7.80
6799	7.75
6800	7.71
6801	7.74
6802	7.77
6803	7.77
6804	7.80
6805	7.80
6806	7.80
6807	7.80
6808	7.80
6809	7.81

6810	7.81
6811	7.81
6812	7.81
6813	7.81
6814	7.81
6815	7.81
6816	7.82
6817	7.81
6818	7.82
6819	7.83
6820	7.84
6821	7.82
6822	7.85
6823	7.84
6824	7.82
6825	7.83
6826	7.84
6827	7.86
6828	7.84
6829	7.84
6830	7.86
6831	7.87
6832	7.87
6833	7.87
6834	7.87
6835	7.84
6836	7.83
6837	7.83
6838	7.83
6839	7.85
6840	7.85
6841	7.86
6842	7.91
6843	7.93
6844	NA
6845	NA
6846	NA
6847	NA

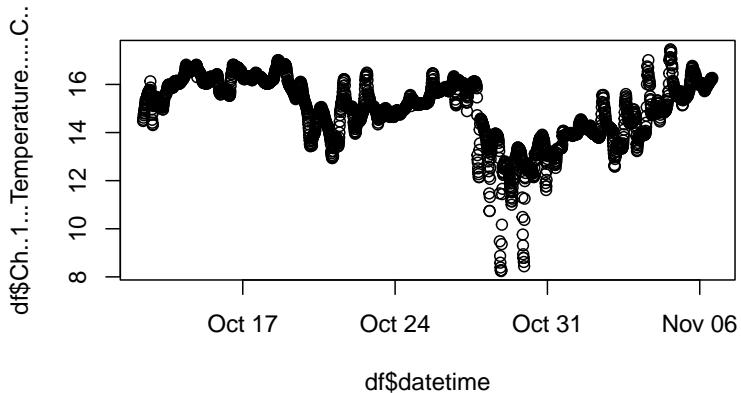
```
df$datetime <- as.POSIXct(strptime(df>Date.Time..EST., format = "%m/%d/%Y %H:%M:%S")) #Switch to EST  
plot(df$datetime, df$Ch..3....pH....pH.)
```



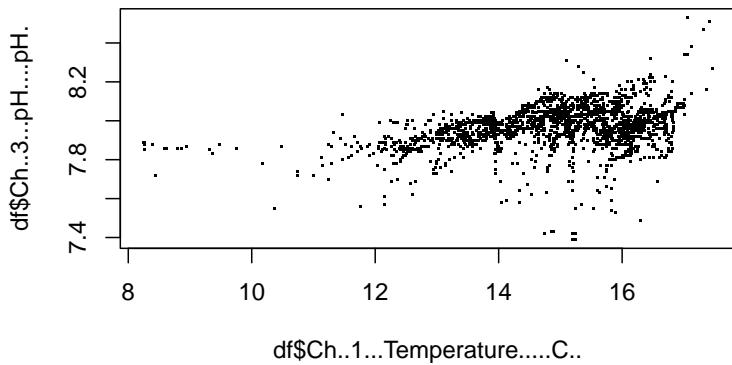
```
#df <- df[df$datetime>"2022-09-28 00:00:00 EDT",]  
#df <- df[df$datetime<"2022-10-11 06:00:00 EDT",]  
df <- df[df$datetime>"2022-10-12 10:00:00 EDT",]  
df <- df[df$datetime<"2022-11-07 13:00:00 EDT",]  
plot(df$datetime, df$Ch..3....pH....pH., type = "l")
```



```
plot(df$datetime, df$Ch..1...Temperature.....C., type = "p")
```



```
plot(df$Ch..1...Temperature.....C., df$Ch..3...pH....pH., pch = '.')
```



```
#Metadata  
SN <- 340  
tris <- NA  
tris_temp <- NA
```

```

data_combined_new <- data.frame(
  Site = rep("Ptown",times = nrow(df)),
  Date = df$datetime,
  Temp = df$Ch..1...Temperature.....C.,
  pH = df$Ch..3...pH....pH.,
  Tris = rep(tris,length(df$datetime)),
  Tris_temp = rep(tris_temp,length(df$datetime)),
  Serial_number = rep(SN,length(df$datetime))
)

data_combined <- rbind(data_combined,data_combined_new)

#Moved up from Eel. It seems like this sensor was previously an Eel Pond sensor but was moved to Ptown
#setwd(wdEel)
# November
#(open_file <- filenames[3])
open_file <- "20573560_Ptown 2022-12-05 16_55_15 EST (Data EST)_actuallyEel.csv"

df <- read.csv(open_file, stringsAsFactors = F)

df <- df[,c(1:5)]
head(df)

X. Date.Time..EST.EDT. Ch..1...Temperature.....C.. Ch..2...Millivolts....mv.
1 1 11/04/2022 15:00:00          29.51          179.72
2 2 11/04/2022 15:15:00          26.92          178.65
3 3 11/04/2022 15:30:00          23.33          176.79
4 4 11/04/2022 15:45:00          22.27          176.02
5 5 11/04/2022 16:00:00          20.98          175.36
6 6 11/04/2022 16:15:00          20.25          175.02
   Ch..3...pH....pH.
1             3.70
2             3.69
3             3.68
4             3.69
5             3.68
6             3.68

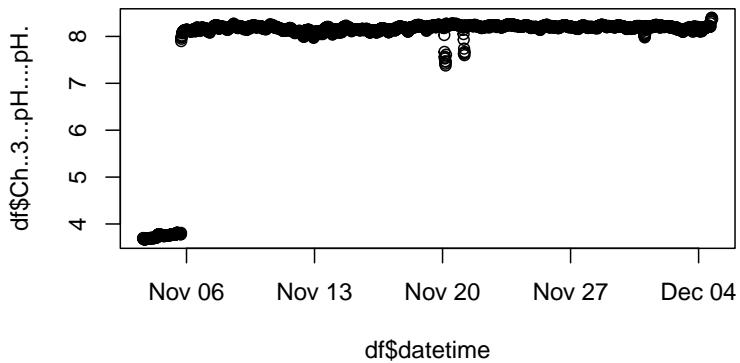
```

```

names(df) <- c("n", "Date.Time..EST.", "Ch..1...Temperature.....C..", "mV",
              "Ch..3...pH....pH." )

df$datetime <- as.POSIXct(strptime(df$Date.Time..EST., format = "%m/%d/%Y %H:%M:%S")) #Change +
plot(df$datetime, df$Ch..3...pH....pH.)

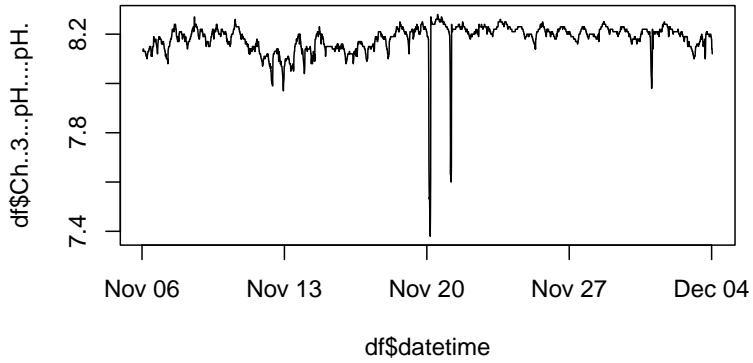
```



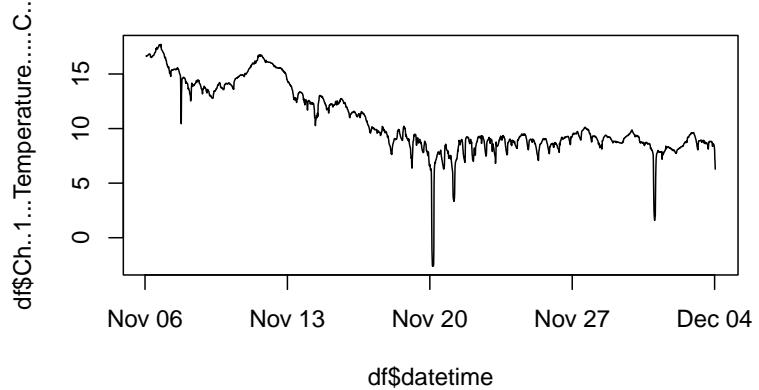
```

df <- df[df$datetime > "2022-11-07 00:00:00 EDT",]
df <- df[df$datetime < "2022-12-05 00:00:00 EDT",]
plot(df$datetime, df$Ch..3...pH....pH., type = "l")

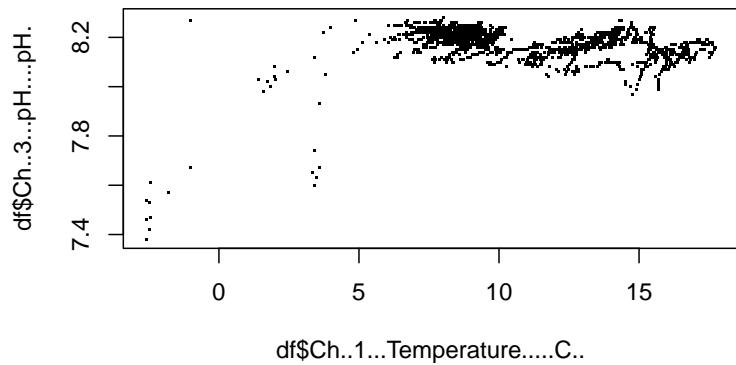
```



```
plot(df$datetime, df$Ch.1...Temperature.....C., type = "l")
```



```
plot(df$Ch.1...Temperature.....C., df$Ch.3...pH....pH., pch = '.')
```



```
#Metadata  
SN <- 560  
tris <- 8.30  
tris_temp <- 22.37
```

```

data_combined_new <- data.frame(
  Site = rep("Ptown",times = nrow(df)),
  Date = df$datetime,
  Temp = df$Ch..1...Temperature.....C.,
  pH = df$Ch..3...pH....pH.,
  Tris = rep(tris,length(df$datetime)),
  Tris_temp = rep(tris_temp,length(df$datetime)),
  Serial_number = rep(SN,length(df$datetime))
)

data_combined <- rbind(data_combined,data_combined_new)

```

```

setwd(wdPtown)
# skip to April to avoid freezing temps

open_file <- "20573560_Ptown 2023-05-10 14_13_07 EDT (Data EDT).csv"
df <- read.csv(open_file, stringsAsFactors = F)

df <- df[,c(1:5)]
head(df)

```

	X.	Date.Time..EDT.	Ch..1...Temperature.....C.	Ch..2...Millivolts....mv.
1	1	4/17/23 12:00	21.17	96.46
2	2	4/17/23 12:15	21.84	96.67
3	3	4/17/23 12:30	22.13	96.92
4	4	4/17/23 12:45	22.20	97.14
5	5	4/17/23 13:00	22.30	97.21
6	6	4/17/23 13:15	22.35	97.28
		Ch..3...pH....pH.		
1		4.82		
2		4.82		
3		4.82		
4		4.82		
5		4.82		
6		4.82		

```

names(df) <- c("n","Date.Time..EDT.", "Ch..1...Temperature.....C.", "mV",
              "Ch..3...pH....pH." )

```

```
head(df)
```

n	Date	Time..EDT.	Ch..1...Temperature.....C.	mV	Ch..3...pH....pH.	
1	1	4/17/23	12:00	21.17	96.46	4.82
2	2	4/17/23	12:15	21.84	96.67	4.82
3	3	4/17/23	12:30	22.13	96.92	4.82
4	4	4/17/23	12:45	22.20	97.14	4.82
5	5	4/17/23	13:00	22.30	97.21	4.82
6	6	4/17/23	13:15	22.35	97.28	4.82

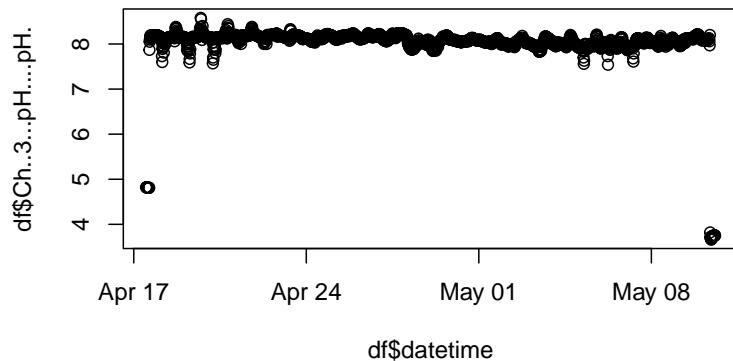
```
tail(df,80)
```

n	Date	Time..EDT.	Ch..1...Temperature.....C.	mV	Ch..3...pH....pH.	
2141	2141	5/9/23	19:00	13.14	-86.77	8.18
2142	2142	5/9/23	19:15	13.06	-87.05	8.19
2143	2143	5/9/23	19:30	13.02	-87.44	8.19
2144	2144	5/9/23	19:45	12.92	-87.80	8.20
2145	2145	5/9/23	20:00	12.85	-87.85	8.20
2146	2146	5/9/23	20:15	12.73	-87.71	8.20
2147	2147	5/9/23	20:30	12.63	-87.99	8.21
2148	2148	5/9/23	20:45	12.63	-88.58	8.22
2149	2149	5/9/23	21:00	12.46	-88.19	8.21
2150	2150	5/9/23	21:15	12.27	-87.89	8.20
2151	2151	5/9/23	21:30	12.17	-87.35	8.20
2152	2152	5/9/23	21:45	12.12	-87.25	8.19
2153	2153	5/9/23	22:00	12.19	-87.57	8.20
2154	2154	5/9/23	22:15	12.22	-87.17	8.19
2155	2155	5/9/23	22:30	12.19	-86.77	8.18
2156	2156	5/9/23	22:45	12.17	-85.92	8.17
2157	2157	5/9/23	23:00	12.15	-84.64	8.14
2158	2158	5/9/23	23:15	12.10	-84.61	8.14
2159	2159	5/9/23	23:30	12.12	-84.07	8.13
2160	2160	5/9/23	23:45	12.15	-83.60	8.12
2161	2161	5/10/23	0:00	12.17	-83.74	8.13
2162	2162	5/10/23	0:15	12.17	-83.30	8.12
2163	2163	5/10/23	0:30	12.15	-82.99	8.11
2164	2164	5/10/23	0:45	12.17	-82.52	8.10
2165	2165	5/10/23	1:00	12.19	-82.32	8.10
2166	2166	5/10/23	1:15	12.10	-81.21	8.08

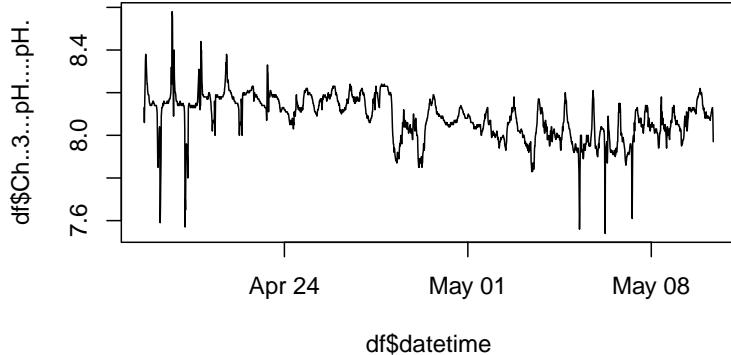
2167	2167	5/10/23 1:30	12.12	-81.31	8.08
2168	2168	5/10/23 1:45	12.19	-81.66	8.09
2169	2169	5/10/23 2:00	12.24	-82.55	8.10
2170	2170	5/10/23 2:15	12.32	-82.80	8.11
2171	2171	5/10/23 2:30	12.27	-81.71	8.09
2172	2172	5/10/23 2:45	12.24	-81.72	8.09
2173	2173	5/10/23 3:00	12.27	-81.38	8.08
2174	2174	5/10/23 3:15	12.32	-81.86	8.09
2175	2175	5/10/23 3:30	12.32	-81.69	8.09
2176	2176	5/10/23 3:45	12.34	-81.88	8.09
2177	2177	5/10/23 4:00	12.36	-81.89	8.09
2178	2178	5/10/23 4:15	12.36	-81.74	8.09
2179	2179	5/10/23 4:30	12.32	-81.39	8.08
2180	2180	5/10/23 4:45	12.29	-81.42	8.08
2181	2181	5/10/23 5:00	12.29	-81.27	8.08
2182	2182	5/10/23 5:15	12.32	-81.22	8.08
2183	2183	5/10/23 5:30	12.34	-81.41	8.08
2184	2184	5/10/23 5:45	12.34	-81.38	8.08
2185	2185	5/10/23 6:00	12.34	-80.89	8.07
2186	2186	5/10/23 6:15	12.34	-81.71	8.09
2187	2187	5/10/23 6:30	12.36	-82.03	8.09
2188	2188	5/10/23 6:45	12.36	-82.14	8.10
2189	2189	5/10/23 7:00	12.36	-82.82	8.11
2190	2190	5/10/23 7:15	12.34	-82.69	8.11
2191	2191	5/10/23 7:30	12.32	-83.28	8.12
2192	2192	5/10/23 7:45	12.34	-83.46	8.12
2193	2193	5/10/23 8:00	12.36	-83.66	8.13
2194	2194	5/10/23 8:15	12.41	-83.02	8.11
2195	2195	5/10/23 8:30	12.61	-80.03	8.06
2196	2196	5/10/23 8:45	13.06	-75.30	7.97
2197	2197	5/10/23 9:00	13.47	-87.85	8.20
2198	2198	5/10/23 9:15	16.32	148.97	3.82
2199	2199	5/10/23 9:30	20.46	156.52	3.72
2200	2200	5/10/23 9:45	17.42	156.58	3.69
2201	2201	5/10/23 10:00	16.30	156.50	3.68
2202	2202	5/10/23 10:15	15.70	157.13	3.66
2203	2203	5/10/23 10:30	15.49	156.11	3.67
2204	2204	5/10/23 10:45	15.58	154.60	3.70
2205	2205	5/10/23 11:00	16.39	154.71	3.71
2206	2206	5/10/23 11:15	17.37	154.55	3.72
2207	2207	5/10/23 11:30	18.27	154.33	3.74

2208	2208	5/10/23 11:45	18.41	154.27	3.74
2209	2209	5/10/23 12:00	18.77	154.22	3.75
2210	2210	5/10/23 12:15	19.17	154.60	3.74
2211	2211	5/10/23 12:30	18.91	155.43	3.73
2212	2212	5/10/23 12:45	19.32	155.05	3.74
2213	2213	5/10/23 13:00	19.67	154.75	3.75
2214	2214	5/10/23 13:15	20.01	154.91	3.75
2215	2215	5/10/23 13:30	20.17	154.99	3.75
2216	2216	5/10/23 13:45	20.01	154.44	3.76
2217	2217	5/10/23 14:00	19.86	154.52	3.75
2218	2218	5/10/23 14:12	NA	NA	NA
2219	2219	5/10/23 14:12	NA	NA	NA
2220	2220	5/10/23 14:12	NA	NA	NA

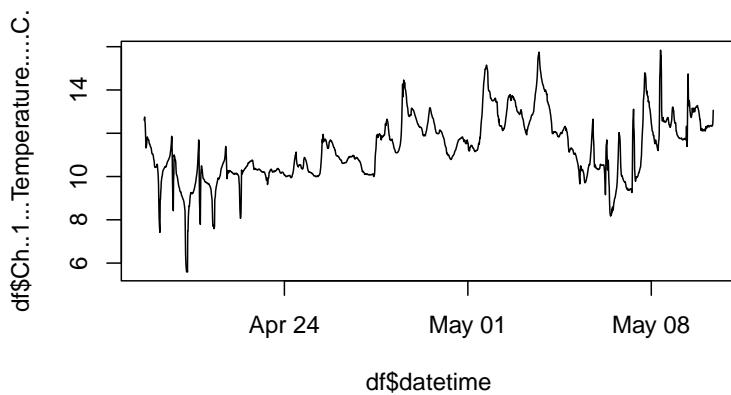
```
df$datetime <- as.POSIXct(strptime(df$date.Time..EDT., format = "%m/%d/%y %H:%M")) #Switch to L
plot(df$datetime, df$Ch..3...pH....pH.)
```



```
df <- df[df$datetime > "2023-04-18 15:15:15",]
df <- df[df$datetime < "2023-05-10 09:00:00",]
plot(df$datetime, df$Ch..3...pH....pH., type = "l")
```



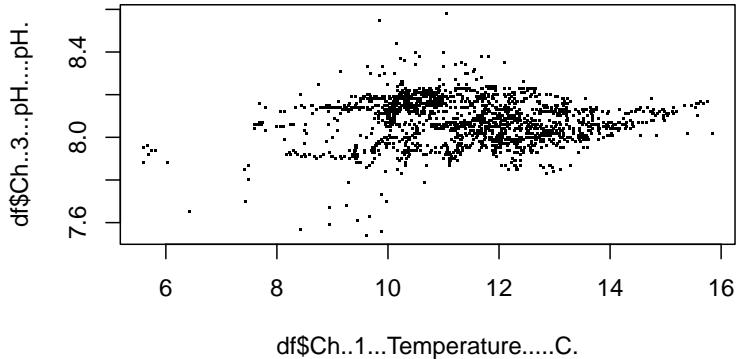
```
plot(df$datetime, df$Ch..1....Temperature.....C., type = "l")
```



```
length(df$Ch..1....Temperature.....C.)
```

[1] 2086

```
plot(df$Ch..1....Temperature.....C., df$Ch..3...pH....pH., pch = '.')
```



```

#Metadata
SN <- 560
tris <- 8.298 #Updated 12/23/23
tris_temp <- 21.4 #Updated 12/23/23

data_combined_new <- data.frame(
  Site = rep("Ptown",times = length(df$datetime)),
  Date = df$datetime,
  Temp = df$Ch..1...Temperature.....C.,
  pH = df$Ch..3...pH....pH.,
  Tris = rep(tris,length(df$datetime)),
  Tris_temp = rep(tris_temp,times = length(df$datetime)),
  Serial_number = rep(SN,times = length(df$datetime))
)

data_combined <- rbind(data_combined,data_combined_new)

setwd(wdPtown)
# skip to April to avoid freezing temps

open_file <- "340_Ptown 2023-06-11 14_35_41 EDT (Data EDT)(1).csv"
df <- read.csv(open_file, stringsAsFactors = F)

df <- df[,c(1:5)]
head(df)

```

```

X.      Date.Time..EDT. Ch.1...Temperature.....C. Ch.2...Millivolts....mV.
1 1 05/09/2023 16:11:07             NA             NA
2 2 05/09/2023 16:11:08             NA             NA
3 3 05/09/2023 16:11:10             NA             NA
4 4 05/10/2023 08:00:00            17.58          174.33
5 5 05/10/2023 08:15:00            17.39          174.15
6 6 05/10/2023 08:30:00            17.20          174.07

Ch.3...pH....pH.
1             NA
2             NA
3             NA
4            3.54
5            3.54
6            3.54

```

```

names(df) <- c("n","Date.Time..EDT.", "Ch..1...Temperature.....C.", "mV",
               "Ch.3...pH....pH." )

```

```
head(df)
```

n	Date.Time..EDT.	Ch..1...Temperature.....C.	mV	Ch.3...pH....pH.
1	1 05/09/2023 16:11:07		NA	NA
2	2 05/09/2023 16:11:08		NA	NA
3	3 05/09/2023 16:11:10		NA	NA
4	4 05/10/2023 08:00:00	17.58	174.33	3.54
5	5 05/10/2023 08:15:00	17.39	174.15	3.54
6	6 05/10/2023 08:30:00	17.20	174.07	3.54

```
tail(df,80)
```

n	Date.Time..EDT.	Ch..1...Temperature.....C.	mV
3026	3026 06/10/2023 19:30:00		15.22 -85.72
3027	3027 06/10/2023 19:45:00		15.25 -85.22
3028	3028 06/10/2023 20:00:00		15.27 -84.91
3029	3029 06/10/2023 20:15:00		15.29 -81.83
3030	3030 06/10/2023 20:30:00		15.34 -82.10
3031	3031 06/10/2023 20:45:00		15.34 -83.02
3032	3032 06/10/2023 21:00:00		15.37 -84.67
3033	3033 06/10/2023 21:15:00		15.41 -84.83

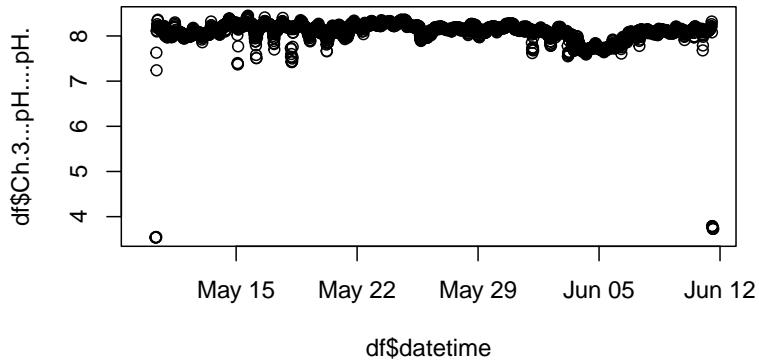
3034	3034	06/10/2023	21:30:00	15.46	-84.53
3035	3035	06/10/2023	21:45:00	15.53	-84.16
3036	3036	06/10/2023	22:00:00	15.70	-84.97
3037	3037	06/10/2023	22:15:00	15.91	-85.25
3038	3038	06/10/2023	22:30:00	16.13	-86.21
3039	3039	06/10/2023	22:45:00	16.20	-85.71
3040	3040	06/10/2023	23:00:00	16.18	-85.07
3041	3041	06/10/2023	23:15:00	16.11	-84.21
3042	3042	06/10/2023	23:30:00	15.96	-79.52
3043	3043	06/10/2023	23:45:00	15.77	-77.63
3044	3044	06/11/2023	00:00:00	15.44	-59.27
3045	3045	06/11/2023	00:15:00	15.53	-65.61
3046	3046	06/11/2023	00:30:00	15.65	-76.21
3047	3047	06/11/2023	00:45:00	15.53	-81.85
3048	3048	06/11/2023	01:00:00	15.60	-82.57
3049	3049	06/11/2023	01:15:00	15.70	-83.63
3050	3050	06/11/2023	01:30:00	15.72	-84.02
3051	3051	06/11/2023	01:45:00	15.68	-83.58
3052	3052	06/11/2023	02:00:00	15.65	-83.14
3053	3053	06/11/2023	02:15:00	15.58	-82.85
3054	3054	06/11/2023	02:30:00	15.58	-83.10
3055	3055	06/11/2023	02:45:00	15.60	-83.86
3056	3056	06/11/2023	03:00:00	15.60	-83.32
3057	3057	06/11/2023	03:15:00	15.60	-83.85
3058	3058	06/11/2023	03:30:00	15.63	-83.83
3059	3059	06/11/2023	03:45:00	15.65	-84.47
3060	3060	06/11/2023	04:00:00	15.63	-84.75
3061	3061	06/11/2023	04:15:00	15.60	-84.58
3062	3062	06/11/2023	04:30:00	15.60	-84.08
3063	3063	06/11/2023	04:45:00	15.58	-83.55
3064	3064	06/11/2023	05:00:00	15.58	-83.11
3065	3065	06/11/2023	05:15:00	15.58	-82.50
3066	3066	06/11/2023	05:30:00	15.56	-83.16
3067	3067	06/11/2023	05:45:00	15.56	-82.24
3068	3068	06/11/2023	06:00:00	15.53	-82.36
3069	3069	06/11/2023	06:15:00	15.53	-82.92
3070	3070	06/11/2023	06:30:00	15.49	-83.60
3071	3071	06/11/2023	06:45:00	15.49	-83.21
3072	3072	06/11/2023	07:00:00	15.46	-82.69
3073	3073	06/11/2023	07:15:00	15.44	-83.38
3074	3074	06/11/2023	07:30:00	15.46	-83.24

3075	3075	06/11/2023	07:45:00		15.46	-83.74
3076	3076	06/11/2023	08:00:00		15.49	-84.02
3077	3077	06/11/2023	08:15:00		15.49	-84.49
3078	3078	06/11/2023	08:30:00		15.51	-83.80
3079	3079	06/11/2023	08:45:00		15.56	-84.66
3080	3080	06/11/2023	09:00:00		15.60	-84.99
3081	3081	06/11/2023	09:15:00		15.68	-85.83
3082	3082	06/11/2023	09:30:00		15.77	-85.99
3083	3083	06/11/2023	09:45:00		15.82	-85.72
3084	3084	06/11/2023	10:00:00		15.89	-85.58
3085	3085	06/11/2023	10:15:00		15.99	-85.89
3086	3086	06/11/2023	10:30:00		16.18	-86.46
3087	3087	06/11/2023	10:45:00		16.51	-87.27
3088	3088	06/11/2023	11:00:00		17.01	-87.22
3089	3089	06/11/2023	11:15:00		17.51	-87.58
3090	3090	06/11/2023	11:30:00		18.32	-88.47
3091	3091	06/11/2023	11:45:00		19.13	-89.17
3092	3092	06/11/2023	12:00:00		19.98	-90.44
3093	3093	06/11/2023	12:15:00		21.13	-93.13
3094	3094	06/11/2023	12:30:00		22.23	-96.67
3095	3095	06/11/2023	12:45:00		22.90	-83.22
3096	3096	06/11/2023	13:00:00		24.39	164.54
3097	3097	06/11/2023	13:15:00		24.05	164.21
3098	3098	06/11/2023	13:30:00		23.35	164.32
3099	3099	06/11/2023	13:45:00		21.99	164.46
3100	3100	06/11/2023	14:00:00		22.13	166.18
3101	3101	06/11/2023	14:15:00		22.35	166.46
3102	3102	06/11/2023	14:30:00		22.54	166.85
3103	3103	06/11/2023	14:35:14		NA	NA
3104	3104	06/11/2023	14:35:15		NA	NA
3105	3105	06/11/2023	14:35:16		NA	NA
Ch.3...pH....pH.						
3026			8.15			
3027			8.15			
3028			8.14			
3029			8.08			
3030			8.09			
3031			8.11			
3032			8.13			
3033			8.14			
3034			8.13			

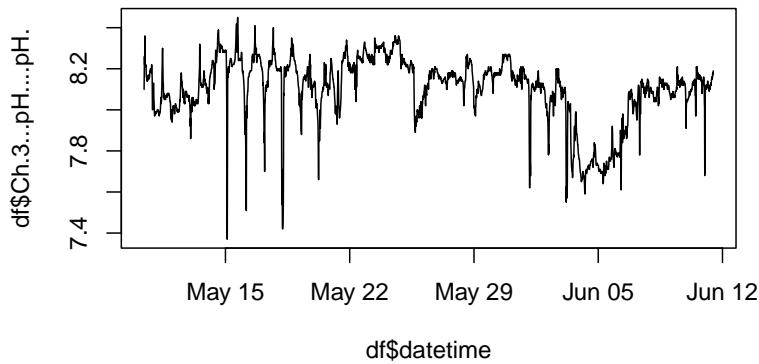
3035	8.12
3036	8.14
3037	8.14
3038	8.16
3039	8.15
3040	8.14
3041	8.12
3042	8.04
3043	8.01
3044	7.68
3045	7.79
3046	7.98
3047	8.08
3048	8.10
3049	8.11
3050	8.12
3051	8.11
3052	8.11
3053	8.10
3054	8.11
3055	8.12
3056	8.11
3057	8.12
3058	8.12
3059	8.13
3060	8.14
3061	8.13
3062	8.12
3063	8.11
3064	8.11
3065	8.10
3066	8.11
3067	8.09
3068	8.09
3069	8.10
3070	8.12
3071	8.11
3072	8.10
3073	8.11
3074	8.11
3075	8.12

3076	8.12
3077	8.13
3078	8.12
3079	8.13
3080	8.14
3081	8.15
3082	8.16
3083	8.15
3084	8.15
3085	8.15
3086	8.16
3087	8.18
3088	8.17
3089	8.18
3090	8.19
3091	8.20
3092	8.22
3093	8.26
3094	8.32
3095	8.08
3096	3.79
3097	3.79
3098	3.78
3099	3.76
3100	3.73
3101	3.73
3102	3.73
3103	NA
3104	NA
3105	NA

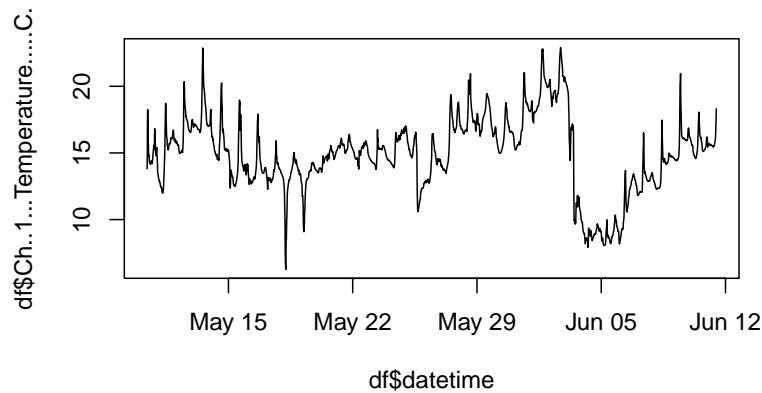
```
df$datetime <- as.POSIXct(strptime(df$date.Time..EDT., format = "%m/%d/%Y %H:%M")) #Switch to L
```



```
df <- df[df$datetime>"2023-05-10 09:45:00",]
df <- df[df$datetime<"2023-06-11 11:45:00",]
plot(df$datetime, df$Ch.3...pH..., type = "l")
```



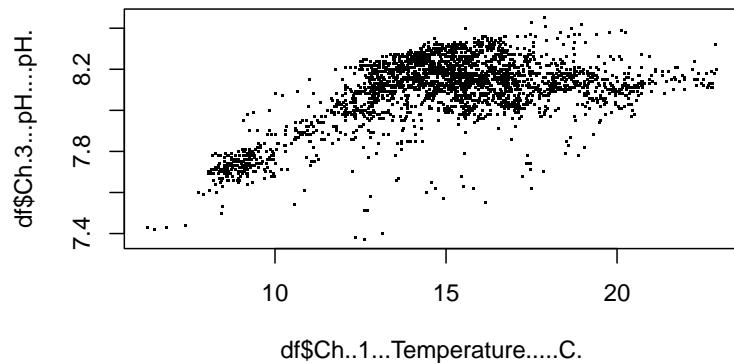
```
plot(df$datetime, df$Ch.1...Temperature....C., type = "l")
```



```
length(df$Ch..1...Temperature.....C.)
```

```
[1] 3079
```

```
plot(df$Ch..1...Temperature.....C., df$Ch.3...pH....pH., pch = '.')
```



```
#Metadata  
SN <- 340  
tris <- 8.19
```

```

tris_temp <- 21.3

data_combined_new <- data.frame(
  Site = rep("Ptown", times = nrow(df)),
  Date = df$datetime,
  Temp = df$Ch..1...Temperature....C.,
  pH = df$Ch.3...pH....pH.,
  Tris = rep(tris,length(df$datetime)),
  Tris_temp = rep(tris_temp,length(df$datetime)),
  Serial_number = rep(SN,length(df$datetime))
)

data_combined <- rbind(data_combined,data_combined_new)

```

0.1.3 Eel Pond

```

setwd(wdEel)
filenames <- dir()
filenames

[1] "20573560_Ptown 2022-12-05 16_55_15 EST (Data EST)_actuallyEel.csv"
[2] "20573560_Ptown 2022-12-05 16_55_15 EST (Data EST)_actuallyEel.csv.orig"
[3] "21333340_Eel_Pond 2022-06-13 08_27_36 EDT (Data EDT).csv"
[4] "21333340_Eel_Pond 2022-06-13 08_27_36 EDT (Data EDT).csv.orig"
[5] "21333340_Eel_Pond 2022-08-05 17_43_02 EDT (Data EDT).csv"
[6] "21333340_Eel_Pond 2022-08-05 17_43_02 EDT (Data EDT).csv.orig"
[7] "21333340_Eel_Pond 2022-09-11 18_27_18 EDT (Data EDT).csv"
[8] "21333340_Eel_Pond 2022-09-11 18_27_18 EDT (Data EDT).csv.orig"
[9] "21333340_Eel_Pond 2022-11-07 15_11_16 EST (Data EST)_actuallyPtown.csv"
[10] "21333340_Eel_Pond 2022-11-07 15_11_16 EST (Data EST)_actuallyPtown.csv.orig"
[11] "21333340_Eel_Pond 2023-04-14 15_22_46 EDT (Data EDT)(1).csv"
[12] "21333340_Eel_Pond 2023-05-08 09_10_54 EDT (Data EDT).csv"
[13] "21333340_Eel_Pond 2023-05-08 09_10_54 EDT (Data EDT).csv.orig"
[14] "Eel_076 2022-09-27 13_35_57 EDT (Data EDT).csv"
[15] "Eel_076 2022-09-27 13_35_57 EDT (Data EDT).csv.orig"
[16] "Eel_076 2022-11-07 09_15_44 EST (Data EST).xlsx"
[17] "Eel_076 2022-11-07 09_15_44 EST (Data EST).xlsx.orig"
[18] "Eel_076 2022-11-07 15_16_01 EST (Data EST)(1).csv"

```

```
[19] "Eel_076 2022-11-07 15_16_01 EST (Data EST)(1).csv.orig"  
[20] "Eel_076 2023-06-11 14_39_04 EDT (Data EDT).csv"
```

```
setwd(wdEel)  
#mid June to Aug  
#(open_file <- filenames[2])  
(open_file <- "21333340_Eel_Pond 2022-08-05 17_43_02 EDT (Data EDT).csv")
```

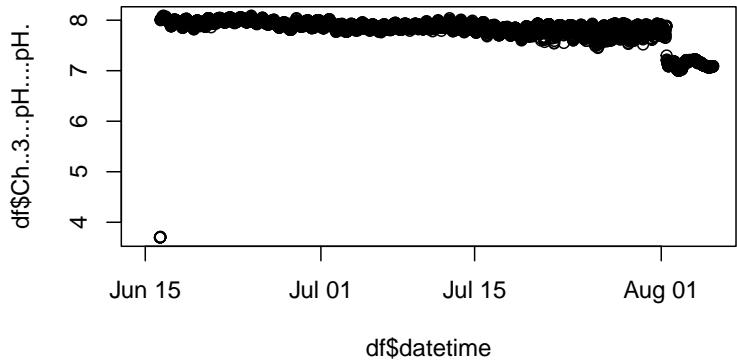
```
[1] "21333340_Eel_Pond 2022-08-05 17_43_02 EDT (Data EDT).csv"
```

```
df <- read.csv(open_file, stringsAsFactors = F)  
  
df <- df[,c(1:5)]  
head(df)
```

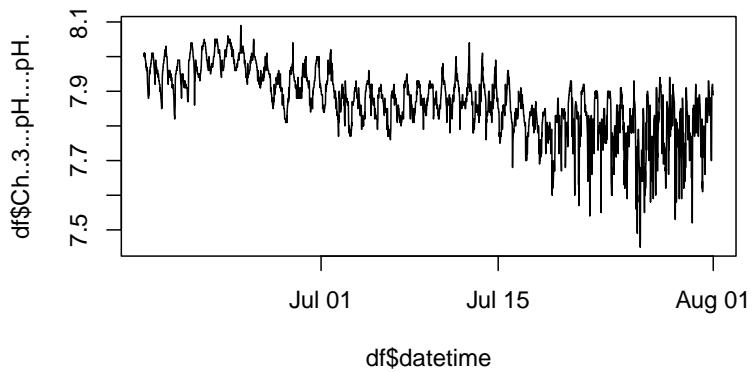
X.	Date.Time..EDT.	Ch..1...Temperature.....C..	Ch..2...Millivolts....mv.
1	06/14/2022 21:26:05	NA	NA
2	06/16/2022 08:00:00	17.53	177.04
3	06/16/2022 08:15:00	17.77	177.05
4	06/16/2022 08:30:00	17.94	177.04
5	06/16/2022 08:45:00	18.13	177.05
6	06/16/2022 09:00:00	18.27	177.02

X.	Ch..3...pH....pH.
1	NA
2	3.70
3	3.70
4	3.70
5	3.70
6	3.71

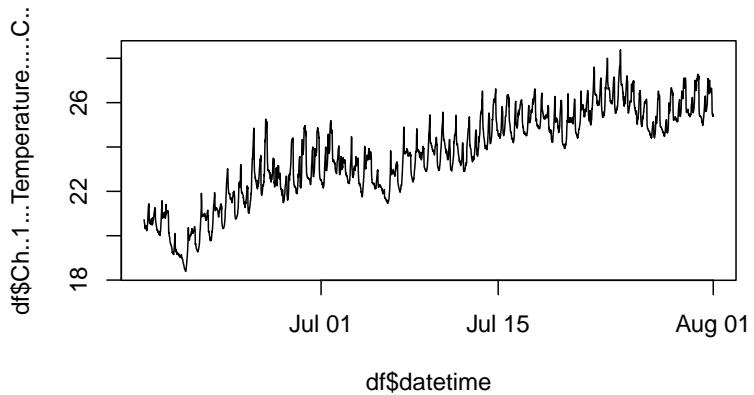
```
names(df) <- c("n","Date.Time..EDT.", "Ch..1...Temperature.....C..","mV",  
"Ch..3...pH....pH." )  
  
df$datetime <- as.POSIXct(strptime(df$Date.Time..EDT., format = "%m/%d/%Y %H:%M:%S"))  
plot(df$datetime, df$Ch..3...pH....pH.)
```



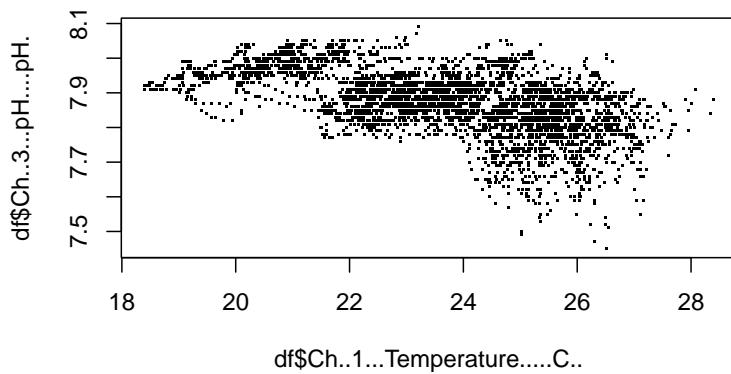
```
df <- df[df$datetime<"2022-08-01 00:09:00 EDT",]
df <- df[df$datetime>"2022-06-17 00:00:00 EDT",]
plot(df$datetime, df$Ch...3...pH....pH., type = "l")
```



```
plot(df$datetime, df$Ch..1....Temperature.....C.., type = "l")
```



```
plot(df$Ch..1...Temperature.....C., df$Ch..3...pH....pH., pch = '.')
```



```
#Metadata
SN <- 340
tris <- 8.14
tris_temp <- 24.73

data_combined_new <- data.frame(
  Site = rep("Eel",times = nrow(df)),
  Date = df$datetime,
```

```

Temp = df$Ch..1...Temperature.....C.,
pH = df$Ch..3...pH....pH.,
Tris = rep(tris,length(df$datetime)),
Tris_temp = rep(tris_temp,length(df$datetime)),
Serial_number = rep(SN,length(df$datetime))
)

data_combined <- rbind(data_combined,data_combined_new)
# Aug to lateAug is on logger that is still logging in Eel Pond and
# needs to be retrieved.
# There was an issue with the tool when swapping them out

```

```

setwd(wdEel)
#Aug to lateAug
(open_file <- filenames[4])

```

```

[1] "21333340_Eel_Pond 2022-06-13 08_27_36 EDT (Data EDT).csv.orig"

(open_file <- "Eel_076 2022-09-27 13_35_57 EDT (Data EDT).csv")

```

```
[1] "Eel_076 2022-09-27 13_35_57 EDT (Data EDT).csv"
```

```

df <- read.csv(open_file, stringsAsFactors = F)

df <- df[,c(1:5)]
head(df)

```

	X.	Date.Time..EDT.	Ch..1...Temperature.....C..	Ch..2...Millivolts....mv.
1	1	08/01/2022 06:00:00	24.53	181.61
2	2	08/01/2022 06:15:00	24.39	181.61
3	3	08/01/2022 06:30:00	23.30	181.57
4	4	08/01/2022 06:45:00	22.51	181.33
5	5	08/01/2022 07:00:00	21.94	181.11
6	6	08/01/2022 07:15:00	22.15	180.91
		Ch..3...pH....pH.		
1		3.62		
2		3.62		
3		3.61		

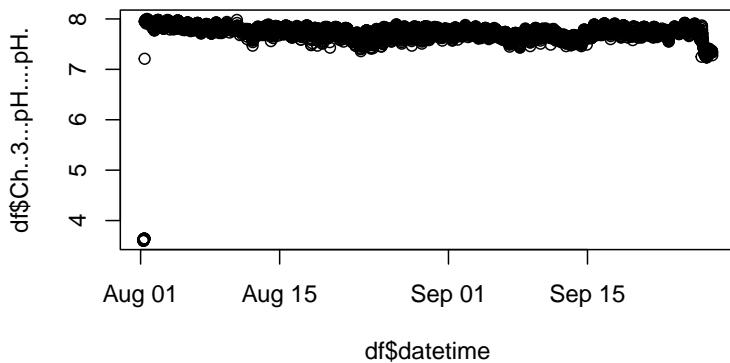
```

4          3.60
5          3.60
6          3.60

names(df) <- c("n","Date.Time..EDT.", "Ch..1...Temperature.....C..","mV",
               "Ch..3...pH....pH." )

df$datetime <- as.POSIXct(strptime(df>Date.Time..EDT., format = "%m/%d/%Y %H:%M:%S"))
plot(df$datetime, df$Ch..3...pH....pH.)

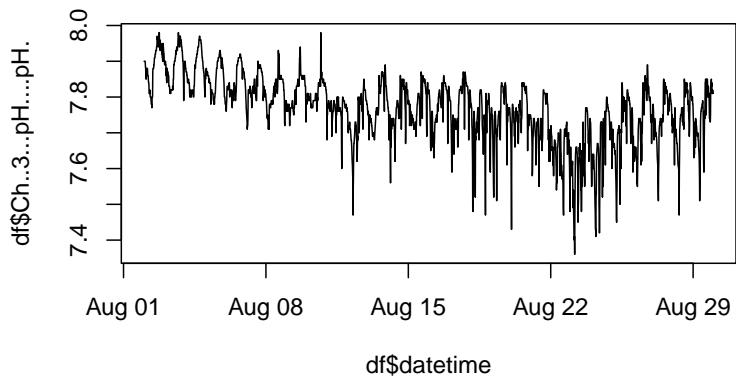
```



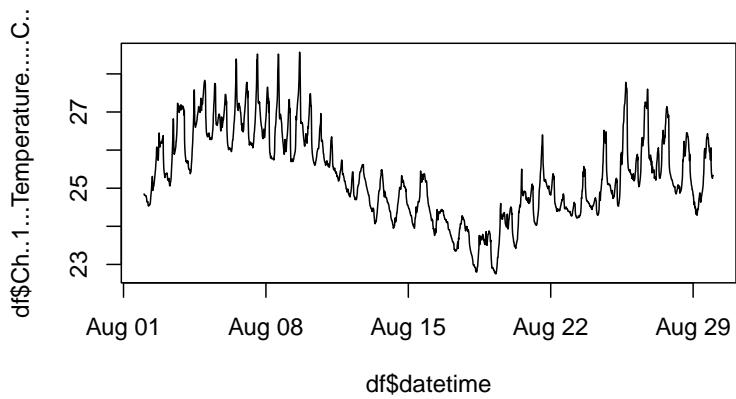
```

df <- df[df$datetime<"2022-08-30 00:00:00 EDT",]
df <- df[df$datetime>"2022-08-02 00:00:00 EDT",]
plot(df$datetime, df$Ch..3...pH....pH., type = "l")

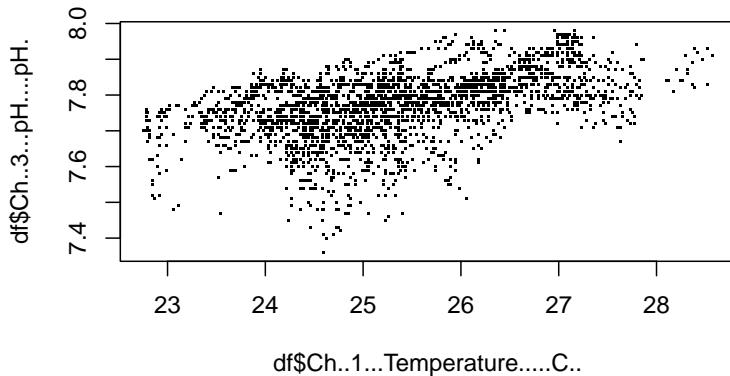
```



```
plot(df$datetime, df$Ch..1....Temperature.....C.., type = "l")
```



```
plot(df$Ch..1....Temperature.....C.., df$Ch..3...pH....pH., pch = '.')
```



```

#Metadata
SN <- 076
tris <- 8.15
tris_temp <- 24.85

data_combined_new <- data.frame(
  Site = rep("Eel",times = nrow(df)),
  Date = df$datetime,
  Temp = df$Ch..1...Temperature.....C.,
  pH = df$Ch..3...pH...pH.,
  Tris = rep(tris,length(df$datetime)),
  Tris_temp = rep(tris_temp,length(df$datetime)),
  Serial_number = rep(SN,length(df$datetime))
)

data_combined <- rbind(data_combined,data_combined_new)
# Aug to lateAug is on logger that is still logging in Eel Pond and
# needs to be retrieved.
# There was an issue with the tool when swapping them out

# # lateAug to mid Sep (was still logging and deployed -
# # but lost ~14 days of data due to calibrating before downloading data)
# (open_file <- filenames[3])
# open_file <- "21333340_Eel_Pond 2022-09-11 18_27_18 EDT (Data EDT).csv"
#

```

```

# df <- read.csv(open_file, stringsAsFactors = F)
# df$datetime <- as.POSIXct(strptime(df$date.Time..EDT., format = "%m/%d/%Y %H:%M:%S"))
# plot(df$datetime, df$Ch..3...pH....pH.)
# df <- df[df$datetime>"2022-08-31 00:00:00 EDT",]
# df <- df[df$datetime<"2022-09-11 00:00:00 EDT",]
# plot(df$datetime, df$Ch..3...pH....pH., type = "l")
# plot(df$datetime, df$Ch..1...Temperature.....C.., type = "l")
# plot(df$Ch..1...Temperature.....C.., df$Ch..3...pH....pH., pch = '.')
#
# data_combined_new <- data.frame(
#   Site = rep("Eel",times = nrow(df)),
#   Date = df$datetime,
#   Temp = df$Ch..1...Temperature.....C..,
#   pH = df$Ch..3...pH....pH.
# )
#
#
# data_combined <- rbind(data_combined,data_combined_new)

```

```

setwd(wdEel)
# #October???
# #October is not clear.
# #If the other legger was still logging and deployed at this site
# (open_file <- filenames[3])
# open_file <- "Eel_076 2022-11-07 15_16_01 EST (Data EST)(1).csv"
# head(df, 20)
# df <- read.csv(open_file, stringsAsFactors = F)
# df$datetime <- as.POSIXct(strptime(df$date.Time..EST., format = "%m/%d/%Y %H:%M:%S")) #Change
# plot(df$datetime, df$Ch..3...pH....pH.)
# df <- df[df$datetime>"2022-09-25 00:00:00 EDT",]
# # df <- df[df$datetime<"2022-09-30 00:00:00 EDT",]
#
# #df <- df[df$datetime<"2022-08-31 00:00:00 EDT",]
# #df <- df[df$datetime>"2022-08-26 00:00:00 EDT",]
# plot(df$datetime, df$Ch..3...pH....pH., type = "l")
# plot(df$datetime, df$Ch..1...Temperature.....C.., type = "p")
# plot(df$Ch..1...Temperature.....C.., df$Ch..3...pH....pH., pch = '.')
#
# data_combined_new <- data.frame(
#   Site = rep("Eel",times = nrow(df)),

```

```

#   Date = df$datetime,
#   Temp = df$Ch..1...Temperature....C.,
#   pH = df$Ch..3...pH....pH.
# )
#
#
# data_combined <- rbind(data_combined,data_combined_new)

# October
# This sensor seems to be deployed at Eel pond originally, recalibrated without a change in the
# and then moved to Ptown. The segment at Eel Pond ends on Monday 9/26/22.
# It was deployed on 8/31/22. Which is a day we were at Eel Pond
filenames

[1] "20573560_Ptown 2022-12-05 16_55_15 EST (Data EST)_actuallyEel.csv"
[2] "20573560_Ptown 2022-12-05 16_55_15 EST (Data EST)_actuallyEel.csv.orig"
[3] "21333340_Eel_Pond 2022-06-13 08_27_36 EDT (Data EDT).csv"
[4] "21333340_Eel_Pond 2022-06-13 08_27_36 EDT (Data EDT).csv.orig"
[5] "21333340_Eel_Pond 2022-08-05 17_43_02 EDT (Data EDT).csv"
[6] "21333340_Eel_Pond 2022-08-05 17_43_02 EDT (Data EDT).csv.orig"
[7] "21333340_Eel_Pond 2022-09-11 18_27_18 EDT (Data EDT).csv"
[8] "21333340_Eel_Pond 2022-09-11 18_27_18 EDT (Data EDT).csv.orig"
[9] "21333340_Eel_Pond 2022-11-07 15_11_16 EST (Data EST)_actuallyPtown.csv"
[10] "21333340_Eel_Pond 2022-11-07 15_11_16 EST (Data EST)_actuallyPtown.csv.orig"
[11] "21333340_Eel_Pond 2023-04-14 15_22_46 EDT (Data EDT)(1).csv"
[12] "21333340_Eel_Pond 2023-05-08 09_10_54 EDT (Data EDT).csv"
[13] "21333340_Eel_Pond 2023-05-08 09_10_54 EDT (Data EDT).csv.orig"
[14] "Eel_076 2022-09-27 13_35_57 EDT (Data EDT).csv"
[15] "Eel_076 2022-09-27 13_35_57 EDT (Data EDT).csv.orig"
[16] "Eel_076 2022-11-07 09_15_44 EST (Data EST).xlsx"
[17] "Eel_076 2022-11-07 09_15_44 EST (Data EST).xlsx.orig"
[18] "Eel_076 2022-11-07 15_16_01 EST (Data EST)(1).csv"
[19] "Eel_076 2022-11-07 15_16_01 EST (Data EST)(1).csv.orig"
[20] "Eel_076 2023-06-11 14_39_04 EDT (Data EDT).csv"

open_file <- "21333340_Eel_Pond 2022-11-07 15_11_16 EST (Data EST)_actuallyPtown.csv"
df <- read.csv(open_file, stringsAsFactors = F)

df <- df[,c(1:5)]
head(df)

```

```

X. Date.Time..EST.EDT. Ch..1...Temperature.....C.. Ch..2...Millivolts....mv.
1 1 08/28/2022 14:03:02           24.65          -178.86
2 2 08/28/2022 14:12:50           NA             NA
3 3 08/28/2022 14:12:51           NA             NA
4 4 08/28/2022 14:12:53           NA             NA
5 5 08/28/2022 14:18:02           24.63          152.60
6 6 08/28/2022 14:33:02           25.09          156.57

Ch..3...pH....pH.
1           9.93
2             NA
3             NA
4             NA
5           4.12
6           4.06

```

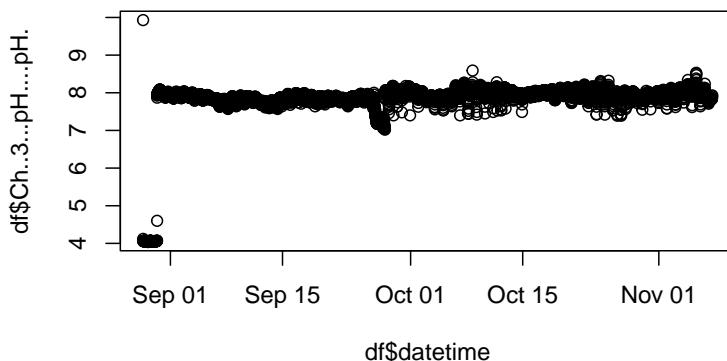
```

names(df) <- c("n", "Date.Time..EST.", "Ch..1...Temperature.....C..", "mV",
               "Ch..3...pH....pH." )

df$datetime <- as.POSIXct(strptime(df$Date.Time..EST., format = "%m/%d/%Y %H:%M:%S")) #Switch to UTC

plot(df$datetime, df$Ch..3...pH....pH.)

```

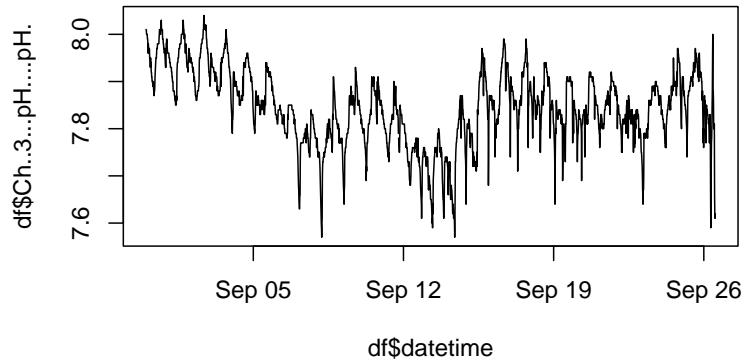


```

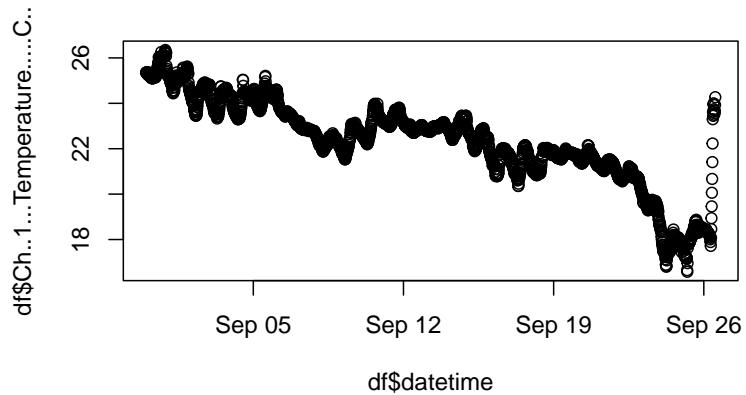
# df <- df[df$datetime>"2022-09-25 00:00:00 EDT",]
# df <- df[df$datetime<"2022-09-29 13:00:00 EDT",]
df <- df[df$datetime>"2022-08-31 00:00:00 EDT",]

```

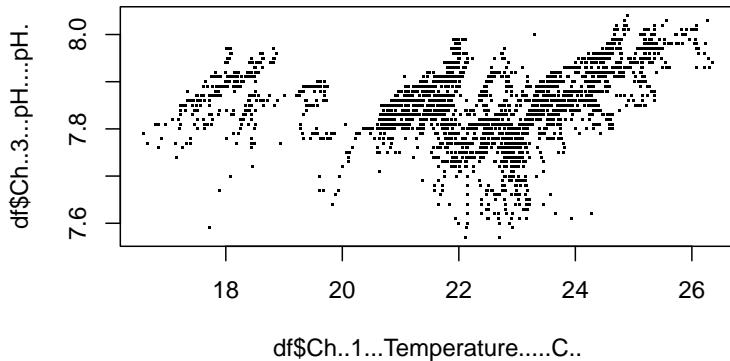
```
df <- df[df$datetime<"2022-09-26 13:00:00 EDT",]  
plot(df$datetime, df$Ch..3....pH....pH., type = "l")
```



```
plot(df$datetime, df$Ch..1....Temperature.....C.., type = "p")
```



```
plot(df$Ch..1....Temperature.....C.., df$Ch..3....pH....pH., pch = '.')
```



```
#Metadata
SN <- 340
tris <- 8.18
tris_temp <- 24.73

data_combined_new <- data.frame(
  Site = rep("Eel",times = nrow(df)),
  Date = df$datetime,
  Temp = df$Ch..1...Temperature.....C.,
  pH = df$Ch..3...pH....pH.,
  Tris = rep(tris,length(df$datetime)),
  Tris_temp = rep(tris_temp,length(df$datetime)),
  Serial_number = rep(SN,length(df$datetime))
)

data_combined <- rbind(data_combined,data_combined_new)

setwd(wdEel)
# Early October
#(open_file <- filenames[3])
open_file <- "Eel_076 2022-11-07 15_16_01 EST (Data EST)(1).csv"
head(df, 20)
```

n	Date.Time..EST.	Ch..1...Temperature.....C..	mV
236	08/31/2022 00:03:02	25.36	-69.21

237	237	08/31/2022	00:18:02	25.36	-69.19
238	238	08/31/2022	00:33:02	25.33	-68.96
239	239	08/31/2022	00:48:02	25.36	-68.94
240	240	08/31/2022	01:03:02	25.36	-68.55
241	241	08/31/2022	01:18:02	25.33	-68.39
242	242	08/31/2022	01:33:02	25.28	-68.25
243	243	08/31/2022	01:48:02	25.31	-68.05
244	244	08/31/2022	02:03:02	25.26	-67.47
245	245	08/31/2022	02:18:02	25.36	-66.60
246	246	08/31/2022	02:33:02	25.33	-67.31
247	247	08/31/2022	02:48:02	25.36	-67.22
248	248	08/31/2022	03:03:02	25.23	-66.75
249	249	08/31/2022	03:18:02	25.21	-66.53
250	250	08/31/2022	03:33:02	25.21	-66.77
251	251	08/31/2022	03:48:02	25.21	-66.72
252	252	08/31/2022	04:03:02	25.21	-65.96
253	253	08/31/2022	04:18:02	25.19	-65.75
254	254	08/31/2022	04:33:02	25.19	-65.58
255	255	08/31/2022	04:48:02	25.16	-64.85

	Ch..3...pH....pH.	datetime
236	8.01	2022-08-31 00:03:02
237	8.01	2022-08-31 00:18:02
238	8.00	2022-08-31 00:33:02
239	8.00	2022-08-31 00:48:02
240	8.00	2022-08-31 01:03:02
241	8.00	2022-08-31 01:18:02
242	7.99	2022-08-31 01:33:02
243	7.99	2022-08-31 01:48:02
244	7.98	2022-08-31 02:03:02
245	7.96	2022-08-31 02:18:02
246	7.98	2022-08-31 02:33:02
247	7.97	2022-08-31 02:48:02
248	7.97	2022-08-31 03:03:02
249	7.96	2022-08-31 03:18:02
250	7.97	2022-08-31 03:33:02
251	7.97	2022-08-31 03:48:02
252	7.95	2022-08-31 04:03:02
253	7.95	2022-08-31 04:18:02
254	7.95	2022-08-31 04:33:02
255	7.93	2022-08-31 04:48:02

```

df <- read.csv(open_file, stringsAsFactors = F)

df <- df[,c(1:5)]
head(df)

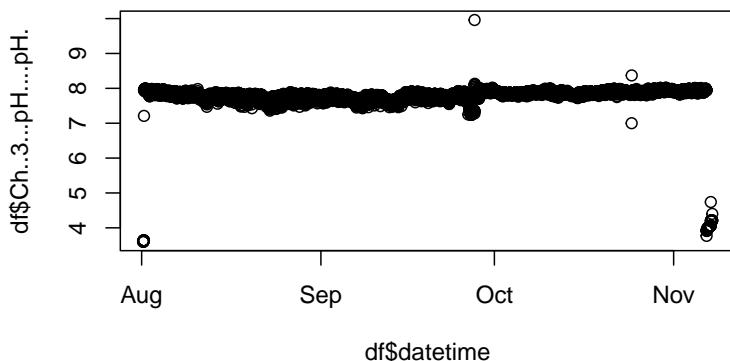
X. Date.Time..EST.EDT. Ch..1...Temperature.....C.. Ch..2...Millivolts....mv.
1 1 08/01/2022 06:00:00          24.53           181.61
2 2 08/01/2022 06:15:00          24.39           181.61
3 3 08/01/2022 06:30:00          23.30           181.57
4 4 08/01/2022 06:45:00          22.51           181.33
5 5 08/01/2022 07:00:00          21.94           181.11
6 6 08/01/2022 07:15:00          22.15           180.91

Ch..3...pH....pH.
1             3.62
2             3.62
3             3.61
4             3.60
5             3.60
6             3.60

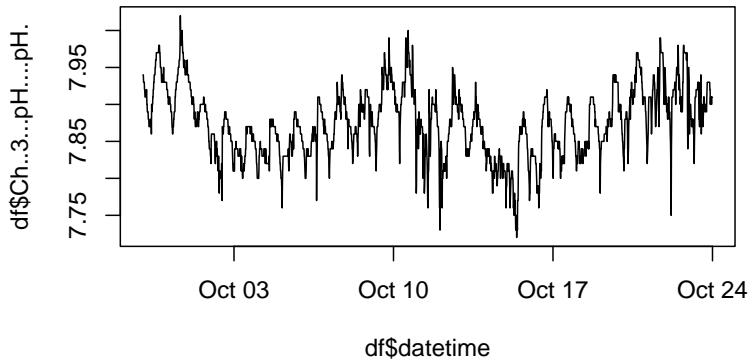
names(df) <- c("n","Date.Time..EST.", "Ch..1...Temperature.....C..","mV",
              "Ch..3...pH....pH." )

df$datetime <- as.POSIXct(strptime(df>Date.Time..EST., format = "%m/%d/%Y %H:%M:%S")) #Change to
plot(df$datetime, df$Ch..3...pH....pH.)

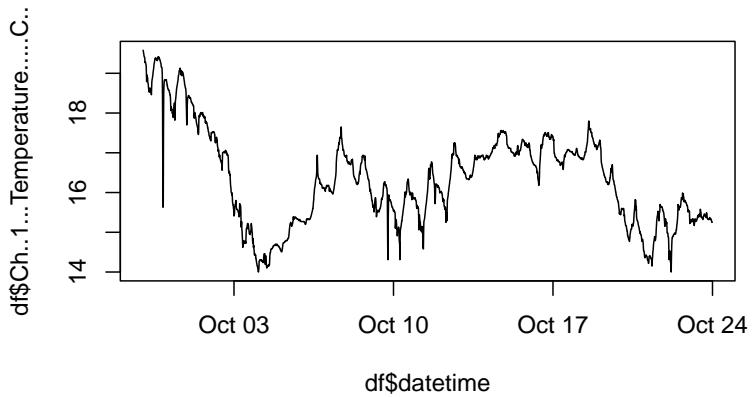
```



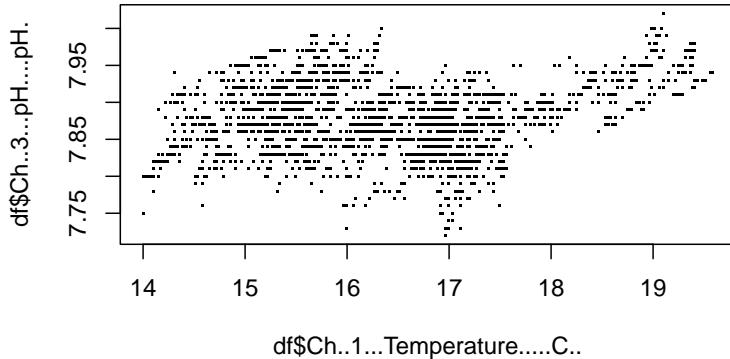
```
df <- df[df$datetime>"2022-09-29 00:00:00 EDT",]  
df <- df[df$datetime<"2022-10-24 00:00:00 EDT",]  
#df <- df[df$datetime<"2022-11-06 00:00:00 EDT",]  
plot(df$datetime, df$Ch...3...pH....pH., type = "l")
```



```
plot(df$datetime, df$Ch...1....Temperature.....C..., type = "l")
```



```
plot(df$Ch...1....Temperature.....C..., df$Ch...3...pH....pH., pch = '.')
```



```
#Metadata
SN <- 076
tris <- NA
tris_temp <- NA

data_combined_new <- data.frame(
  Site = rep("Eel",times = nrow(df)),
  Date = df$datetime,
  Temp = df$Ch..1...Temperature.....C.,
  pH = df$Ch..3...pH....pH.,
  Tris = rep(tris,length(df$datetime)),
  Tris_temp = rep(tris_temp,length(df$datetime)),
  Serial_number = rep(SN,length(df$datetime))
)

data_combined <- rbind(data_combined,data_combined_new)

setwd(wdEel)
# late October
#(open_file <- filenames[3])
open_file <- "Eel_076 2022-11-07 15_16_01 EST (Data EST)(1).csv"
head(df, 20)
```

n	Date.Time..EST.	Ch..1...Temperature.....C..	mV
5649	5649 09/29/2022 00:15:00	19.58	-62.19

5650	5650	09/29/2022	00:30:00	19.56	-61.97
5651	5651	09/29/2022	00:45:00	19.51	-61.81
5652	5652	09/29/2022	01:00:00	19.48	-61.56
5653	5653	09/29/2022	01:15:00	19.44	-61.56
5654	5654	09/29/2022	01:30:00	19.41	-61.50
5655	5655	09/29/2022	01:45:00	19.41	-61.30
5656	5656	09/29/2022	02:00:00	19.39	-60.91
5657	5657	09/29/2022	02:15:00	19.27	-60.49
5658	5658	09/29/2022	02:30:00	19.29	-60.61
5659	5659	09/29/2022	02:45:00	19.29	-60.67
5660	5660	09/29/2022	03:00:00	19.25	-60.66
5661	5661	09/29/2022	03:15:00	19.22	-60.64
5662	5662	09/29/2022	03:30:00	19.20	-60.20
5663	5663	09/29/2022	03:45:00	18.98	-60.78
5664	5664	09/29/2022	04:00:00	18.79	-60.92
5665	5665	09/29/2022	04:15:00	18.96	-59.92
5666	5666	09/29/2022	04:30:00	18.91	-59.64
5667	5667	09/29/2022	04:45:00	18.91	-59.28
5668	5668	09/29/2022	05:00:00	18.89	-59.06
		Ch..3...pH....pH.	datetime		
5649		7.94	2022-09-29 00:15:00		
5650		7.94	2022-09-29 00:30:00		
5651		7.93	2022-09-29 00:45:00		
5652		7.93	2022-09-29 01:00:00		
5653		7.93	2022-09-29 01:15:00		
5654		7.93	2022-09-29 01:30:00		
5655		7.92	2022-09-29 01:45:00		
5656		7.92	2022-09-29 02:00:00		
5657		7.91	2022-09-29 02:15:00		
5658		7.91	2022-09-29 02:30:00		
5659		7.91	2022-09-29 02:45:00		
5660		7.91	2022-09-29 03:00:00		
5661		7.91	2022-09-29 03:15:00		
5662		7.91	2022-09-29 03:30:00		
5663		7.92	2022-09-29 03:45:00		
5664		7.92	2022-09-29 04:00:00		
5665		7.90	2022-09-29 04:15:00		
5666		7.90	2022-09-29 04:30:00		
5667		7.89	2022-09-29 04:45:00		
5668		7.89	2022-09-29 05:00:00		

```

df <- read.csv(open_file, stringsAsFactors = F)

df <- df[,c(1:5)]
head(df)

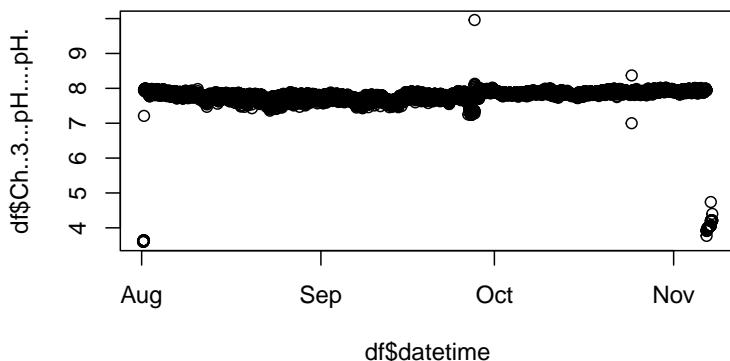
X. Date.Time..EST.EDT. Ch..1...Temperature.....C.. Ch..2...Millivolts....mv.
1 1 08/01/2022 06:00:00           24.53           181.61
2 2 08/01/2022 06:15:00           24.39           181.61
3 3 08/01/2022 06:30:00           23.30           181.57
4 4 08/01/2022 06:45:00           22.51           181.33
5 5 08/01/2022 07:00:00           21.94           181.11
6 6 08/01/2022 07:15:00           22.15           180.91

Ch..3...pH....pH.
1             3.62
2             3.62
3             3.61
4             3.60
5             3.60
6             3.60

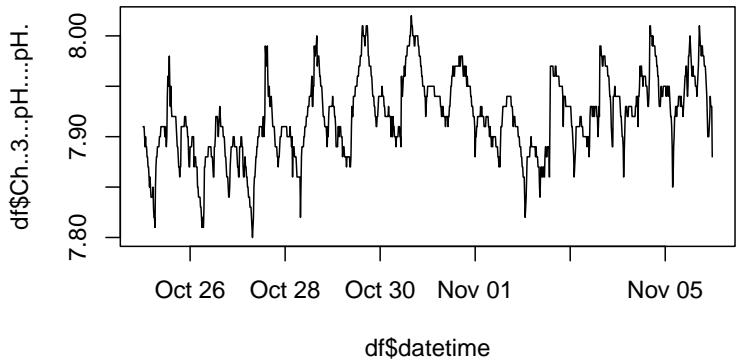
names(df) <- c("n","Date.Time..EST.", "Ch..1...Temperature.....C..","mV",
              "Ch..3...pH....pH." )

df$datetime <- as.POSIXct(strptime(df>Date.Time..EST., format = "%m/%d/%Y %H:%M:%S")) #Change to
plot(df$datetime, df$Ch..3...pH....pH.)

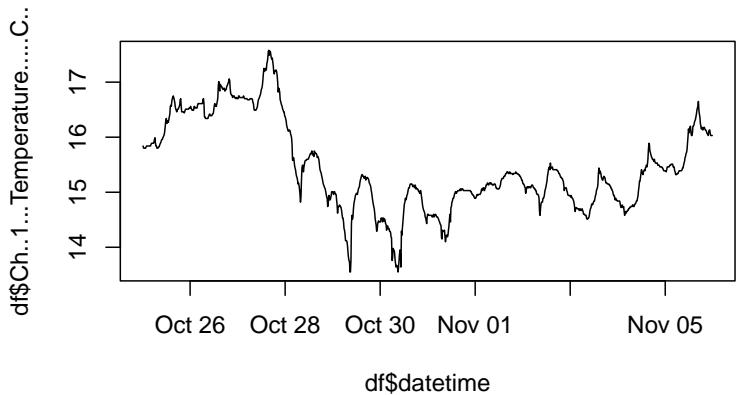
```



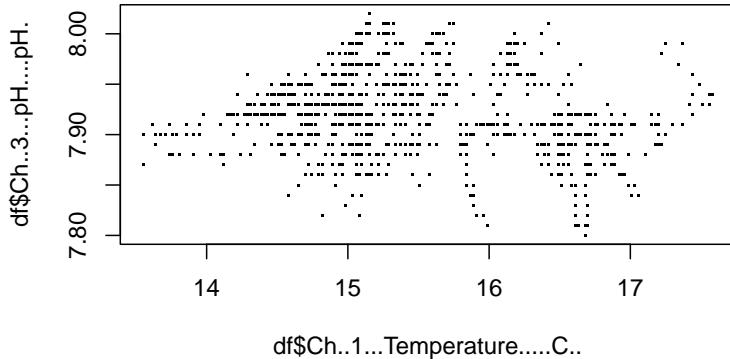
```
#df <- df[df$datetime>"2022-09-29 00:00:00 EDT",]
df <- df[df$datetime>"2022-10-25 00:00:00 EDT",]
df <- df[df$datetime<"2022-11-06 00:00:00 EDT",]
plot(df$datetime, df$Ch...3...pH....pH., type = "l")
```



```
plot(df$datetime, df$Ch..1....Temperature.....C.., type = "l")
```



```
plot(df$Ch..1....Temperature.....C.., df$Ch..3...pH....pH., pch = '.')
```



```

#Metadata
SN <- 076
tris <- 8.38
tris_temp <- 16.13

data_combined_new <- data.frame(
  Site = rep("Eel",times = nrow(df)),
  Date = df$datetime,
  Temp = df$Ch..1...Temperature.....C.,
  pH = df$Ch..3...pH....pH.,
  Tris = rep(tris,length(df$datetime)),
  Tris_temp = rep(tris_temp,length(df$datetime)),
  Serial_number = rep(SN,length(df$datetime))
)

data_combined <- rbind(data_combined,data_combined_new)

setwd(wdPtown)
# November
# This last file seems a bit out of place. However, I can confirm that this is Provincetown because it has the same header as the other files.

#I need to revisit this because the way I explained it above, the pH's don't match with the previous ones.

open_file <- "Ptown_076 2022-12-05 21_20_00 EST (Data EST).csv"
df <- read.csv(open_file, stringsAsFactors = F)

```

```
df <- df[,c(1:5)]  
head(df)
```

	X.	Date.Time..EST.	Ch..1...Temperature.....C..	Ch..2...Millivolts....mv.
1	1	11/07/2022 15:30:00	17.22	-74.78
2	2	11/07/2022 15:45:00	16.80	-80.85
3	3	11/07/2022 16:00:00	16.51	-83.35
4	4	11/07/2022 16:15:00	16.03	-83.33
5	5	11/07/2022 16:30:00	15.75	-80.56
6	6	11/07/2022 16:45:00	15.25	-70.85
	Ch..3...pH....pH.			
1		8.23		
2		8.34		
3		8.39		
4		8.39		
5		8.34		
6		8.17		

```
names(df) <- c("n","Date.Time..EST.", "Ch..1...Temperature.....C..","mV",  
"Ch..3...pH....pH." )
```

```
head(df)
```

n	Date.Time..EST.	Ch..1...Temperature.....C..	mV	Ch..3...pH....pH.
1	1 11/07/2022 15:30:00	17.22	-74.78	8.23
2	2 11/07/2022 15:45:00	16.80	-80.85	8.34
3	3 11/07/2022 16:00:00	16.51	-83.35	8.39
4	4 11/07/2022 16:15:00	16.03	-83.33	8.39
5	5 11/07/2022 16:30:00	15.75	-80.56	8.34
6	6 11/07/2022 16:45:00	15.25	-70.85	8.17

```
tail(df,80)
```

n	Date.Time..EST.	Ch..1...Temperature.....C..	mV
2637	2637 12/05/2022 02:30:00	17.68	180.02
2638	2638 12/05/2022 02:45:00	17.75	180.05
2639	2639 12/05/2022 03:00:00	17.84	180.08
2640	2640 12/05/2022 03:15:00	17.91	180.10

2641	2641	12/05/2022	03:30:00	17.99	180.13
2642	2642	12/05/2022	03:45:00	18.06	180.15
2643	2643	12/05/2022	04:00:00	18.15	180.18
2644	2644	12/05/2022	04:15:00	18.22	180.21
2645	2645	12/05/2022	04:30:00	18.30	180.22
2646	2646	12/05/2022	04:45:00	18.37	180.26
2647	2647	12/05/2022	05:00:00	18.46	180.29
2648	2648	12/05/2022	05:15:00	18.53	180.32
2649	2649	12/05/2022	05:30:00	18.60	180.33
2650	2650	12/05/2022	05:45:00	18.68	180.36
2651	2651	12/05/2022	06:00:00	18.75	180.40
2652	2652	12/05/2022	06:15:00	18.79	180.41
2653	2653	12/05/2022	06:30:00	18.87	180.41
2654	2654	12/05/2022	06:45:00	18.91	180.43
2655	2655	12/05/2022	07:00:00	18.96	180.44
2656	2656	12/05/2022	07:15:00	19.01	180.44
2657	2657	12/05/2022	07:30:00	19.03	180.46
2658	2658	12/05/2022	07:45:00	19.08	180.46
2659	2659	12/05/2022	08:00:00	19.13	180.46
2660	2660	12/05/2022	08:15:00	19.15	180.46
2661	2661	12/05/2022	08:30:00	19.20	180.46
2662	2662	12/05/2022	08:45:00	19.22	180.47
2663	2663	12/05/2022	09:00:00	19.25	180.47
2664	2664	12/05/2022	09:15:00	19.29	180.47
2665	2665	12/05/2022	09:30:00	19.32	180.47
2666	2666	12/05/2022	09:45:00	19.37	180.49
2667	2667	12/05/2022	10:00:00	19.41	180.49
2668	2668	12/05/2022	10:15:00	19.34	180.47
2669	2669	12/05/2022	10:30:00	19.32	180.46
2670	2670	12/05/2022	10:45:00	19.22	180.40
2671	2671	12/05/2022	11:00:00	18.84	180.29
2672	2672	12/05/2022	11:15:00	18.58	180.26
2673	2673	12/05/2022	11:30:00	18.51	180.22
2674	2674	12/05/2022	11:45:00	18.53	180.16
2675	2675	12/05/2022	12:00:00	18.58	180.11
2676	2676	12/05/2022	12:15:00	18.68	180.10
2677	2677	12/05/2022	12:30:00	18.10	179.46
2678	2678	12/05/2022	12:45:00	17.15	178.94
2679	2679	12/05/2022	13:00:00	16.13	178.41
2680	2680	12/05/2022	13:15:00	15.32	177.91
2681	2681	12/05/2022	13:30:00	14.53	177.49

2682	2682	12/05/2022	13:45:00		14.03	177.11
2683	2683	12/05/2022	14:00:00		14.22	177.07
2684	2684	12/05/2022	14:15:00		14.46	177.07
2685	2685	12/05/2022	14:30:00		14.74	177.08
2686	2686	12/05/2022	14:45:00		14.96	177.08
2687	2687	12/05/2022	15:00:00		15.10	177.10
2688	2688	12/05/2022	15:15:00		15.13	177.08
2689	2689	12/05/2022	15:30:00		15.03	177.05
2690	2690	12/05/2022	15:45:00		14.84	176.99
2691	2691	12/05/2022	16:00:00		14.58	176.91
2692	2692	12/05/2022	16:15:00		14.24	176.79
2693	2693	12/05/2022	16:30:00		13.83	176.63
2694	2694	12/05/2022	16:45:00		13.35	176.44
2695	2695	12/05/2022	17:00:00		12.61	176.13
2696	2696	12/05/2022	17:15:00		11.83	175.86
2697	2697	12/05/2022	17:30:00		11.95	175.77
2698	2698	12/05/2022	17:45:00		12.44	175.76
2699	2699	12/05/2022	18:00:00		12.92	175.82
2700	2700	12/05/2022	18:15:00		13.35	175.88
2701	2701	12/05/2022	18:30:00		13.88	175.97
2702	2702	12/05/2022	18:45:00		14.41	176.11
2703	2703	12/05/2022	19:00:00		14.86	176.24
2704	2704	12/05/2022	19:15:00		15.22	176.40
2705	2705	12/05/2022	19:30:00		15.41	176.52
2706	2706	12/05/2022	19:45:00		15.53	176.63
2707	2707	12/05/2022	20:00:00		15.70	176.74
2708	2708	12/05/2022	20:15:00		15.84	176.83
2709	2709	12/05/2022	20:30:00		15.89	176.91
2710	2710	12/05/2022	20:45:00		15.91	176.93
2711	2711	12/05/2022	21:00:00		15.72	176.94
2712	2712	12/05/2022	21:15:00		14.82	176.63
2713	2713	12/05/2022	21:18:54		NA	NA
2714	2714	12/05/2022	21:18:55		NA	NA
2715	2715	12/05/2022	21:18:56		NA	NA
2716	2716	12/05/2022	21:19:50		NA	NA

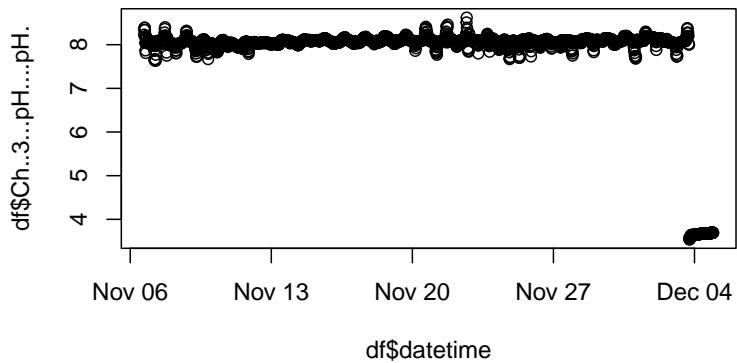
Ch..3....pH....pH.

2637	3.66
2638	3.66
2639	3.66
2640	3.66
2641	3.66

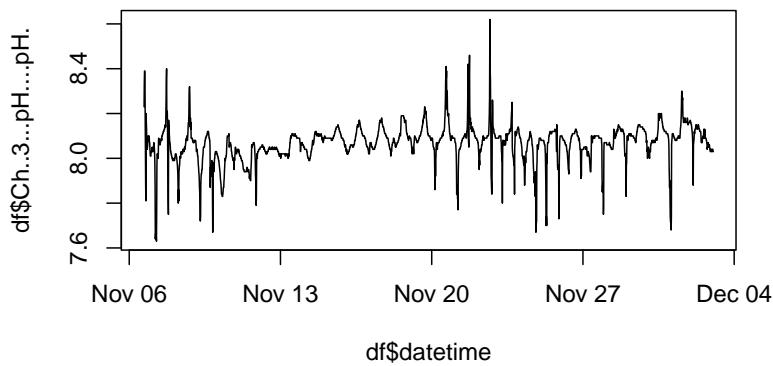
2642	3.66
2643	3.66
2644	3.66
2645	3.66
2646	3.66
2647	3.66
2648	3.66
2649	3.66
2650	3.67
2651	3.67
2652	3.67
2653	3.67
2654	3.67
2655	3.67
2656	3.67
2657	3.67
2658	3.67
2659	3.67
2660	3.67
2661	3.67
2662	3.67
2663	3.67
2664	3.67
2665	3.67
2666	3.67
2667	3.67
2668	3.67
2669	3.67
2670	3.67
2671	3.67
2672	3.67
2673	3.67
2674	3.67
2675	3.67
2676	3.67
2677	3.67
2678	3.67
2679	3.67
2680	3.67
2681	3.67
2682	3.67

```
2683      3.67
2684      3.68
2685      3.68
2686      3.68
2687      3.68
2688      3.68
2689      3.68
2690      3.68
2691      3.68
2692      3.68
2693      3.68
2694      3.67
2695      3.67
2696      3.67
2697      3.67
2698      3.68
2699      3.68
2700      3.68
2701      3.69
2702      3.69
2703      3.70
2704      3.70
2705      3.70
2706      3.70
2707      3.70
2708      3.70
2709      3.70
2710      3.70
2711      3.69
2712      3.69
2713        NA
2714        NA
2715        NA
2716        NA
```

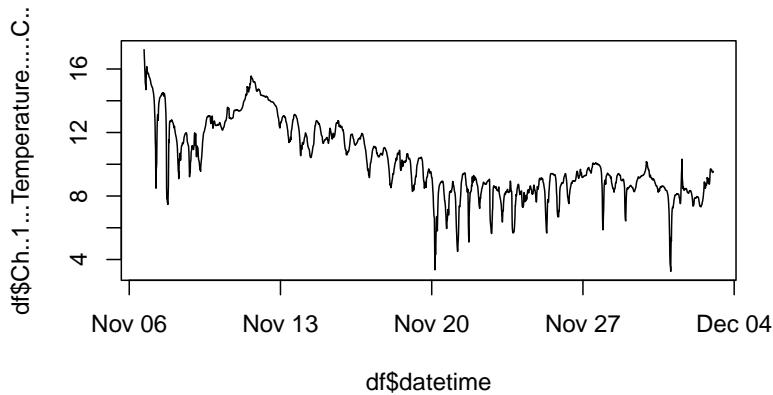
```
df$datetime <- as.POSIXct(strptime(df$date.Time..EST., format = "%m/%d/%Y %H:%M:%S")) #Switch to EST
plot(df$datetime, df$Ch..3...pH....pH.)
```



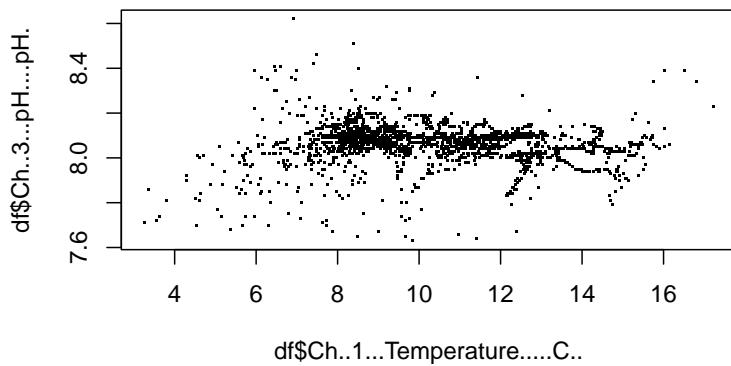
```
df <- df[df$datetime>"2022-11-07 13:00:00 EDT",]#"2022-11-07 13:00:00 EDT"
df <- df[df$datetime<"2022-12-04 00:00:00 EDT",]
plot(df$datetime, df$Ch...3...pH....pH., type = "l")
```



```
plot(df$datetime, df$Ch...1....Temperature.....C..., type = "l")
```



```
plot(df$Ch..1...Temperature.....C., df$Ch..3...pH....pH., pch = '.')
```



```
#Metadata
SN <- 076
tris <- 8.3
tris_temp <- 22.37

data_combined_new <- data.frame(
  Site = rep("Eel",times = nrow(df)),
  Date = df$datetime,
```

```

Temp = df$Ch..1...Temperature.....C.,
pH = df$Ch..3...pH....pH.,
Tris = rep(tris,length(df$datetime)),
Tris_temp = rep(tris_temp,length(df$datetime)),
Serial_number = rep(SN,length(df$datetime))
)

data_combined <- rbind(data_combined,data_combined_new)

setwd(wdEel)
# May
#(open_file <- filenames[3])
open_file <- "21333340_Eel_Pond 2023-05-08 09_10_54 EDT (Data EDT).csv"

df <- read.csv(open_file, stringsAsFactors = F)

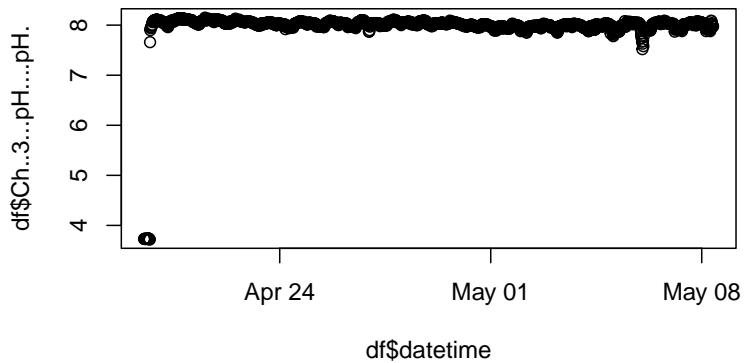
df <- df[,c(1:5)]
head(df)

  X.      Date.Time..EDT. Ch..1...Temperature.....C. Ch..2...Millivolts....mv.
1  1 04/19/2023 12:00:00          19.27           161.54
2  2 04/19/2023 12:15:00          19.25           161.52
3  3 04/19/2023 12:30:00          19.37           161.52
4  4 04/19/2023 12:45:00          19.39           161.55
5  5 04/19/2023 13:00:00          19.44           161.57
6  6 04/19/2023 13:15:00          19.46           161.57
  Ch..3...pH....pH.
1            3.73
2            3.73
3            3.73
4            3.73
5            3.73
6            3.73

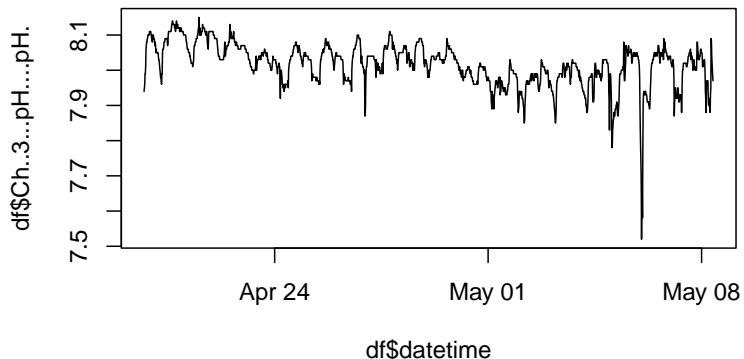
names(df) <- c("n","Date.Time..EDT.", "Ch..1...Temperature.....C.", "mV",
              "Ch..3...pH....pH. ")

df$datetime <- as.POSIXct(strptime(df>Date.Time..EDT., format = "%m/%d/%Y %H:%M:%S")) #Change to
plot(df$datetime, df$Ch..3...pH....pH.)

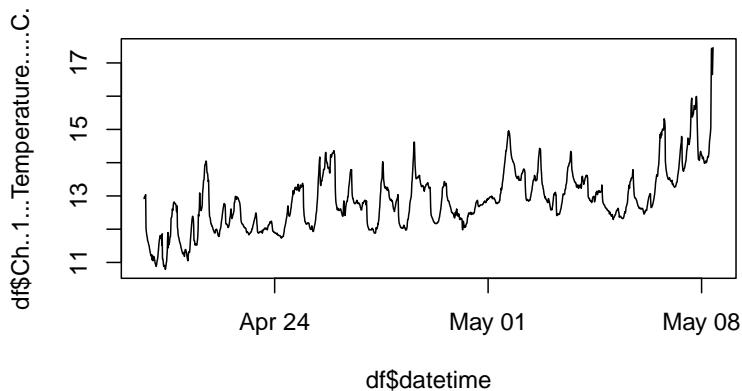
```



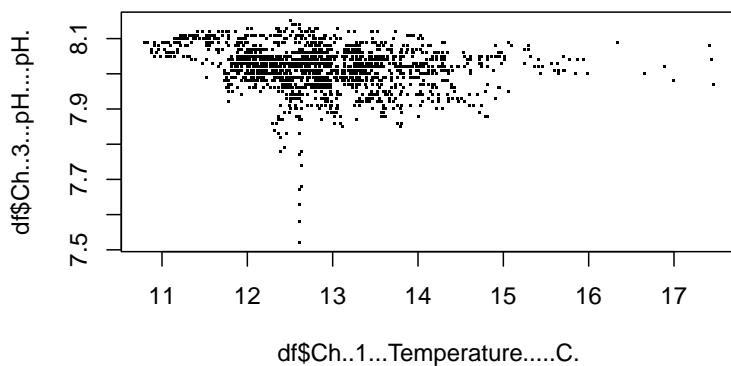
```
df <- df[df$datetime>"2023-04-19 17:00:00 EDT",]  
#df <- df[df$datetime<"2022-12-05 00:00:00 EDT",]  
plot(df$datetime, df$Ch...3...pH...pH., type = "l")
```



```
plot(df$datetime, df$Ch...1...Temperature.....C., type = "l")
```



```
plot(df$Ch..1...Temperature.....C., df$Ch..3...pH....pH., pch = '.')
```



```
#Metadata
SN <- 340
tris <- 8.37
tris_temp <- 18.00

data_combined_new <- data.frame(
  Site = rep("Eel",times = nrow(df)),
  Date = df$datetime,
```

```

Temp = df$Ch..1...Temperature.....C.,
pH = df$Ch..3...pH....pH.,
Tris = rep(tris,length(df$datetime)),
Tris_temp = rep(tris_temp,length(df$datetime)),
Serial_number = rep(SN,length(df$datetime))
)

data_combined <- rbind(data_combined,data_combined_new)

setwd(wdEel)
# June
#(open_file <- filenames[3])
open_file <- "Eel_076 2023-06-11 14_39_04 EDT (Data EDT).csv"

df <- read.csv(open_file, stringsAsFactors = F)

df <- df[,c(1:5)]
head(df)

X.      Date.Time..EDT. Ch.1...Temperature.....C. Ch.2...Millivolts....mv.
1  1 05/08/2023 06:52:33          NA             NA
2  2 05/08/2023 06:52:35          NA             NA
3  3 05/08/2023 06:55:09          NA             NA
4  4 05/08/2023 06:55:10          NA             NA
5  5 05/08/2023 07:00:00         16.11          -101.69
6  6 05/08/2023 07:15:00         16.70           -89.83

Ch.3...pH....pH.
1          NA
2          NA
3          NA
4          NA
5          8.31
6          8.09

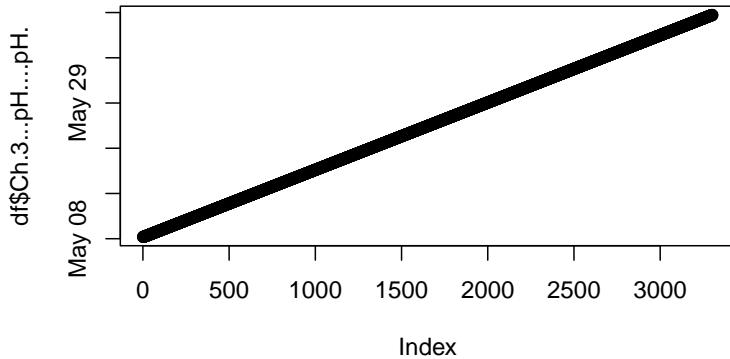
names(df) <- c("n","Date.Time..EDT.", "Ch.1...Temperature.....C.", "mV",
              "Ch.3...pH....pH" )

tail(df)

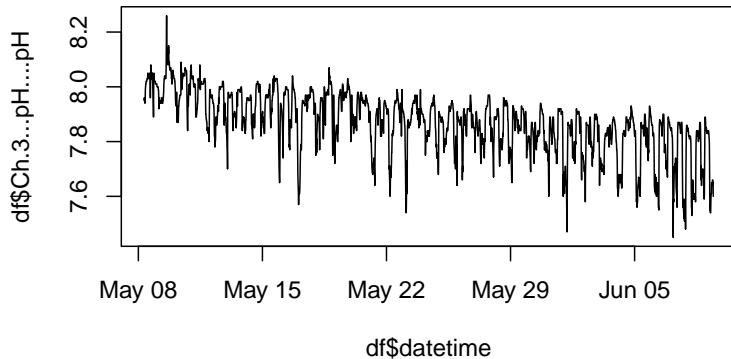
```

n	Date.Time..EDT.	Ch.1...Temperature.....C.	mV	Ch.3...pH....pH
3297	3297 06/11/2023 14:00:00		26.89	-13.50
3298	3298 06/11/2023 14:15:00		26.72	-13.78
3299	3299 06/11/2023 14:30:00		26.40	-13.89
3300	3300 06/11/2023 14:38:52		NA	NA
3301	3301 06/11/2023 14:38:53		NA	NA
3302	3302 06/11/2023 14:38:54		NA	NA

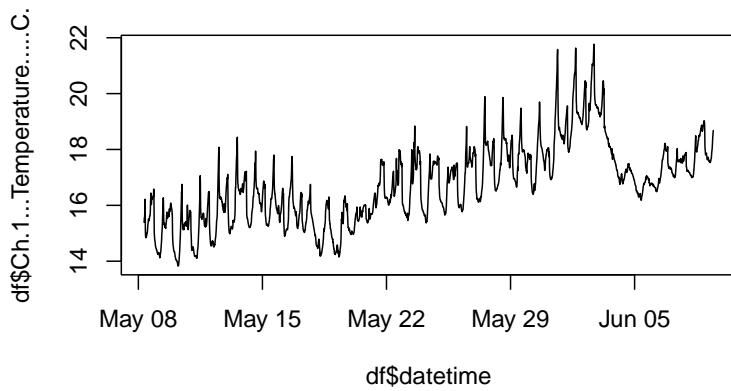
```
df$datetime <- as.POSIXct(strptime(df$date..EDT., format = "%m/%d/%Y %H:%M:%S")) #EDT
plot(df$datetime, df$Ch.3...pH....pH.)
```



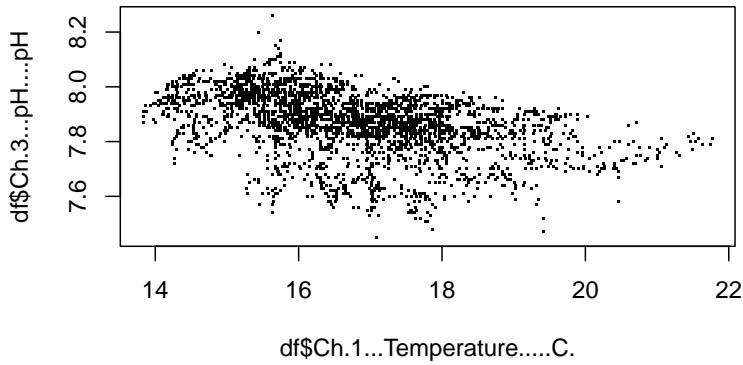
```
df <- df[df$datetime > "2023-05-08 07:30:00 EDT",]
df <- df[df$datetime < "2023-06-09 10:45:00 EDT",]
plot(df$datetime, df$Ch.3...pH....pH, type = "l")
```



```
plot(df$datetime, df$Ch.1...Temperature.....C., type = "l")
```



```
plot(df$Ch.1...Temperature.....C., df$Ch.3...pH....pH, pch = '.')
```



```
#Metadata
SN <- 076
tris <- 8.33
tris_temp <- 17.37

data_combined_new <- data.frame(
  Site = rep("Eel", times = length(df$datetime)),
  Date = df$datetime,
  Temp = df$Ch.1...Temperature.....C.,
  pH = df$Ch.3...pH....pH,
  Tris = rep(tris, length(df$datetime)),
  Tris_temp = rep(tris_temp, times = length(df$datetime)),
  Serial_number = rep(SN, times = length(df$datetime))
)

data_combined <- rbind(data_combined,data_combined_new)
```

0.2 Combine

```
setwd("~/GitHub/EAD-ASEB-Ssolidissima-OA/projects/Seawater data/data/HOB0pHdata")

write.csv(data_combined, file = "pH data combined.csv")
```

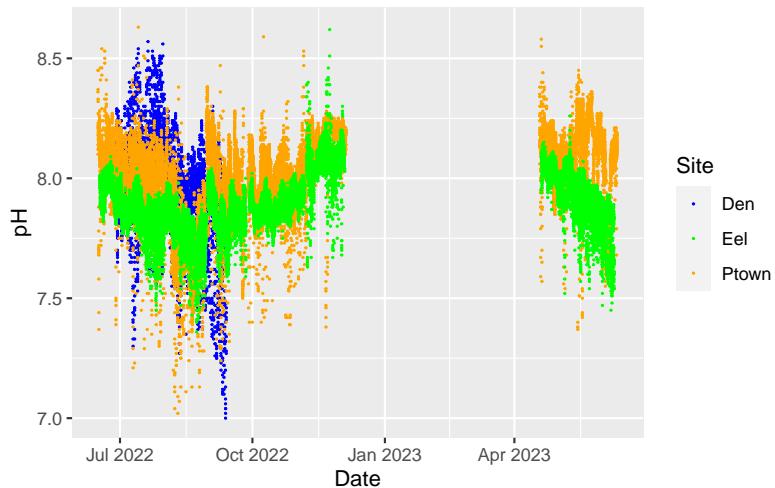
```

# rm(list = ls())
# data_combined <- read.csv("pH data combined.csv", stringsAsFactors = FALSE)

g1 <- ggplot(data_combined, aes(x=Date, y=pH, colour = Site))+ 
  geom_point(cex = .1) + 
  scale_color_manual(values=c("blue", "green", "orange"))
g1

```

Warning: Removed 9 rows containing missing values (`geom_point()`).



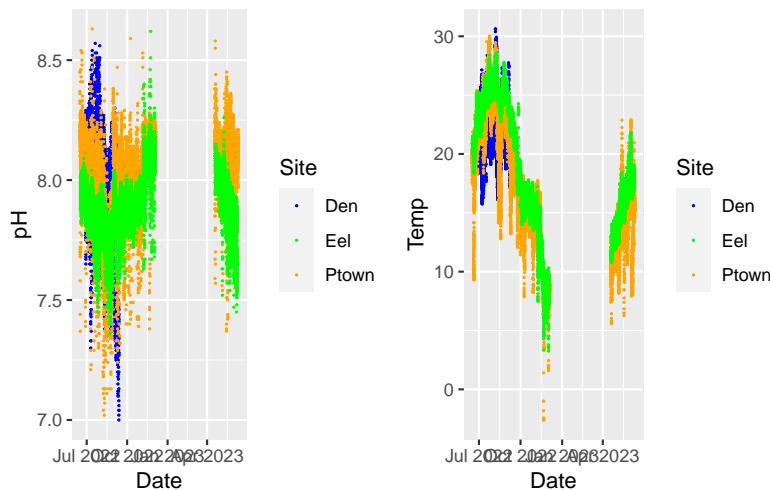
```

g2 <- ggplot(data_combined, aes(x=Date, y=Temp, colour = Site))+ 
  geom_point(cex = .1) + 
  scale_color_manual(values=c("blue", "green", "orange"))

ggarrange(g1, g2)

```

Warning: Removed 9 rows containing missing values (`geom_point()`).
 Removed 9 rows containing missing values (`geom_point()`).



```
min(data_combined$Date)
```

```
[1] "2022-06-15 18:09:46 EDT"
```

0.3 Calculate tris buffer offset and SW pH from NBS scale

```
tris_at_temp <- function(Tris_temp) {
  y=(11911.08-18.2499*35-0.039336*35*35)/(Tris_temp+273.15)+(-366.27059+0.53993607*35+0.000163
  return(y)
}

test_dat <- data_combined[1:5,]
data_combined$offset <- data_combined$Tris-tris_at_temp(data_combined$Tris_temp)

plot(data_combined$offset, na.rm = TRUE)
```

```
Warning in plot.window(...): "na.rm" is not a graphical parameter
```

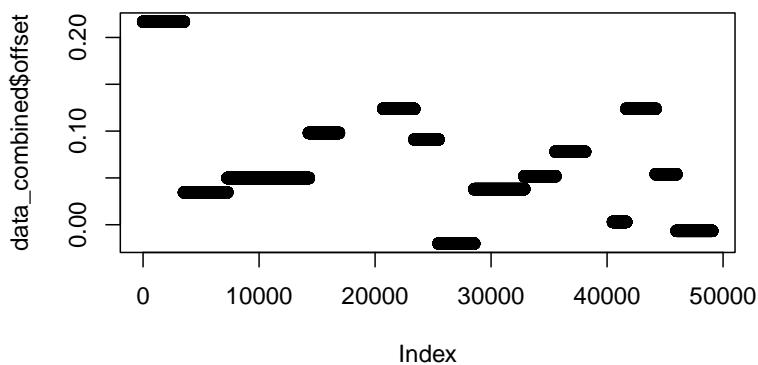
```
Warning in plot.xy(xy, type, ...): "na.rm" is not a graphical parameter
```

```
Warning in axis(side = side, at = at, labels = labels, ...): "na.rm" is not a graphical parameter
```

```
Warning in axis(side = side, at = at, labels = labels, ...): "na.rm" is not a graphical parameter
```

```
Warning in box(...): "na.rm" is not a graphical parameter
```

```
Warning in title(...): "na.rm" is not a graphical parameter
```



```
mean_offset <- mean(data_combined$offset, na.rm = TRUE)

data_combined[is.na(data_combined$offset), "offset"] <- mean_offset

data_combined$pH_corr <- data_combined$pH - data_combined$offset

if(params$Total_scale_corr==TRUE){
  data_combined$pH <- data_combined$pH_corr}
```

0.3.1 Plot tides

```

# Plot tides
setwd("~/GitHub/EAD-ASEB-Ssolidissima-0A/projects/Seawater data/data/Tides")
dir()

[1] "Den_Aug.txt"      "Den_Jul.txt"       "Den_Sep.txt"
[4] "Hyannis_Apr23.txt" "Hyannis_Aug.txt"   "Hyannis_Dec.txt"
[7] "Hyannis_Jul.txt"    "Hyannis_Jun23.txt" "Hyannis_May23.txt"
[10] "Hyannis_Nov.txt"   "Hyannis_Oct.txt"   "Hyannis_Sep.txt"
[13] "Ptown_Apr23.txt"   "Ptown_Aug.txt"    "Ptown_Dec.txt"
[16] "Ptown_Jul.txt"     "Ptown_Jun23.txt" "Ptown_May23.txt"
[19] "Ptown_Nov.txt"     "Ptown_Oct.txt"    "Ptown_Sep.txt"

tides.Ptown1 <- read.table("Ptown_Jul.txt", skip =14)
tides.Ptown2 <- read.table("Ptown_Aug.txt", skip =14)
tides.Ptown3 <- read.table("Ptown_Sep.txt", skip =14)
tides.Ptown4 <- read.table("Ptown_Oct.txt", skip =14)
tides.Ptown5 <- read.table("Ptown_Nov.txt", skip =14)
tides.Ptown6 <- read.table("Ptown_Dec.txt", skip =14)
tides.Ptown7 <- read.table("Ptown_Apr23.txt", skip =14)
tides.Ptown8 <- read.table("Ptown_May23.txt", skip =14)
tides.Ptown9 <- read.table("Ptown_Jun23.txt", skip =14)
tides.Den1 <- read.table("Den_Jul.txt", skip =14)
tides.Den2 <- read.table("Den_Aug.txt", skip =14)
tides.Den3 <- read.table("Den_Sep.txt", skip =14)
tides.Eel1 <- read.table("Hyannis_Jul.txt", skip =14)
tides.Eel2 <- read.table("Hyannis_Aug.txt", skip =14)
tides.Eel3 <- read.table("Hyannis_Sep.txt", skip =14)
tides.Eel4 <- read.table("Hyannis_Oct.txt", skip =14)
tides.Eel5 <- read.table("Hyannis_Nov.txt", skip =14)
tides.Eel6 <- read.table("Hyannis_Dec.txt", skip =14)
tides.Eel7 <- read.table("Hyannis_Apr23.txt", skip =14)
tides.Eel8 <- read.table("Hyannis_May23.txt", skip =14)
tides.Eel9 <- read.table("Hyannis_Jun23.txt", skip =14)

tides.Ptown <- rbind(tides.Ptown1, tides.Ptown2, tides.Ptown3,
                     tides.Ptown4, tides.Ptown5, tides.Ptown6,
                     tides.Ptown7, tides.Ptown8, tides.Ptown9)
tides.Den <- rbind(tides.Den1, tides.Den2, tides.Den3)
tides.Eel <- rbind(tides.Eel1, tides.Eel2, tides.Eel3,

```

```

        tides.Eel4, tides.Eel5, tides.Eel6,
        tides.Eel7, tides.Eel8, tides.Eel9)

tides <- data.frame(tides = rbind(tides.Ptown,tides.Den,tides.Eel),
                     Site = as.factor(c(rep("Ptown",times=nrow(tides.Ptown)),
                                         rep("Den",times=nrow(tides.Den)),
                                         rep("Eel",times=nrow(tides.Eel)))))

names(tides)<- c("date","day","Time","Tideheight","Site")

Date1 <- paste(tides$date,tides$Time,sep=" ")

tides>Date <- as.POSIXct(strptime(Date1,format = "%Y/%m/%d %H:%M"),tz="Etc/GMT+5")

tides$Date[tides$Site=="Eel"] <- tides$Date[tides$Site=="Eel"]+ minutes(60) #Eel Pond tide is off by 1 hour

tides$state <- rep(NA, length.out = nrow(tides))
# for(i in 2:nrow(tides)){
#   if(tides$Tideheight[i]>tides$Tideheight[i-1]){ tides$state[i] <- "Ebb"} else {tides$state[i] <- "Flood"}
# }

# tides$state <- ifelse(tides$Tideheight > dplyr::lag(tides$Tideheight, default = first(tides$Tideheight)), "Flood", "Ebb")

library(dplyr)
#https://www.statology.org/r-lag/

# Now 'tide_group' column will contain a unique identifier for each pair of ebb and flood tide

tides$previous_Tideheight <- lag(tides$Tideheight, n = 3)
tides <- tides %>%
  mutate(state = case_when(Tideheight > previous_Tideheight
                           | Tideheight > lag(previous_Tideheight,
                                               n = 3) ~ "Flood",
                           Tideheight < previous_Tideheight
                           | Tideheight < lag(previous_Tideheight,
                                               n = 3) ~ "Ebb",
                           TRUE ~ "Slack"),
        tide_group = cumsum(state != lag(state, default = first(state)))))

tides$state[1:3] <- "Flood"

```

```

data_combined$Date<-as.POSIXct(data_combined$Date, tz="Etc/GMT+5")
data_combined$Date<-round_date(data_combined$Date, unit="minutes")

data_combined$Site<-as.factor(data_combined$Site)
data_combined$H_plus_conc<-10^(-data_combined$pH)

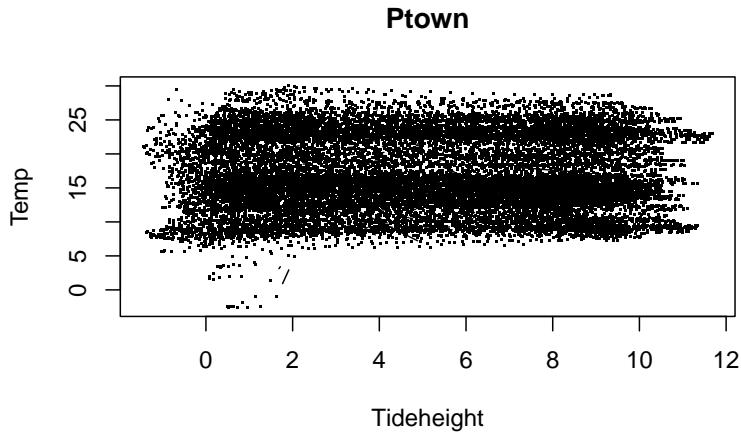
joined_df <- left_join(x=data_combined, y=tides, by= c("Date","Site"))

```

```

# Plot tides -----
Ptown <- joined_df[joined_df$Site=="Ptown",]
plot(Temp~Tideheight, pch =".",type ="b", data = Ptown, main="Ptown")

```

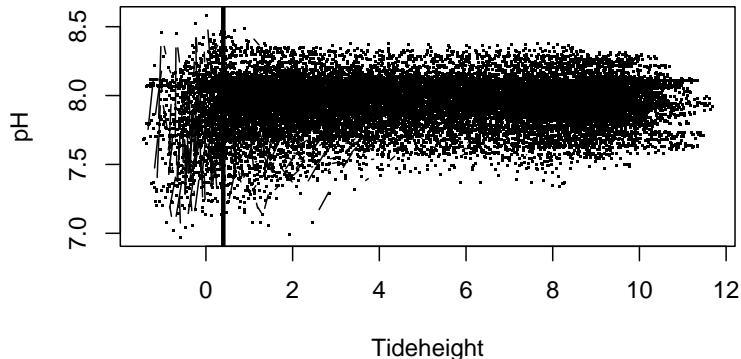


```

plot(pH~Tideheight,pch=".",data = Ptown, type = "b", main="Ptown")
lines(x=c(.4,.4),y=c(5,9),lwd=3)

```

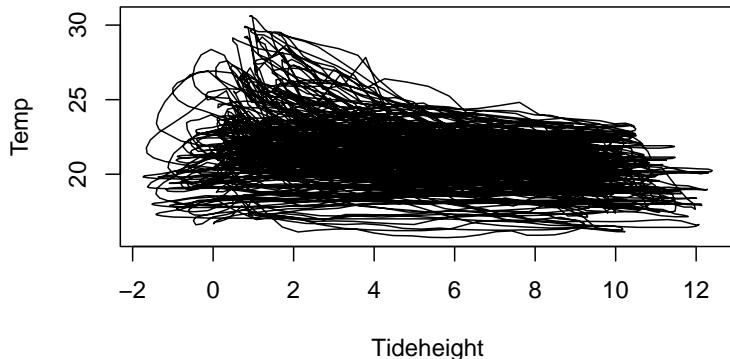
Ptown



```
# This graph seems to demonstrate that pH goes up or down at low tide (I'm guessing night or day)
# but below a certain point goes to ~7.8 - I think this is when the pH probe is not
# in water anymore.
```

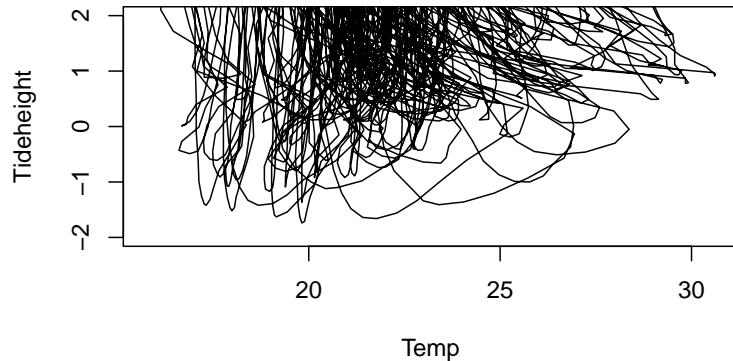
```
Den <- joined_df[joined_df$Site=="Den",]
plot(Temp~Tideheight, pch =".",type ="l", data = Den, main="Den")
```

Den



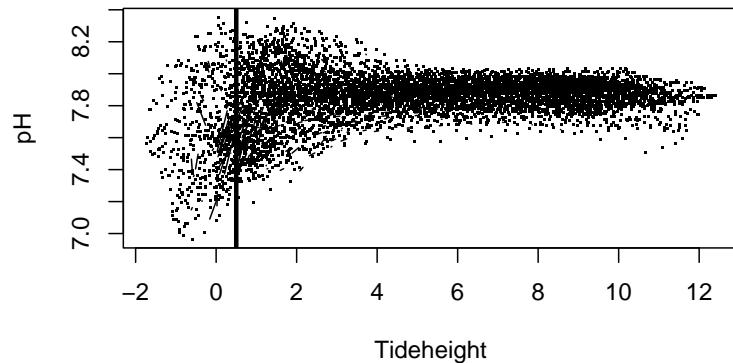
```
plot(Tideheight~Temp, pch =".",type ="l", data = Den, ylim = c(-2,2),main="Den")
```

Den



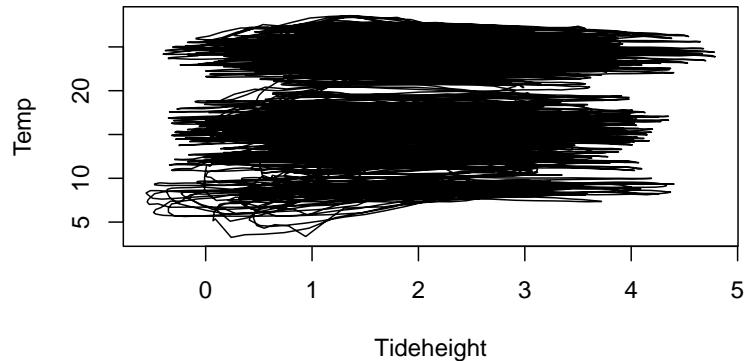
```
# What concerns me are the lines where temp rapidly increases at low tide. I expect some slow t  
# I also wonder about introducing a lag of a few hours after the sensor has been out of the wa  
  
plot(pH~Tideheight,pch=".",data = Den, type = "b", main="Den")  
lines(x=c(.5,.5),y=c(5,9),lwd=3)
```

Den



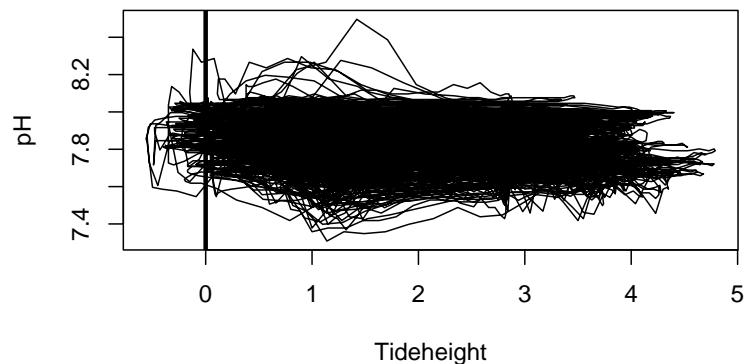
```
Eel <- joined_df[joined_df$Site=="Eel",]  
plot(Temp~Tideheight, pch =".",type ="l", data = Eel, main="Eel")
```

Eel

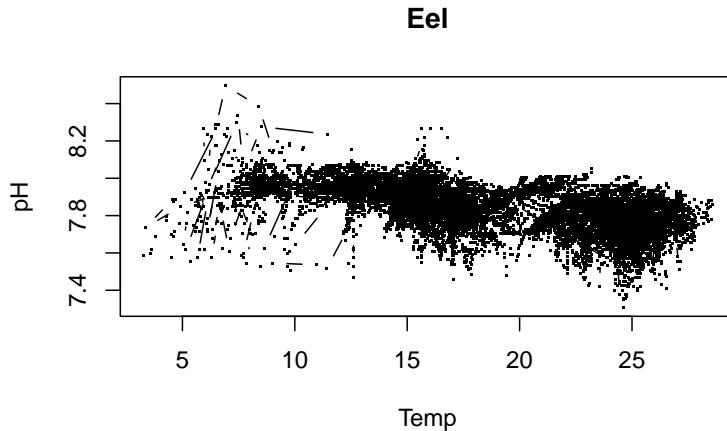


```
plot(pH~Tideheight,pch=".",data = Eel, type = "l", main="Eel")
lines(x=c(0,0),y=c(5,30),lwd=3)
```

Eel



```
plot(pH~Temp, pch =".",type ="b", data = Eel, main="Eel")
```



```
# Not clear from data what tide is too low. It might be more wind-dependent.
# At 0.5ft tide the cages are on sediment that is about 2.5 ft deep.
# The sensor is installed <0.5ft above the sediment. However there is likely a mismatch between
# The actual water level - the site water level does not vary much.
# Honestly, I don't see anything in this data linked with the tides to suggest that I'm getting
# If anything, the highest temps occur when the low tides are really minimal.
# The HOBO temp logger will probably help the most in figuring this out,
# it will help determine if there is some wind/tide-driven bad data or not due to sensor exposure.
```

Exclude pH from times water below low tide mark.

```
Ptown <- Ptown[Ptown$Tideheight>2,] #This was at 0.4
Den <- Den[Den$Tideheight>3,] #This was at .5
Eel <- Eel[Eel$Tideheight>2,] #This was at 0.
```

```
# Aggregate data -----
Ptown.hr <- aggregate(Ptown[,c("pH", "Temp", "H_plus_conc")],
                      list(Date = cut(Ptown[,c("Date")], breaks="hour")),
                      mean, na.rm = TRUE)
Den.hr <- aggregate(Den[,c("pH", "Temp", "H_plus_conc")],
                      list(Date = cut(Den[,c("Date")], breaks="hour")),
                      mean, na.rm = TRUE)
Eel.hr <- aggregate(Eel[,c("pH", "Temp", "H_plus_conc")],
                      list(Date = cut(Eel[,c("Date")], breaks="hour"))),
```

```

mean, na.rm = TRUE)

all.data.hr <- data.frame(
  rbind(Ptown.hr,Den.hr,Eel.hr),
  Site = as.factor(c(rep("Ptown",times = nrow(Ptown.hr)),
    rep("Den",times = nrow(Den.hr)),
    rep("Eel",times = nrow(Eel.hr)))),
  )
all.data.hr$date<- as.POSIXct(all.data.hr$date)

# Calc true mean not arithmetic mean
all.data.hr$pH <- -log10(all.data.hr$H_plus_conc)

all.data.hr.break <- all.data.hr[1:3,]
all.data.hr.break$date <- "2023-01-01 00:00:00"
all.data.hr.break$pH <- NA
all.data.hr.break$temp <- NA
all.data.hr.break$Site <- c("Ptown", "Eel", "Den")
all.data.hr <- rbind(all.data.hr,all.data.hr.break)

head(all.data.hr)

```

	Date	pH	Temp	H_plus_conc	Site
1	2022-07-01	8.012621	21.5750	9.713577e-09	Ptown
2	2022-07-01	8.010142	21.5825	9.769170e-09	Ptown
3	2022-07-01	8.015114	21.6875	9.657984e-09	Ptown
4	2022-07-01	8.027563	21.9675	9.385050e-09	Ptown
5	2022-07-01	7.994369	21.7700	1.013051e-08	Ptown
6	2022-07-01	7.977621	21.6700	1.052881e-08	Ptown

```

Ptown.dy <- aggregate(Ptown[,c("pH","Temp","H_plus_conc")],
  list(Date = cut(Ptown[,c("Date")], breaks="day")),
  mean, na.rm = TRUE)
Den.dy <- aggregate(Den[,c("pH","Temp","H_plus_conc")],
  list(Date = cut(Den[,c("Date")], breaks="day")),
  mean, na.rm = TRUE)
Eel.dy <- aggregate(Eel[,c("pH","Temp","H_plus_conc")],
  list(Date = cut(Eel[,c("Date")], breaks="day")),
  mean, na.rm = TRUE)

```

```

all.data.dy <- data.frame(
  rbind(Ptown.dy,Den.dy,Eel.dy),
  Site = as.factor(c(rep("Ptown",times = nrow(Ptown.dy)),
    rep("Den",times = nrow(Den.dy)),
    rep("Eel",times = nrow(Eel.dy)))),
  )
all.data.dy$Date<- as.POSIXct(all.data.dy$Date)

# Calc true mean not arithmetic mean
all.data.dy$pH <- -log10(all.data.dy$H_plus_conc)

all.data.dy.break <- all.data.dy[1:3,]
all.data.dy.break$Date <- "2023-01-01 00:00:00"
all.data.dy.break$pH <- NA
all.data.dy.break$Temp <- NA
all.data.dy.break$Site <- c("Ptown", "Eel", "Den")
all.data.dy <- rbind(all.data.dy,all.data.dy.break)

head(all.data.dy)

```

	Date	pH	Temp	H_plus_conc	Site
1	2022-07-01	8.051273	23.00237	8.886418e-09	Ptown
2	2022-07-02	7.941035	22.71263	1.145422e-08	Ptown
3	2022-07-03	7.936546	22.49120	1.157321e-08	Ptown
4	2022-07-04	7.973772	22.86960	1.062253e-08	Ptown
5	2022-07-05	7.973823	22.56342	1.062129e-08	Ptown
6	2022-07-06	7.877860	21.95789	1.324769e-08	Ptown

```

Ptown.wk <- aggregate(Ptown[,c("pH","Temp","H_plus_conc")],
  list(Date = cut(Ptown[,c("Date")], breaks="week")),
  mean, na.rm = TRUE)
Den.wk <- aggregate(Den[,c("pH","Temp","H_plus_conc")],
  list(Date = cut(Den[,c("Date")], breaks="week")),
  mean, na.rm = TRUE)
Eel.wk <- aggregate(Eel[,c("pH","Temp","H_plus_conc")],
  list(Date = cut(Eel[,c("Date")], breaks="week")),
  mean, na.rm = TRUE)

all.data.wk <- data.frame(

```

```

  rbind(Ptown.wk,Den.wk,Eel.wk),
  Site = as.factor(c(rep("Ptown",times = nrow(Ptown.wk)),
    rep("Den",times = nrow(Den.wk)),
    rep("Eel",times = nrow(Eel.wk))))
  )
all.data.wk$Date<- as.POSIXct(all.data.wk$Date)

# Calc true mean not arithmetic mean
all.data.wk$pH <- -log10(all.data.wk$H_plus_conc)

head(all.data.wk)

```

	Date	pH	Temp	H_plus_conc	Site
1	2022-06-27	7.973310	22.73648	1.063383e-08	Ptown
2	2022-07-04	7.960365	22.64361	1.095557e-08	Ptown
3	2022-07-11	7.975953	23.07258	1.056931e-08	Ptown
4	2022-07-18	7.950433	26.22679	1.120900e-08	Ptown
5	2022-07-25	7.934652	24.50816	1.162379e-08	Ptown
6	2022-08-01	7.851386	25.47081	1.408037e-08	Ptown

```

all.data.wk.break <- all.data.wk[1:3,]
all.data.wk.break$Date <- "2023-01-01 00:00:00"
all.data.wk.break$pH <- NA
all.data.wk.break$Temp <- NA
all.data.wk.break$Site <- c("Ptown", "Eel", "Den")
all.data.wk <- rbind(all.data.wk,all.data.wk.break)
tail(all.data.wk)

```

	Date	pH	Temp	H_plus_conc	Site
72	2023-05-22	7.828177	16.79041	1.485329e-08	Eel
73	2023-05-29	7.791064	18.32975	1.617840e-08	Eel
74	2023-06-05	7.700131	17.34392	1.994659e-08	Eel
75	2023-01-01	NA	NA	1.063383e-08	Ptown
76	2023-01-01	NA	NA	1.095557e-08	Eel
77	2023-01-01	NA	NA	1.056931e-08	Den

0.3.2 Final plots

```
# plot(all.data$Date, all.data$pH,
#       color = as.factor(data_combined$Site), pch = ".")  
  
sampling.pH <- data.frame(  
  x = as.POSIXct(c("2022-06-29 00:00:00 EDT","2022-08-2 00:00:00 EDT","2022-09-9 00:00:00 EDT"  
  y = c(7.5,7.5,7.5,7.5,7.5,7.5)  
)  
sampling.temp <- data.frame(  
  x = as.POSIXct(c("2022-06-29 00:00:00 EDT","2022-08-2 00:00:00 EDT","2022-09-9 00:00:00 EDT"  
  y = c(5,5,5,5,5,5)  
)  
  
theme_set(theme_bw())  
  
colors <- c(  
  "Den"="blue",  
  "Eel"="green",  
  "Ptown"="orange",  
  "Date sampled"="black"  
)  
  
#scale_color_manual(values=c("blue", "green", "orange"))  
  
# Now I'm aggregating by day... and this is a bit easier to digest when comparing among sites  
head(all.data.wk)
```

	Date	pH	Temp	H_plus_conc	Site
1	2022-06-27	7.973310	22.73648	1.063383e-08	Ptown
2	2022-07-04	7.960365	22.64361	1.095557e-08	Ptown
3	2022-07-11	7.975953	23.07258	1.056931e-08	Ptown
4	2022-07-18	7.950433	26.22679	1.120900e-08	Ptown
5	2022-07-25	7.934652	24.50816	1.162379e-08	Ptown
6	2022-08-01	7.851386	25.47081	1.408037e-08	Ptown

```
str(all.data.wk)
```

```
'data.frame': 77 obs. of 5 variables:
```

```
$ Date      : POSIXct, format: "2022-06-27" "2022-07-04" ...
$ pH        : num  7.97 7.96 7.98 7.95 7.93 ...
$ Temp      : num  22.7 22.6 23.1 26.2 24.5 ...
$ H_plus_conc: num  1.06e-08 1.10e-08 1.06e-08 1.12e-08 1.16e-08 ...
$ Site      : Factor w/ 3 levels "Den","Eel","Ptown": 3 3 3 3 3 3 3 3 3 3 ...
```

```
gg1 <- ggplot(all.data.wk, aes(x=Date, y=pH, colour = Site))+  
  geom_point(cex = 1) +  
  geom_line(cex = 1)+  
  ylim(7.5,8.3)+  
  #xlim(as.POSIXct(c("2022-06-29 00:00:00 EDT","2022-09-9 00:00:00 EDT")))+  
  geom_point(data = sampling.pH, aes(x,y), color = "black", cex = 5)+  
  scale_color_manual(values=colors) +  
  guides(colour = guide_legend(override.aes = list(size=5)))+  
  ggtitle('B. pH')
```

Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
 i Please use `linewidth` instead.

```
ylab = expression("Temperature " ( degree*C))

gg2 <- ggplot(all.data.wk, aes(x=Date, y=Temp, colour = Site))+  
  geom_point(cex = 1) +  
  geom_line(cex = 1)+  
  labs(y=ylab)+  
  #ylim(14,30)+  
  #xlim(as.POSIXct(c("2022-06-29 00:00:00 EDT","2022-09-9 00:00:00 EDT")))+  
  geom_point(data = sampling.temp, aes(x,y), color = "black", cex = 5)+  
  scale_color_manual(values=colors) +  
  guides(colour = guide_legend(override.aes = list(size=5)))+  
  ggtitle('A. Temp')

ggarrange(gg2, gg1, common.legend = TRUE, legend="bottom")
```

Warning: Removed 3 rows containing missing values (`geom_point()`).

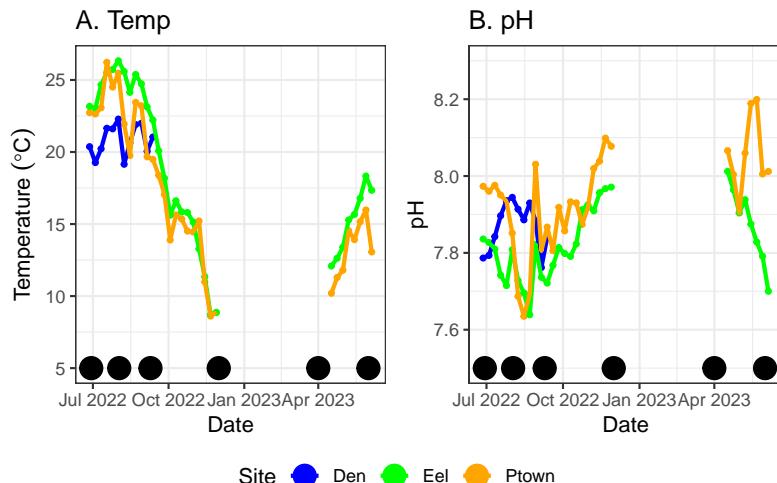
Warning: Removed 1 row containing missing values (`geom_line()`).

```
Warning: Removed 3 rows containing missing values (`geom_point()`).
```

```
Warning: Removed 1 row containing missing values (`geom_line()`).
```

```
Warning: Removed 3 rows containing missing values (`geom_point()`).
```

```
Warning: Removed 1 row containing missing values (`geom_line()`).
```



```
#
```

```
setwd("~/GitHub/EAD-ASEB-Ssolidissima-0A/projects/Seawater data/data/HOB0pHdata")
write.csv(all.data.wk, file="Summary_weekly_ave.csv")
write.csv(all.data.dy, file="Summary_daily_ave.csv")
```

0.3.3 Compare with surfclam carbonate chemistry data

```
dir_data <- "~/GitHub/EAD-ASEB-Ssolidissima-0A/projects/Seawater data/data/Carbonate chem/"
carb_dat <- read.csv(paste(dir_data,"Surfclam_carbonate_data_wgraphs.csv", sep = ""), stringsAsFactors=FALSE)
carb_dat$date <- as.POSIXct(carb_dat$date, format = "%m/%d/%y")
carb_dat <- carb_dat[!is.na(carb_dat$date),]
carb_dat$Site[carb_dat$Site == "Eel Pond"] <- "Eel"
carb_dat$Site[carb_dat$Site == "Provincetown"] <- "Ptown"
carb_dat$Site[carb_dat$Site == "East Dennis (aquaculture)"] <- "Den"
carb_dat$Site <- as.factor(carb_dat$Site)
```

```

sampling.pH <- data.frame(
  x = as.POSIXct(c("2022-06-29 00:00:00 EDT","2022-08-2 00:00:00 EDT","2022-09-9 00:00:00 EDT"),
  y = rep(7.2, 6)
)
sampling.temp <- data.frame(
  x = as.POSIXct(c("2022-06-29 00:00:00 EDT","2022-08-2 00:00:00 EDT","2022-09-9 00:00:00 EDT"),
  y = rep(0,6)
)

gg1 <- ggplot(all.data.dy, aes(x=Date, y=pH, colour = Site))+  

  geom_line(cex = 1)+  

  ylim(7.2,8.3)+  

  #xlim(as.POSIXct(c("2022-06-29 00:00:00 EDT","2022-09-9 00:00:00 EDT")))+  

  geom_point(data = sampling.pH, aes(x,y), color = "black", cex = 1)+  

  scale_color_manual(values=colors) +  

  guides(colour = guide_legend(override.aes = list(size=5)))+  

  ggtitle('B. pH')+  

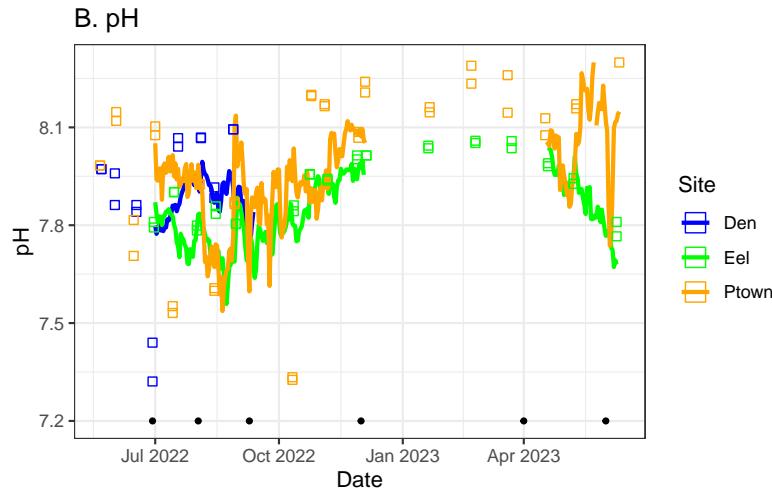
  geom_point(data = carb_dat[carb_dat$Stratum=="Bottom water",],aes(x=Date, y = InSitu.pH, colo

gg1

```

Warning: Removed 1 row containing missing values (`geom_line()`).

Warning: Removed 1 rows containing missing values (`geom_point()`).



```

ylab = expression("Temperature " ( degree*C))

gg2 <- ggplot(all.data.dy, aes(x=Date, y=Temp, colour = Site))+  

  geom_line(cex = 1)+  

  labs(y=ylab)+  

  ylim(0,30)+  

  #xlim(as.POSIXct(c("2022-06-29 00:00:00 EDT","2022-09-9 00:00:00 EDT")))+  

  geom_point(data = sampling.temp, aes(x,y), color = "black", cex = 1)+  

  scale_color_manual(values=colors) +  

  guides(colour = guide_legend(override.aes = list(size=5)))+  

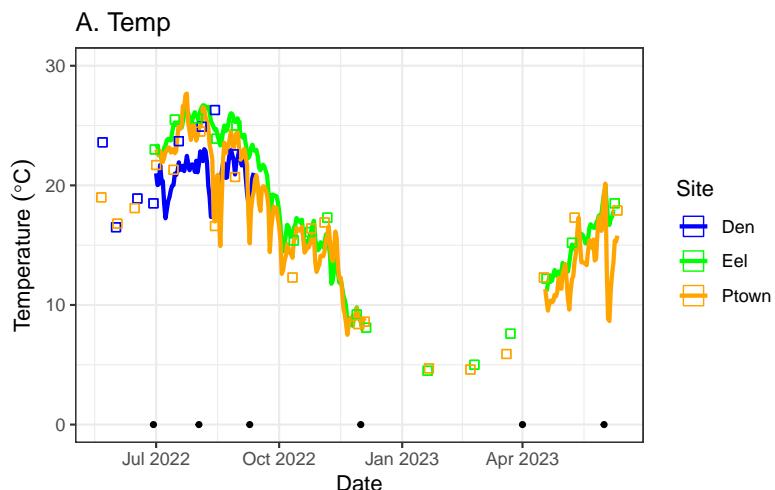
  ggtitle('A. Temp')+  

  geom_point(data = carb_dat[carb_dat$Stratum=="Bottom water",],aes(x=Date, y = Water.Sediment  

gg2

```

Warning: Removed 1 row containing missing values (`geom_line()`).

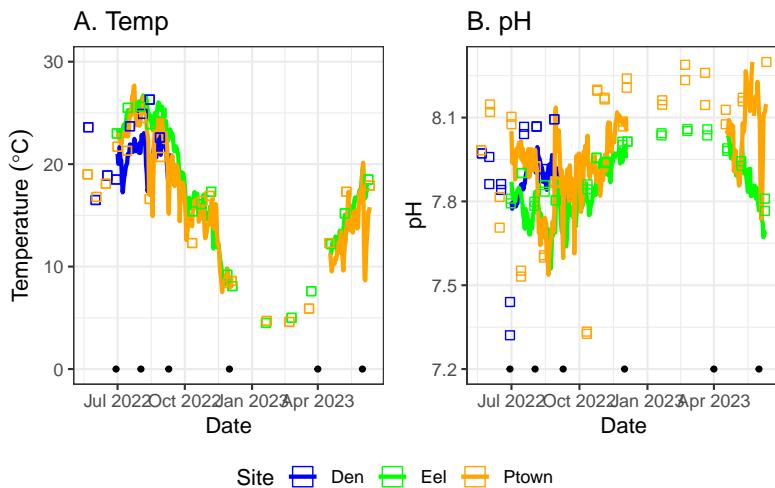


```
ggarrange(gg2, gg1, common.legend = TRUE, legend="bottom")
```

Warning: Removed 1 row containing missing values (`geom_line()`).

Warning: Removed 1 row containing missing values (`geom_line()`).
 Removed 1 row containing missing values (`geom_line()`).

```
Warning: Removed 1 rows containing missing values (`geom_point()`).
```



```
# I need to add all of the tide data so I can subset the data using tide data

# Tide data websites:
#https://tidesandcurrents.noaa.gov/map/index.html
# Dennis https://tidesandcurrents.noaa.gov/noaatidepredictions.html?id=8447241&units=standard&
# Falmouth - only L/H: https://tidesandcurrents.noaa.gov/noaatidepredictions.html?id=8447865
# Hyannis - https://tidesandcurrents.noaa.gov/noaatidepredictions.html?id=8447605
# Note I'm using the Hyannis dataset because they have down to 1min predictions. Falmouth does
# Compared to Eel Pond Entrance on Willy Weather, Hyannis is ~30 min behind - which is perfect
# Willy Weather says it is.
```

0.4 Read YSI data

```
data_dir <- "~/GitHub/EAD-ASEB-Ssolidissima-0A/projects/Seawater data/data/"
YSI <- read.csv(paste(data_dir,"YSI_Eel_Ptown.csv",sep = ""), stringsAsFactors = FALSE)

YSI$datetime <- as.POSIXct(YSI$datetime)
YSI$Hplus_P <- 10^-YSI$pH_P
YSI$Hplus_E <- 10^-YSI$pH_E
```

```

YSI.Ptown.dy <- aggregate(YSI[,c("Temp_P","pH_P","Sal_P","Hplus_P")],
                           by = list(Date = cut(YSI$datetime, breaks="day")),
                           FUN = mean, na.rm=TRUE)
YSI.Eel.dy <- aggregate(YSI[,c("Temp_E","pH_E", "Sal_E","Hplus_E")],
                           by = list(Date = cut(YSI$datetime, breaks="day")),
                           FUN = mean, na.rm=TRUE)
HOBOpH.Ptown.dy <- all.data.dy[all.data.dy$Site == "Ptown",]
HOBOpH.Eel.dy <- all.data.dy[all.data.dy$Site == "Eel",]
joined_Ptown <- merge(x=HOBOpH.Ptown.dy, y=YSI.Ptown.dy, by = "Date", all=T)
joined_Eel <- merge(x=HOBOpH.Eel.dy, y=YSI.Eel.dy, by = "Date", all=T)

mod_Ptown_Hplus <- lm(data = joined_Ptown,H_plus_conc ~ Hplus_P * Temp_P * Sal_P)
mod_Ptown_pH <- lm(data = joined_Ptown,pH ~ pH_P * Temp_P * Sal_P)
mod_Ptown_Temp <- lm(data = joined_Ptown,Temp ~ pH_P + Temp_P + Sal_P)
mod_Ptown_Temp <- lm(data = joined_Ptown,Temp ~ Temp_P)
summary(mod_Ptown_Hplus)

```

Call:

```
lm(formula = H_plus_conc ~ Hplus_P * Temp_P * Sal_P, data = joined_Ptown)
```

Residuals:

Min	1Q	Median	3Q	Max
-8.905e-09	-2.191e-09	-5.589e-10	1.478e-09	1.279e-08

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	5.346e-07	4.309e-07	1.240	0.217
Hplus_P	-7.609e+01	6.031e+01	-1.262	0.209
Temp_P	-4.099e-08	2.735e-08	-1.499	0.136
Sal_P	-1.772e-08	1.367e-08	-1.296	0.197
Hplus_P:Temp_P	5.043e+00	3.894e+00	1.295	0.197
Hplus_P:Sal_P	2.571e+00	1.907e+00	1.349	0.179
Temp_P:Sal_P	1.368e-09	8.657e-10	1.580	0.116
Hplus_P:Temp_P:Sal_P	-1.694e-01	1.230e-01	-1.377	0.170

Residual standard error: 3.582e-09 on 158 degrees of freedom

(417 observations deleted due to missingness)

Multiple R-squared: 0.4328, Adjusted R-squared: 0.4076

F-statistic: 17.22 on 7 and 158 DF, p-value: < 2.2e-16

```
summary(mod_Ptown_pH)
```

Call:

```
lm(formula = pH ~ pH_P * Temp_P * Sal_P, data = joined_Ptown)
```

Residuals:

Min	1Q	Median	3Q	Max
-0.27818	-0.06027	0.01158	0.07656	0.30333

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	295.61470	216.39227	1.366	0.174
pH_P	-35.22421	26.54541	-1.327	0.186
Temp_P	-19.86457	13.58235	-1.463	0.146
Sal_P	-9.74548	6.88751	-1.415	0.159
pH_P:Temp_P	2.45630	1.66394	1.476	0.142
pH_P:Sal_P	1.19350	0.84509	1.412	0.160
Temp_P:Sal_P	0.66495	0.43124	1.542	0.125
pH_P:Temp_P:Sal_P	-0.08219	0.05284	-1.556	0.122

Residual standard error: 0.1079 on 158 degrees of freedom
(417 observations deleted due to missingness)

Multiple R-squared: 0.5366, Adjusted R-squared: 0.5161
F-statistic: 26.14 on 7 and 158 DF, p-value: < 2.2e-16

```
summary(mod_Ptown_Temp)
```

Call:

```
lm(formula = Temp ~ Temp_P, data = joined_Ptown)
```

Residuals:

Min	1Q	Median	3Q	Max
-5.3191	-0.4547	0.0158	0.5600	2.6649

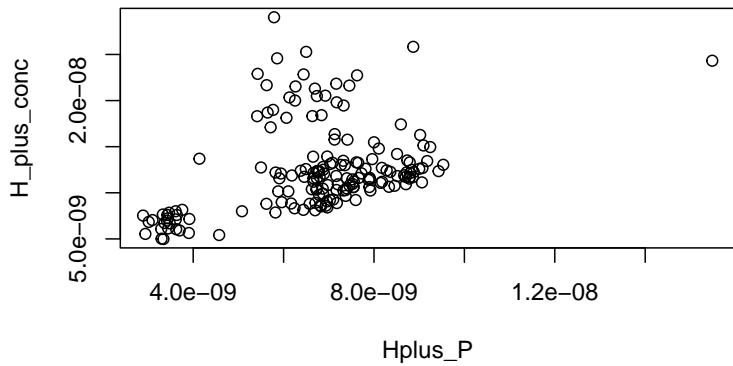
Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-0.20433	0.35192	-0.581	0.562

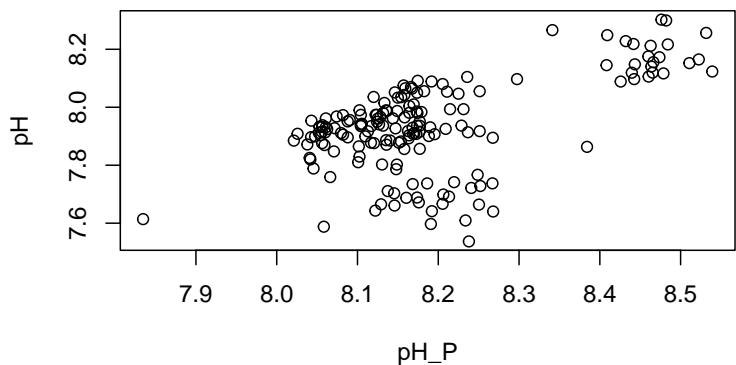
```
Temp_P      1.02776    0.01907  53.894   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.156 on 164 degrees of freedom
(417 observations deleted due to missingness)
Multiple R-squared:  0.9466,    Adjusted R-squared:  0.9462
F-statistic:  2905 on 1 and 164 DF,  p-value: < 2.2e-16
```

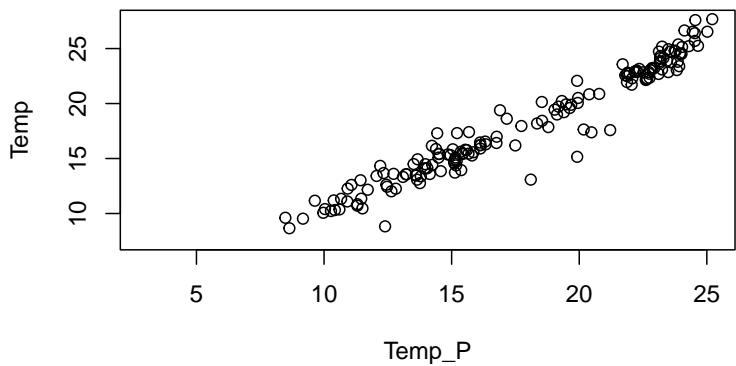
```
plot(data = joined_Ptown, H_plus_conc ~ Hplus_P)
```



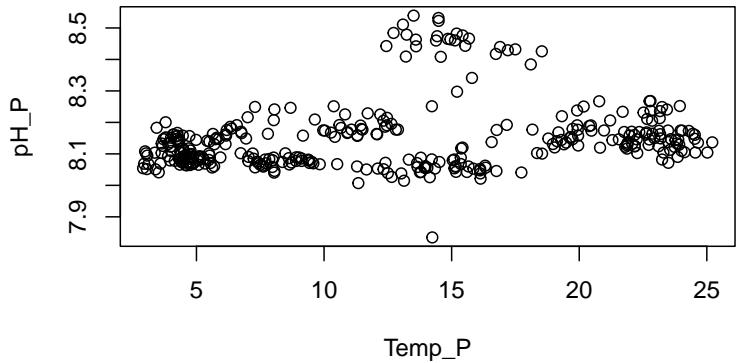
```
plot(data = joined_Ptown, pH ~ pH_P)
```



```
plot(data = joined_Ptown, Temp ~ Temp_P)
```



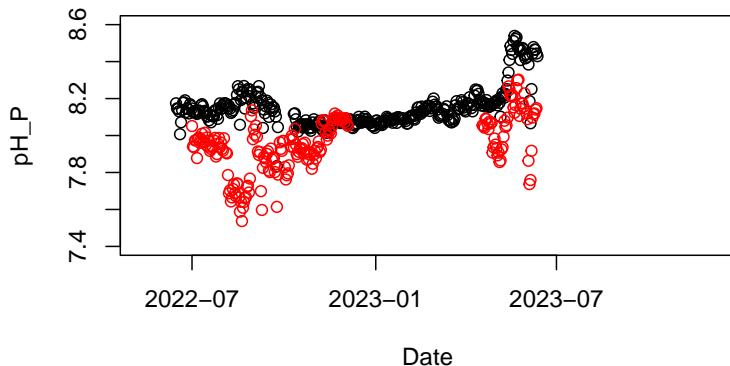
```
plot(data = joined_Ptown, pH_P ~ Temp_P)
```



```
str(joined_Ptown)
```

```
'data.frame': 583 obs. of 9 variables:
 $ Date      : POSIXct, format: "2022-05-13" "2022-05-14" ...
 $ pH        : num  NA NA NA NA NA NA NA NA NA ...
 $ Temp       : num  NA NA NA NA NA NA NA NA NA ...
 $ H_plus_conc: num  NA NA NA NA NA NA NA NA NA ...
 $ Site       : Factor w/ 3 levels "Den","Eel","Ptown": NA NA NA NA NA NA NA NA NA ...
 $ Temp_P     : num  NaN NaN NaN NaN NaN NaN NaN NaN ...
 $ pH_P       : num  NaN NaN NaN NaN NaN NaN NaN NaN ...
 $ Sal_P      : num  NaN NaN NaN NaN NaN NaN NaN NaN ...
 $ Hplus_P    : num  NaN NaN NaN NaN NaN NaN NaN NaN ...
```

```
par(mfrow = c(1, 1))
plot(data = joined_Ptown, pH_P ~ Date, ylim = c(7.4,8.6))
points(data = joined_Ptown, pH ~ Date, col = "red")
```



```

mod_Eel_Hplus <- lm(data = joined_Eel,H_plus_conc ~ Hplus_E * Temp_E * Sal_E)
mod_Eel_pH   <- lm(data = joined_Eel,pH ~ pH_E * Temp_E * Sal_E)
mod_Eel_Temp <- lm(data = joined_Eel,Temp ~ pH_E + Temp_E + Sal_E)
mod_Eel_Temp <- lm(data = joined_Eel,Temp ~ Temp_E)
summary(mod_Eel_Hplus) #R2 .67

```

Call:

```
lm(formula = H_plus_conc ~ Hplus_E * Temp_E * Sal_E, data = joined_Eel)
```

Residuals:

Min	1Q	Median	3Q	Max
-4.496e-09	-1.899e-09	-1.030e-11	1.602e-09	8.704e-09

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	2.018e-07	1.055e-07	1.912	0.0575 .
Hplus_E	-1.117e+01	6.078e+00	-1.837	0.0680 .
Temp_E	-9.338e-09	4.269e-09	-2.187	0.0301 *
Sal_E	-6.372e-09	3.450e-09	-1.847	0.0665 .
Hplus_E:Temp_E	4.762e-01	2.343e-01	2.032	0.0437 *
Hplus_E:Sal_E	3.658e-01	1.999e-01	1.830	0.0690 .
Temp_E:Sal_E	3.185e-10	1.395e-10	2.282	0.0237 *
Hplus_E:Temp_E:Sal_E	-1.554e-02	7.739e-03	-2.008	0.0463 *

```
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 2.556e-09 on 168 degrees of freedom  
(406 observations deleted due to missingness)  
Multiple R-squared: 0.4547, Adjusted R-squared: 0.432  
F-statistic: 20.01 on 7 and 168 DF, p-value: < 2.2e-16
```

```
summary(mod_Eel_pH) #R2 .72
```

```
Call:  
lm(formula = pH ~ pH_E * Temp_E * Sal_E, data = joined_Eel)
```

Residuals:

Min	1Q	Median	3Q	Max
-0.166677	-0.047985	-0.001394	0.054256	0.143649

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	95.120748	47.547700	2.001	0.0471 *
pH_E	-11.347889	6.027607	-1.883	0.0615 .
Temp_E	-4.081968	1.872206	-2.180	0.0306 *
Sal_E	-2.812696	1.562081	-1.801	0.0736 .
pH_E:Temp_E	0.535349	0.238349	2.246	0.0260 *
pH_E:Sal_E	0.366748	0.197951	1.853	0.0657 .
Temp_E:Sal_E	0.130779	0.061581	2.124	0.0352 *
pH_E:Temp_E:Sal_E	-0.017214	0.007834	-2.197	0.0294 *

```
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 0.06889 on 168 degrees of freedom  
(406 observations deleted due to missingness)  
Multiple R-squared: 0.4827, Adjusted R-squared: 0.4612  
F-statistic: 22.4 on 7 and 168 DF, p-value: < 2.2e-16
```

```
summary(mod_Eel_Temp) #R2 .99
```

Call:

```

lm(formula = Temp ~ Temp_E, data = joined_Eel)

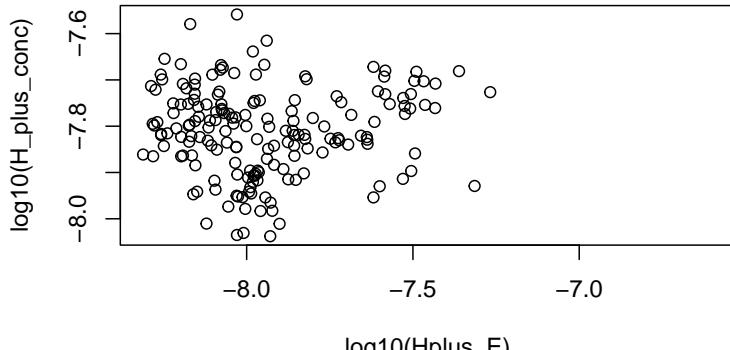
Residuals:
    Min      1Q  Median      3Q     Max 
-3.11962 -0.56847  0.00967  0.58154  2.64312 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 0.29563   0.28097   1.052   0.294    
Temp_E       0.94898   0.01335  71.081  <2e-16 ***  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

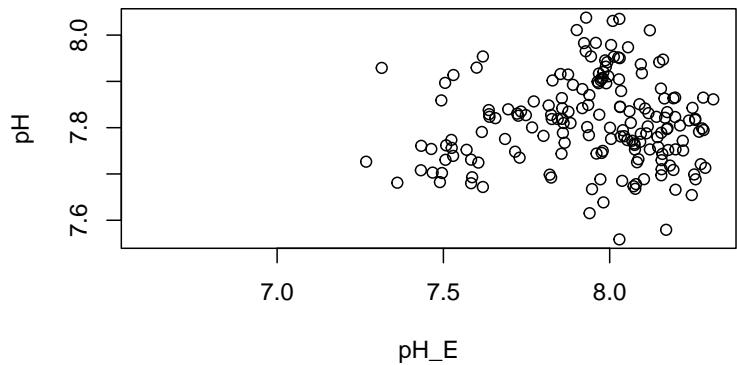
Residual standard error: 0.8546 on 174 degrees of freedom
(406 observations deleted due to missingness)
Multiple R-squared:  0.9667,    Adjusted R-squared:  0.9665 
F-statistic:  5053 on 1 and 174 DF,  p-value: < 2.2e-16

```

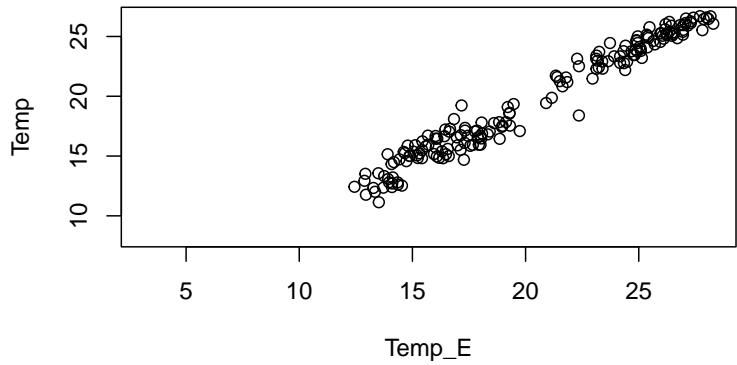
```
plot(data = joined_Eel, log10(H_plus_conc) ~ log10(Hplus_E))
```



```
plot(data = joined_Eel, pH ~ pH_E)
```



```
plot(data = joined_Eel, Temp ~ Temp_E)
```



```
joined_Eel$Temp.pred <- predict.lm(object = mod_Eel_Temp, newdata = joined_Eel,
                                     na.action = na.pass)
joined_Ptown$Temp.pred <- predict.lm(object = mod_Ptown_Temp, newdata = joined_Ptown, na.action =
# joined_Eel$pH.pred <- predict.lm(object = mod_Eel_pH, newdata = joined_Eel, na.action = na.p
# joined_Ptown$pH.pred <- predict.lm(object = mod_Ptown_pH, newdata = joined_Ptown, na.action =
```

```

sampling.pH <- data.frame(
  x = as.POSIXct(c("2022-06-29 00:00:00 EDT","2022-08-2 00:00:00 EDT","2022-09-9 00:00:00 EDT"),
  y = c(7.5,7.5,7.5,7.5,7.5,7.5)
)
sampling.temp <- data.frame(
  x = as.POSIXct(c("2022-06-29 00:00:00 EDT","2022-08-2 00:00:00 EDT","2022-09-9 00:00:00 EDT"),
  y = c(0,0,0,0,0,0)
)

theme_set(theme_bw())

colors <- c(
  "Den"="blue",
  "Eel"="green",
  "Ptown"="orange",
  "Date sampled"="black"
)

#scale_color_manual(values=c("blue", "green", "orange"))

# Now I'm aggregating by day... and this is a bit easier to digest when comparing among sites

gg1 <- ggplot(all.data.dy, aes(x=Date, y=pH, colour = Site))+ 
  geom_point(cex = 1) + 
  geom_line(cex = 1)+ 
  ylim(7.5,8.3)+ 
  xlim(as.POSIXct(c("2022-06-29 00:00:00 EDT","2023-06-10 00:00:00 EDT")))+ 
  geom_point(data = sampling.pH, aes(x,y), color = "black", cex = 1)+ 
  #geom_point(data = joined_Ptown, aes(x=as.POSIXct(Date),y=pH_P), color = "orange", cex = .5)+ 
  #geom_point(data = joined_Eel, aes(x=as.POSIXct(Date),y=pH_E), color = "green", cex = .5)+ 
  scale_color_manual(values=colors) + 
  guides(colour = guide_legend(override.aes = list(size=5)))+ 
  geom_point(data = carb_dat[carb_dat$Stratum=="Bottom water",],aes(x=Date, y = InSitu.pH, cex = 5))+ 
  ggtitle('B. pH')

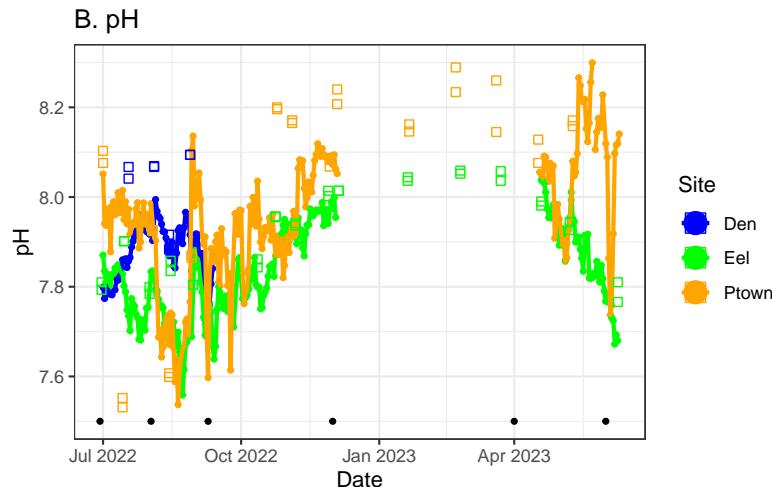
gg1

```

Warning: Removed 5 rows containing missing values (`geom_point()`).

Warning: Removed 2 rows containing missing values (`geom_line()`).

Warning: Removed 18 rows containing missing values (`geom_point()`).



```
ylab = expression("Temperature " ( degree*C))

all.data.dy <- all.data.dy[all.data.dy$Site != "Den",]
all.data.dy$Site <- as.factor(all.data.dy$Site)
all.data.dy$Site <- factor(all.data.dy$Site, levels=c ('Ptown', 'Eel', 'Den'))

gg2 <- ggplot(all.data.dy, aes(x=Date, y=Temp, colour = Site))+
  geom_point(cex = 1) +
  geom_line(cex = 1)+ 
  labs(y=ylab)+ 
  #ylim(14,30)+ 
  xlim(as.POSIXct(c("2022-08-01 00:00:00 EDT","2023-06-15 00:00:00 EDT")))+ 
  geom_point(data = sampling.temp, aes(x,y), color = "black", cex = 1)+ 
  geom_point(data = joined_Ptown, aes(x=as.POSIXct(Date),y=Temp_P), color = "orange", cex = .5)+ 
  geom_point(data = joined_Eel, aes(x=as.POSIXct(Date),y=Temp_E), color = "green", cex = .5)+ 
  #scale_color_manual(values=colors) + 
  guides(colour = guide_legend(override.aes = list(size=5)))+
  geom_point(data = carb_dat[carb_dat$Stratum=="Bottom water",],aes(x=Date, y = Water.Sediment
ggtitle('A. Temp')

gg2
```

Warning: Removed 64 rows containing missing values (`geom_point()`).

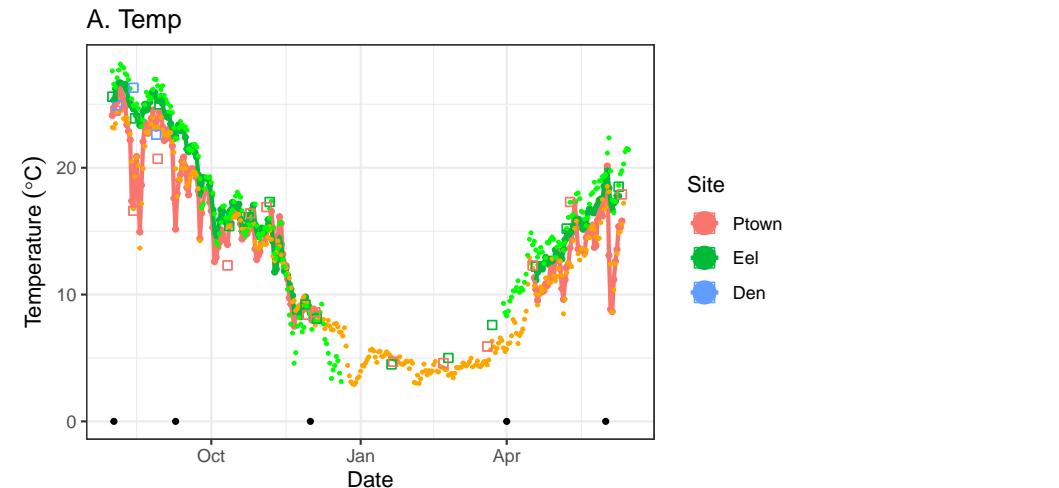
```
Warning: Removed 62 rows containing missing values (`geom_line()`).
```

```
Warning: Removed 1 rows containing missing values (`geom_point()`).
```

```
Warning: Removed 284 rows containing missing values (`geom_point()`).
```

```
Warning: Removed 361 rows containing missing values (`geom_point()`).
```

```
Warning: Removed 24 rows containing missing values (`geom_point()`).
```



```
ggarrange(gg2, gg1, common.legend = TRUE, legend="bottom", ncol = 2)
```

```
Warning: Removed 64 rows containing missing values (`geom_point()`).
```

```
Warning: Removed 62 rows containing missing values (`geom_line()`).
```

```
Warning: Removed 1 rows containing missing values (`geom_point()`).
```

```
Warning: Removed 284 rows containing missing values (`geom_point()`).
```

```
Warning: Removed 361 rows containing missing values (`geom_point()`).
```

Warning: Removed 24 rows containing missing values (`geom_point()`).

Warning: Removed 64 rows containing missing values (`geom_point()`).

Warning: Removed 62 rows containing missing values (`geom_line()`).

Warning: Removed 1 rows containing missing values (`geom_point()`).

Warning: Removed 284 rows containing missing values (`geom_point()`).

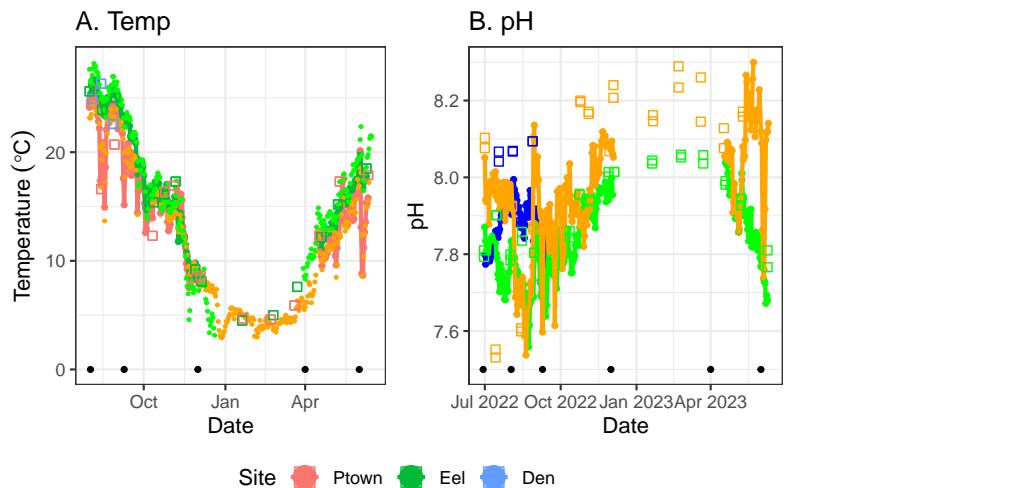
Warning: Removed 361 rows containing missing values (`geom_point()`).

Warning: Removed 24 rows containing missing values (`geom_point()`).

Warning: Removed 5 rows containing missing values (`geom_point()`).

Warning: Removed 2 rows containing missing values (`geom_line()`).

Warning: Removed 18 rows containing missing values (`geom_point()`).



I think I'll go back to the original colors, but this points out the fact that SW temps usually but do not always match. I want to input the HOBO SW temp data and make sure that

this matches with the HOBO temp. I want to check that spike in July in Ptown - is this real or not? I also want to check that the HOBO sensor pH readings are not influenced substantially by fouling. To do this I need to check the cleaning schedule. This also suggests that the Eel Pond pH's of 8.1 in December / Jan may not be correct. The daily averages are so stable this seems wrong. I already cut out bad data (pH 7.5) from the SONDE in Eel Pond so this makes me less confident in the dataset generally if there is no explanation.

0.5 Sediment temp loggers

```
data_dir <- "~/GitHub/EAD-ASEB-Ssolidissima-0A/projects/Seawater data/data/Sediment_temp_loggers"
Eel_sediment <- read.csv(paste(data_dir,"Eel_sediment_425 2023-06-12 13_45_19 EDT (Data EDT)(1).csv"))
names(Eel_sediment) <- c("n","Date","Temp","Button.Down")
str(Eel_sediment)
```

```
'data.frame': 30270 obs. of 7 variables:
 $ n          : int 1 2 3 4 5 6 7 8 9 10 ...
 $ Date       : chr "08/01/2022 07:00:00" "08/01/2022 07:15:00" "08/01/2022 07:30:00" "08/01/2022 07:45:00" ...
 $ Temp        : num 23.8 23.7 23.3 25.7 25.9 ...
 $ Button.Down: chr "" "" "" ...
 $ NA          : chr "" "" ...
 $ NA          : chr ...
 $ NA          : chr ...
```

```
Eel_sediment$Date <- as.POSIXct(Eel_sediment$Date, format = "%m/%d/%Y %H:%M:%S")
gg3 <- gg2 +
  geom_point(data = Eel_sediment, aes(x=as.POSIXct(Date),y=Temp), color = "purple", cex = .5)
gg3
```

Warning: Removed 64 rows containing missing values (`geom_point()`).

Warning: Removed 62 rows containing missing values (`geom_line()`).

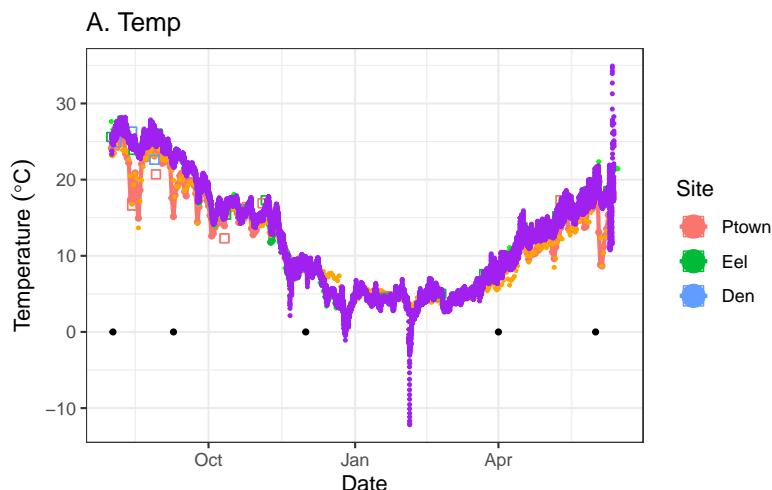
Warning: Removed 1 rows containing missing values (`geom_point()`).

Warning: Removed 284 rows containing missing values (`geom_point()`).

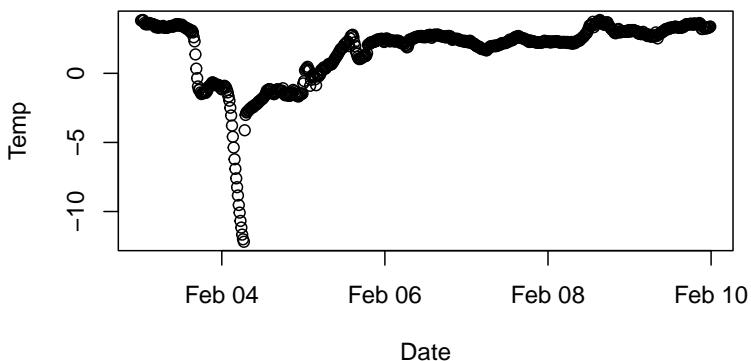
Warning: Removed 361 rows containing missing values (`geom_point()`).

Warning: Removed 24 rows containing missing values (`geom_point()`).

Warning: Removed 3 rows containing missing values (`geom_point()`).



```
h <- Eel_sediment[Eel_sediment$Date > "2023-02-03" & Eel_sediment$Date < "2023-02-10",]  
plot(data = h, Temp~Date) # -10C is quite low. However, we went out on Feb 21st and late Jan as ...
```

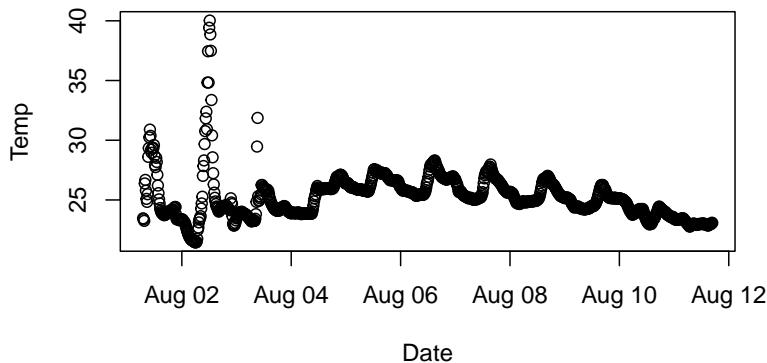


```
data_dir <- "~/GitHub/EAD-ASEB-Ssolidissima-0A/projects/Seawater data/data/Sediment_temp_logger"
Ptown_sediment <- read.csv(paste(data_dir,"Ptown_sediment_489 2023-07-19 10_48_26 EDT (Data ED"))
names(Ptown_sediment) <- c("n","Date","Temp","Button.Down")
str(Ptown_sediment)
```

```
'data.frame': 33812 obs. of 8 variables:
$ n          : int 1 2 3 4 5 6 7 8 9 10 ...
$ Date       : chr "08/01/2022 07:00:00" "08/01/2022 07:15:00" "08/01/2022 07:30:00" "08/01/2022 07:45:00" ...
$ Temp        : num 23.5 23.4 23.2 26.4 26.7 ...
$ Button.Down: chr "" "" "" ...
$ NA          : chr "" "" ...
$ NA          : chr "" ...
$ NA          : chr ...
$ NA          : chr ...
```

```
Ptown_sediment$Date <- as.POSIXct(Ptown_sediment$Date, format = "%m/%d/%Y %H:%M:%S")

h <- head(Ptown_sediment,1000)
plot(data = h,Temp~Date) #I'm confused because it looks like there are diurnal tidal cycles but
```



```
Ptown_sediment <- Ptown_sediment[Ptown_sediment$Date > "2022-08-05",]
gg3 <- gg2+
```

```
geom_point(data = Ptown_sediment, aes(x=Date,y=Temp), color = "purple", cex = .1)  
gg3
```

Warning: Removed 64 rows containing missing values (`geom_point()`).

Warning: Removed 62 rows containing missing values (`geom_line()`).

Warning: Removed 1 rows containing missing values (`geom_point()`).

Warning: Removed 284 rows containing missing values (`geom_point()`).

Warning: Removed 361 rows containing missing values (`geom_point()`).

Warning: Removed 24 rows containing missing values (`geom_point()`).

Warning: Removed 3311 rows containing missing values (`geom_point()`).



0.6 Conductivity

Eel Pond Fall

```

data_dir <- "~/GitHub/EAD-ASEB-Ssolidissima-0A/projects/Seawater data/data/Conductivity/"
Eel_cond <- read.csv(paste(data_dir,"20674248_Eel_Sal.csv",sep=""), skip = 2, stringsAsFactors = TRUE)
names(Eel_cond) <- c("n","Date","Cond_High","Temp","Sp.Cond","Sal")
tail(Eel_cond,500)

```

	n	Date	Cond_High	Temp	Sp.Cond	Sal
6539	6540	12/03/22 08:45:00 AM	27148.1	8.72	49261.5	32.1959
6540	6541	12/03/22 09:00:00 AM	27117.5	8.70	49231.4	32.1738
6541	6542	12/03/22 09:15:00 AM	27093.6	8.72	49163.4	32.1241
6542	6543	12/03/22 09:30:00 AM	27041.4	8.71	49082.0	32.0644
6543	6544	12/03/22 09:45:00 AM	27148.1	8.74	49236.5	32.1776
6544	6545	12/03/22 10:00:00 AM	27069.7	8.76	49070.6	32.0561
6545	6546	12/03/22 10:15:00 AM	27034.9	8.78	48983.3	31.9922
6546	6547	12/03/22 10:30:00 AM	27034.9	8.81	48946.0	31.9649
6547	6548	12/03/22 10:45:00 AM	26993.8	8.77	48921.8	31.9472
6548	6549	12/03/22 11:00:00 AM	26976.6	8.81	48841.4	31.8884
6549	6550	12/03/22 11:15:00 AM	27108.8	8.89	48979.4	31.9893
6550	6551	12/03/22 11:30:00 AM	27161.2	8.95	48998.8	32.0036
6551	6552	12/03/22 11:45:00 AM	27330.6	8.98	49264.5	32.1981
6552	6553	12/03/22 12:00:00 PM	27390.5	9.00	49346.6	32.2582
6553	6554	12/03/22 12:15:00 PM	27428.3	9.05	49351.8	32.2620
6554	6555	12/03/22 12:30:00 PM	27446.1	9.08	49346.2	32.2579
6555	6556	12/03/22 12:45:00 PM	27439.4	9.09	49321.8	32.2401
6556	6557	12/03/22 01:00:00 PM	27439.4	9.15	49247.3	32.1855
6557	6558	12/03/22 01:15:00 PM	27426.1	9.15	49223.6	32.1681
6558	6559	12/03/22 01:30:00 PM	27312.9	9.09	49096.3	32.0749
6559	6560	12/03/22 01:45:00 PM	27359.4	9.06	49216.4	32.1629
6560	6561	12/03/22 02:00:00 PM	27282.0	9.10	49028.9	32.0256
6561	6562	12/03/22 02:15:00 PM	27343.9	9.12	49114.4	32.0882
6562	6563	12/03/22 02:30:00 PM	27306.3	9.15	49010.4	32.0120
6563	6564	12/03/22 02:45:00 PM	27359.4	9.23	49006.1	32.0089
6564	6565	12/03/22 03:00:00 PM	27328.4	9.23	48951.0	31.9686
6565	6566	12/03/22 03:15:00 PM	27097.9	9.12	48676.3	31.7677
6566	6567	12/03/22 03:30:00 PM	27240.1	9.14	48904.9	31.9348
6567	6568	12/03/22 03:45:00 PM	27522.2	9.34	49159.2	32.1209
6568	6569	12/03/22 04:00:00 PM	27805.5	9.60	49338.5	32.2523
6569	6570	12/03/22 04:15:00 PM	27869.7	9.59	49463.9	32.3442
6570	6571	12/03/22 04:30:00 PM	27862.8	9.58	49464.1	32.3443
6571	6572	12/03/22 04:45:00 PM	27837.6	9.58	49419.7	32.3118
6572	6573	12/03/22 05:00:00 PM	27798.6	9.57	49363.4	32.2705

6573	6574	12/03/22	05:15:00	PM	27805.5	9.58	49363.2	32.2704
6574	6575	12/03/22	05:30:00	PM	27741.5	9.61	49213.6	32.1608
6575	6576	12/03/22	05:45:00	PM	27874.3	9.69	49348.5	32.2596
6576	6577	12/03/22	06:00:00	PM	27918.1	9.76	49339.2	32.2528
6577	6578	12/03/22	06:15:00	PM	27885.8	9.77	49270.2	32.2023
6578	6579	12/03/22	06:30:00	PM	27973.6	9.79	49399.5	32.2970
6579	6580	12/03/22	06:45:00	PM	27980.5	9.81	49387.0	32.2878
6580	6581	12/03/22	07:00:00	PM	28006.1	9.82	49419.5	32.3116
6581	6582	12/03/22	07:15:00	PM	28010.7	9.84	49402.9	32.2995
6582	6583	12/03/22	07:30:00	PM	27895.1	9.86	49176.1	32.1334
6583	6584	12/03/22	07:45:00	PM	27768.9	9.90	48906.7	31.9362
6584	6585	12/03/22	08:00:00	PM	27732.4	9.95	48782.2	31.8451
6585	6586	12/03/22	08:15:00	PM	27787.2	9.97	48853.5	31.8973
6586	6587	12/03/22	08:30:00	PM	27823.8	9.93	48966.0	31.9796
6587	6588	12/03/22	08:45:00	PM	27805.5	9.93	48934.1	31.9562
6588	6589	12/03/22	09:00:00	PM	27837.6	9.91	49014.5	32.0151
6589	6590	12/03/22	09:15:00	PM	27950.4	9.97	49138.2	32.1056
6590	6591	12/03/22	09:30:00	PM	27952.8	9.99	49117.9	32.0908
6591	6592	12/03/22	09:45:00	PM	27925.0	10.01	49045.1	32.0374
6592	6593	12/03/22	10:00:00	PM	27849.1	9.99	48937.2	31.9585
6593	6594	12/03/22	10:15:00	PM	27842.2	9.99	48925.1	31.9496
6594	6595	12/03/22	10:30:00	PM	27837.6	10.01	48892.8	31.9260
6595	6596	12/03/22	10:45:00	PM	27862.8	10.02	48924.6	31.9492
6596	6597	12/03/22	11:00:00	PM	27876.6	10.02	48948.6	31.9668
6597	6598	12/03/22	11:15:00	PM	27881.2	10.02	48956.6	31.9727
6598	6599	12/03/22	11:30:00	PM	27816.9	9.95	48929.7	31.9530
6599	6600	12/03/22	11:45:00	PM	27748.4	9.87	48907.4	31.9367
6600	6601	12/04/22	12:00:00	AM	27700.6	9.82	48884.7	31.9200
6601	6602	12/04/22	12:15:00	AM	27664.3	9.79	48857.6	31.9003
6602	6603	12/04/22	12:30:00	AM	27614.5	9.77	48794.7	31.8543
6603	6604	12/04/22	12:45:00	AM	27517.7	9.65	48770.9	31.8369
6604	6605	12/04/22	01:00:00	AM	27282.0	9.59	48429.2	31.5871
6605	6606	12/04/22	01:15:00	AM	27352.7	9.55	48602.3	31.7136
6606	6607	12/04/22	01:30:00	AM	27224.7	9.48	48461.6	31.6108
6607	6608	12/04/22	01:45:00	AM	27058.8	9.38	48289.7	31.4853
6608	6609	12/04/22	02:00:00	AM	26983.0	9.36	48179.7	31.4050
6609	6610	12/04/22	02:15:00	AM	26832.7	9.25	48046.2	31.3076
6610	6611	12/04/22	02:30:00	AM	26766.7	9.11	48098.4	31.3457
6611	6612	12/04/22	02:45:00	AM	26732.7	9.10	48049.9	31.3103
6612	6613	12/04/22	03:00:00	AM	26641.9	8.98	48033.4	31.2983
6613	6614	12/04/22	03:15:00	AM	26549.5	8.73	48172.3	31.3996

6614	6615	12/04/22	03:30:00	AM	26553.7	8.83	48057.9	31.3161
6615	6616	12/04/22	03:45:00	AM	26503.6	8.77	48041.0	31.3038
6616	6617	12/04/22	04:00:00	AM	26499.4	8.73	48082.2	31.3339
6617	6618	12/04/22	04:15:00	AM	26346.0	8.45	48148.0	31.3819
6618	6619	12/04/22	04:30:00	AM	26564.2	8.85	48052.3	31.3121
6619	6620	12/04/22	04:45:00	AM	27268.7	9.17	48918.8	31.9450
6620	6621	12/04/22	05:00:00	AM	27374.9	9.21	49058.3	32.0471
6621	6622	12/04/22	05:15:00	AM	27455.1	9.26	49139.0	32.1062
6622	6623	12/04/22	05:30:00	AM	27497.6	9.28	49189.7	32.1433
6623	6624	12/04/22	05:45:00	AM	27517.7	9.27	49237.7	32.1785
6624	6625	12/04/22	06:00:00	AM	27540.2	9.26	49290.0	32.2168
6625	6626	12/04/22	06:15:00	AM	27578.4	9.28	49333.0	32.2483
6626	6627	12/04/22	06:30:00	AM	27553.7	9.28	49289.2	32.2162
6627	6628	12/04/22	06:45:00	AM	27508.8	9.27	49221.9	32.1669
6628	6629	12/04/22	07:00:00	AM	27511.0	9.31	49176.4	32.1335
6629	6630	12/04/22	07:15:00	AM	27486.4	9.27	49182.2	32.1378
6630	6631	12/04/22	07:30:00	AM	27490.8	9.27	49190.0	32.1435
6631	6632	12/04/22	07:45:00	AM	27576.2	9.34	49254.8	32.1910
6632	6633	12/04/22	08:00:00	AM	27621.3	9.32	49359.5	32.2677
6633	6634	12/04/22	08:15:00	AM	27643.9	9.33	49387.1	32.2879
6634	6635	12/04/22	08:30:00	AM	27553.7	9.16	49438.2	32.3254
6635	6636	12/04/22	08:45:00	AM	27623.5	9.23	49475.2	32.3525
6636	6637	12/04/22	09:00:00	AM	27677.9	9.27	49522.0	32.3868
6637	6638	12/04/22	09:15:00	AM	27682.4	9.28	49517.5	32.3835
6638	6639	12/04/22	09:30:00	AM	27714.2	9.33	49511.7	32.3792
6639	6640	12/04/22	09:45:00	AM	27659.7	9.26	49502.1	32.3722
6640	6641	12/04/22	10:00:00	AM	27639.4	9.23	49503.4	32.3732
6641	6642	12/04/22	10:15:00	AM	27671.1	9.23	49559.7	32.4144
6642	6643	12/04/22	10:30:00	AM	27677.9	9.36	49410.1	32.3048
6643	6644	12/04/22	10:45:00	AM	27673.3	9.34	49426.8	32.3170
6644	6645	12/04/22	11:00:00	AM	27662.0	9.22	49556.0	32.4117
6645	6646	12/04/22	11:15:00	AM	27533.4	9.19	49364.8	32.2716
6646	6647	12/04/22	11:30:00	AM	27346.1	8.88	49416.9	32.3098
6647	6648	12/04/22	11:45:00	AM	27614.5	9.17	49533.9	32.3955
6648	6649	12/04/22	12:00:00	PM	27662.0	9.23	49543.6	32.4026
6649	6650	12/04/22	12:15:00	PM	27637.1	9.20	49536.7	32.3976
6650	6651	12/04/22	12:30:00	PM	27310.7	9.00	49204.0	32.1538
6651	6652	12/04/22	12:45:00	PM	27401.6	9.03	49329.0	32.2454
6652	6653	12/04/22	01:00:00	PM	27441.7	9.06	49363.2	32.2704
6653	6654	12/04/22	01:15:00	PM	27354.9	9.06	49208.4	32.1570
6654	6655	12/04/22	01:30:00	PM	27455.1	9.12	49312.5	32.2332

6655	6656	12/04/22	01:45:00	PM	27152.4	9.04	48871.9	31.9107
6656	6657	12/04/22	02:00:00	PM	27058.8	9.05	48692.5	31.7795
6657	6658	12/04/22	02:15:00	PM	26888.4	9.10	48327.4	31.5128
6658	6659	12/04/22	02:30:00	PM	26888.4	9.10	48327.4	31.5128
6659	6660	12/04/22	02:45:00	PM	26952.8	9.08	48466.7	31.6145
6660	6661	12/04/22	03:00:00	PM	26939.9	9.05	48480.4	31.6245
6661	6662	12/04/22	03:15:00	PM	26939.9	9.00	48541.6	31.6693
6662	6663	12/04/22	03:30:00	PM	26888.4	9.01	48437.4	31.5931
6663	6664	12/04/22	03:45:00	PM	26837.0	8.98	48382.2	31.5528
6664	6665	12/04/22	04:00:00	PM	26747.6	9.00	48198.0	31.4184
6665	6666	12/04/22	04:15:00	PM	26713.7	8.94	48210.5	31.4275
6666	6667	12/04/22	04:30:00	PM	26245.4	8.71	47649.5	31.0184
6667	6668	12/04/22	04:45:00	PM	26470.2	8.88	47847.4	31.1626
6668	6669	12/04/22	05:00:00	PM	27211.6	9.16	48829.5	31.8797
6669	6670	12/04/22	05:15:00	PM	27426.1	9.16	49211.2	32.1591
6670	6671	12/04/22	05:30:00	PM	27481.9	9.17	49298.1	32.2227
6671	6672	12/04/22	05:45:00	PM	27499.8	9.21	49280.3	32.2096
6672	6673	12/04/22	06:00:00	PM	27612.3	9.20	49492.6	32.3653
6673	6674	12/04/22	06:15:00	PM	27634.8	9.20	49532.6	32.3946
6674	6675	12/04/22	06:30:00	PM	27495.3	9.22	49259.9	32.1947
6675	6676	12/04/22	06:45:00	PM	27531.2	9.20	49348.5	32.2596
6676	6677	12/04/22	07:00:00	PM	27659.7	9.19	49589.4	32.4362
6677	6678	12/04/22	07:15:00	PM	27587.4	9.20	49448.4	32.3328
6678	6679	12/04/22	07:30:00	PM	27623.5	9.20	49512.5	32.3799
6679	6680	12/04/22	07:45:00	PM	27598.7	9.16	49518.3	32.3841
6680	6681	12/04/22	08:00:00	PM	27723.3	9.22	49664.9	32.4916
6681	6682	12/04/22	08:15:00	PM	27582.9	9.20	49440.4	32.3270
6682	6683	12/04/22	08:30:00	PM	27610.0	9.11	49600.8	32.4446
6683	6684	12/04/22	08:45:00	PM	27634.8	9.17	49570.0	32.4220
6684	6685	12/04/22	09:00:00	PM	27657.5	9.15	49635.4	32.4699
6685	6686	12/04/22	09:15:00	PM	27481.9	9.10	49385.1	32.2864
6686	6687	12/04/22	09:30:00	PM	27490.8	8.87	49688.7	32.5091
6687	6688	12/04/22	09:45:00	PM	27255.5	8.98	49130.2	32.0998
6688	6689	12/04/22	10:00:00	PM	27286.4	8.92	49260.1	32.1949
6689	6690	12/04/22	10:15:00	PM	27412.7	8.93	49473.7	32.3513
6690	6691	12/04/22	10:30:00	PM	27533.4	8.96	49652.0	32.4821
6691	6692	12/04/22	10:45:00	PM	27477.4	8.98	49526.8	32.3903
6692	6693	12/04/22	11:00:00	PM	27472.9	8.93	49581.4	32.4303
6693	6694	12/04/22	11:15:00	PM	27515.5	8.88	49720.4	32.5323
6694	6695	12/04/22	11:30:00	PM	27497.6	8.83	49751.3	32.5549
6695	6696	12/04/22	11:45:00	PM	27286.4	8.73	49497.7	32.3690

6696	6697	12/05/22	12:00:00	AM	26920.6	8.38	49276.7	32.2070
6697	6698	12/05/22	12:15:00	AM	26405.9	7.87	48978.8	31.9889
6698	6699	12/05/22	12:30:00	AM	26568.4	8.03	49075.2	32.0595
6699	6700	12/05/22	12:45:00	AM	26080.6	7.48	48870.6	31.9097
6700	6701	12/05/22	01:00:00	AM	26198.4	7.47	49101.9	32.0790
6701	6702	12/05/22	01:15:00	AM	26125.2	7.64	48751.2	31.8224
6702	6703	12/05/22	01:30:00	AM	26184.1	7.59	48923.2	31.9483
6703	6704	12/05/22	01:45:00	AM	25690.1	7.21	48484.4	31.6275
6704	6705	12/05/22	02:00:00	AM	25588.3	7.13	48394.9	31.5621
6705	6706	12/05/22	02:15:00	AM	25312.3	6.95	48103.4	31.3493
6706	6707	12/05/22	02:30:00	AM	26509.8	8.01	48993.1	31.9994
6707	6708	12/05/22	02:45:00	AM	25427.4	7.21	47993.4	31.2691
6708	6709	12/05/22	03:00:00	AM	25741.2	7.36	48390.9	31.5592
6709	6710	12/05/22	03:15:00	AM	25701.9	7.41	48255.1	31.4600
6710	6711	12/05/22	03:30:00	AM	25969.9	7.70	48389.0	31.5578
6711	6712	12/05/22	03:45:00	AM	26282.3	7.82	48814.4	31.8687
6712	6713	12/05/22	04:00:00	AM	26145.5	7.74	48662.9	31.7579
6713	6714	12/05/22	04:15:00	AM	25790.6	7.20	48684.9	31.7740
6714	6715	12/05/22	04:30:00	AM	25899.9	7.44	48585.5	31.7013
6715	6716	12/05/22	04:45:00	AM	26957.2	8.22	49545.1	32.4037
6716	6717	12/05/22	05:00:00	AM	27421.6	8.67	49816.6	32.6029
6717	6718	12/05/22	05:15:00	AM	27432.8	8.62	49900.1	32.6641
6718	6719	12/05/22	05:30:00	AM	27359.4	8.61	49780.4	32.5763
6719	6720	12/05/22	05:45:00	AM	27401.6	8.61	49856.5	32.6321
6720	6721	12/05/22	06:00:00	AM	27282.0	8.60	49653.4	32.4831
6721	6722	12/05/22	06:15:00	AM	27386.0	8.58	49866.3	32.6394
6722	6723	12/05/22	06:30:00	AM	27386.0	8.57	49879.0	32.6487
6723	6724	12/05/22	06:45:00	AM	27403.8	8.60	49873.1	32.6443
6724	6725	12/05/22	07:00:00	AM	27306.3	8.57	49735.1	32.5431
6725	6726	12/05/22	07:15:00	AM	26918.4	8.62	48972.5	31.9843
6726	6727	12/05/22	07:30:00	AM	27397.1	8.56	49911.7	32.6727
6727	6728	12/05/22	07:45:00	AM	27417.2	8.57	49935.3	32.6900
6728	6729	12/05/22	08:00:00	AM	27421.6	8.56	49956.0	32.7052
6729	6730	12/05/22	08:15:00	AM	27488.6	8.62	50000.7	32.7380
6730	6731	12/05/22	08:30:00	AM	27439.4	8.60	49937.3	32.6915
6731	6732	12/05/22	08:45:00	AM	27526.7	8.62	50069.3	32.7884
6732	6733	12/05/22	09:00:00	AM	26980.9	8.63	49072.7	32.0577
6733	6734	12/05/22	09:15:00	AM	27555.9	8.60	50147.5	32.8458
6734	6735	12/05/22	09:30:00	AM	27432.8	8.56	49976.2	32.7200
6735	6736	12/05/22	09:45:00	AM	27392.7	8.56	49903.8	32.6669
6736	6737	12/05/22	10:00:00	AM	27567.2	8.62	50142.4	32.8421

6737	6738	12/05/22	10:15:00	AM	27310.7	8.63	49667.3	32.4933
6738	6739	12/05/22	10:30:00	AM	26955.0	8.62	49038.5	32.0326
6739	6740	12/05/22	10:45:00	AM	27589.7	8.61	50195.7	32.8813
6740	6741	12/05/22	11:00:00	AM	27625.8	8.62	50248.0	32.9197
6741	6742	12/05/22	11:15:00	AM	26399.7	8.63	48024.7	31.2919
6742	6743	12/05/22	11:30:00	AM	27668.8	8.71	50210.7	32.8923
6743	6744	12/05/22	11:45:00	AM	27634.8	8.70	50162.3	32.8567
6744	6745	12/05/22	12:00:00	PM	27709.7	8.81	50157.1	32.8529
6745	6746	12/05/22	12:15:00	PM	27689.2	8.78	50158.4	32.8538
6746	6747	12/05/22	12:30:00	PM	27700.6	8.81	50140.7	32.8409
6747	6748	12/05/22	12:45:00	PM	27619.0	8.82	49981.7	32.7240
6748	6749	12/05/22	01:00:00	PM	27700.6	8.79	50166.1	32.8595
6749	6750	12/05/22	01:15:00	PM	27668.8	8.79	50109.0	32.8176
6750	6751	12/05/22	01:30:00	PM	27709.7	8.79	50182.5	32.8715
6751	6752	12/05/22	01:45:00	PM	27664.3	8.78	50113.7	32.8210
6752	6753	12/05/22	02:00:00	PM	27625.8	8.77	50057.2	32.7795
6753	6754	12/05/22	02:15:00	PM	27591.9	8.72	50059.7	32.7814
6754	6755	12/05/22	02:30:00	PM	27564.9	8.73	49998.5	32.7364
6755	6756	12/05/22	02:45:00	PM	27484.1	8.68	49916.5	32.6762
6756	6757	12/05/22	03:00:00	PM	27472.9	8.71	49858.3	32.6335
6757	6758	12/05/22	03:15:00	PM	27348.3	8.68	49672.0	32.4968
6758	6759	12/05/22	03:30:00	PM	27251.1	8.62	49572.4	32.4238
6759	6760	12/05/22	03:45:00	PM	27183.1	8.56	49525.3	32.3892
6760	6761	12/05/22	04:00:00	PM	27078.3	8.50	49411.4	32.3057
6761	6762	12/05/22	04:15:00	PM	26996.0	8.45	49325.3	32.2426
6762	6763	12/05/22	04:30:00	PM	26924.9	8.39	49271.9	32.2035
6763	6764	12/05/22	04:45:00	PM	26849.8	8.34	49198.4	32.1497
6764	6765	12/05/22	05:00:00	PM	917.9	7.15	1754.5	0.8865
6765	6766	12/05/22	05:15:00	PM	825.8	6.67	1599.3	0.8045
6766	6767	12/05/22	05:30:00	PM	777.0	6.39	1516.4	0.7610
6767	6768	12/05/22	05:45:00	PM	563.3	7.14	1077.2	0.5328
6768	6769	12/05/22	06:00:00	PM	531.3	7.62	1002.9	0.4947
6769	6770	12/05/22	06:15:00	PM	550.5	9.72	983.1	0.4846
6770	6771	12/05/22	06:30:00	PM	561.9	12.21	942.0	0.4636
6771	6772	12/05/22	06:45:00	PM	563.8	13.64	912.6	0.4486
6772	6773	12/05/22	07:00:00	PM	568.8	14.08	910.9	0.4478
6773	6774	12/05/22	07:15:00	PM	603.6	14.60	954.7	0.4701
6774	6775	12/05/22	07:30:00	PM	640.2	15.33	995.2	0.4907
6775	6776	12/05/22	07:45:00	PM	663.4	16.16	1011.3	0.4990
6776	6777	12/05/22	08:00:00	PM	671.0	15.56	1037.4	0.5124
6777	6778	12/05/22	08:15:00	PM	673.5	14.25	1074.2	0.5312

6778	6779	12/05/22	08:30:00	PM	617.1	14.20	985.4	0.4858
6779	6780	12/05/22	08:45:00	PM	514.7	14.75	811.2	0.3972
6780	6781	12/05/22	09:00:00	PM	338.1	15.07	528.8	0.2561
6781	6782	12/05/22	09:15:00	PM	251.6	14.94	394.8	0.1905
6782	6783	12/05/22	09:30:00	PM	307.8	15.55	476.0	0.2301
6783	6784	12/05/22	09:45:00	PM	225.7	15.25	351.5	0.1695
6784	6785	12/05/22	10:00:00	PM	80.4	14.09	128.7	0.0647
6785	6786	12/05/22	10:15:00	PM	0.0	15.56	0.0	0.0123
6786	6787	12/05/22	10:30:00	PM	0.0	17.93	0.0	0.0123
6787	6788	12/05/22	10:45:00	PM	0.0	19.46	0.0	0.0123
6788	6789	12/05/22	11:00:00	PM	0.0	20.33	0.0	0.0123
6789	6790	12/05/22	11:15:00	PM	0.0	20.80	0.0	0.0123
6790	6791	12/05/22	11:30:00	PM	0.0	21.04	0.0	0.0123
6791	6792	12/05/22	11:45:00	PM	0.0	21.14	0.0	0.0123
6792	6793	12/06/22	12:00:00	AM	0.0	21.30	0.0	0.0123
6793	6794	12/06/22	12:15:00	AM	0.0	21.46	0.0	0.0123
6794	6795	12/06/22	12:30:00	AM	0.0	21.58	0.0	0.0123
6795	6796	12/06/22	12:45:00	AM	0.0	21.56	0.0	0.0123
6796	6797	12/06/22	01:00:00	AM	0.0	21.60	0.0	0.0123
6797	6798	12/06/22	01:15:00	AM	0.0	21.76	0.0	0.0123
6798	6799	12/06/22	01:30:00	AM	0.0	21.83	0.0	0.0123
6799	6800	12/06/22	01:45:00	AM	0.0	21.80	0.0	0.0123
6800	6801	12/06/22	02:00:00	AM	0.0	21.70	0.0	0.0123
6801	6802	12/06/22	02:15:00	AM	0.0	21.75	0.0	0.0123
6802	6803	12/06/22	02:30:00	AM	0.0	21.81	0.0	0.0123
6803	6804	12/06/22	02:45:00	AM	0.0	21.92	0.0	0.0123
6804	6805	12/06/22	03:00:00	AM	0.0	21.92	0.0	0.0123
6805	6806	12/06/22	03:15:00	AM	0.0	21.86	0.0	0.0123
6806	6807	12/06/22	03:30:00	AM	0.0	21.92	0.0	0.0123
6807	6808	12/06/22	03:45:00	AM	0.0	21.96	0.0	0.0123
6808	6809	12/06/22	04:00:00	AM	0.0	21.92	0.0	0.0123
6809	6810	12/06/22	04:15:00	AM	0.0	21.81	0.0	0.0123
6810	6811	12/06/22	04:30:00	AM	0.0	21.86	0.0	0.0123
6811	6812	12/06/22	04:45:00	AM	0.0	21.98	0.0	0.0123
6812	6813	12/06/22	05:00:00	AM	0.0	22.02	0.0	0.0123
6813	6814	12/06/22	05:15:00	AM	0.0	22.00	0.0	0.0123
6814	6815	12/06/22	05:30:00	AM	0.0	21.87	0.0	0.0123
6815	6816	12/06/22	05:45:00	AM	0.0	21.87	0.0	0.0123
6816	6817	12/06/22	06:00:00	AM	0.0	21.96	0.0	0.0123
6817	6818	12/06/22	06:15:00	AM	0.0	22.00	0.0	0.0123
6818	6819	12/06/22	06:30:00	AM	0.0	22.01	0.0	0.0123

6819	6820	12/06/22	06:45:00	AM	0.0	21.90	0.0	0.0123
6820	6821	12/06/22	07:00:00	AM	0.0	21.95	0.0	0.0123
6821	6822	12/06/22	07:15:00	AM	0.0	22.00	0.0	0.0123
6822	6823	12/06/22	07:30:00	AM	0.0	22.02	0.0	0.0123
6823	6824	12/06/22	07:45:00	AM	0.0	21.95	0.0	0.0123
6824	6825	12/06/22	08:00:00	AM	0.0	21.89	0.0	0.0123
6825	6826	12/06/22	08:15:00	AM	0.0	22.01	0.0	0.0123
6826	6827	12/06/22	08:30:00	AM	0.0	22.12	0.0	0.0123
6827	6828	12/06/22	08:45:00	AM	0.0	22.11	0.0	0.0123
6828	6829	12/06/22	09:00:00	AM	0.0	21.98	0.0	0.0123
6829	6830	12/06/22	09:15:00	AM	0.0	22.07	0.0	0.0123
6830	6831	12/06/22	09:30:00	AM	0.0	22.16	0.0	0.0123
6831	6832	12/06/22	09:45:00	AM	0.0	22.19	0.0	0.0123
6832	6833	12/06/22	10:00:00	AM	0.0	22.13	0.0	0.0123
6833	6834	12/06/22	10:15:00	AM	0.0	22.13	0.0	0.0123
6834	6835	12/06/22	10:30:00	AM	0.0	22.21	0.0	0.0123
6835	6836	12/06/22	10:45:00	AM	0.0	22.22	0.0	0.0123
6836	6837	12/06/22	11:00:00	AM	0.0	22.11	0.0	0.0123
6837	6838	12/06/22	11:15:00	AM	0.0	22.19	0.0	0.0123
6838	6839	12/06/22	11:30:00	AM	0.0	22.27	0.0	0.0123
6839	6840	12/06/22	11:45:00	AM	0.0	22.22	0.0	0.0123
6840	6841	12/06/22	12:00:00	PM	0.0	22.23	0.0	0.0123
6841	6842	12/06/22	12:15:00	PM	0.0	22.29	0.0	0.0123
6842	6843	12/06/22	12:30:00	PM	0.0	22.28	0.0	0.0123
6843	6844	12/06/22	12:45:00	PM	0.0	22.21	0.0	0.0123
6844	6845	12/06/22	01:00:00	PM	0.0	22.32	0.0	0.0123
6845	6846	12/06/22	01:15:00	PM	0.0	22.36	0.0	0.0123
6846	6847	12/06/22	01:30:00	PM	0.0	22.25	0.0	0.0123
6847	6848	12/06/22	01:45:00	PM	0.0	22.27	0.0	0.0123
6848	6849	12/06/22	02:00:00	PM	0.0	22.34	0.0	0.0123
6849	6850	12/06/22	02:15:00	PM	0.0	22.32	0.0	0.0123
6850	6851	12/06/22	02:30:00	PM	0.0	22.30	0.0	0.0123
6851	6852	12/06/22	02:45:00	PM	0.0	22.39	0.0	0.0123
6852	6853	12/06/22	03:00:00	PM	0.0	22.38	0.0	0.0123
6853	6854	12/06/22	03:15:00	PM	0.0	22.32	0.0	0.0123
6854	6855	12/06/22	03:30:00	PM	0.0	22.39	0.0	0.0123
6855	6856	12/06/22	03:45:00	PM	0.0	22.40	0.0	0.0123
6856	6857	12/06/22	04:00:00	PM	0.0	22.35	0.0	0.0123
6857	6858	12/06/22	04:15:00	PM	0.0	22.43	0.0	0.0123
6858	6859	12/06/22	04:30:00	PM	0.0	22.46	0.0	0.0123
6859	6860	12/06/22	04:45:00	PM	0.0	22.38	0.0	0.0123

6860	6861	12/06/22	05:00:00	PM	0.0	22.48	0.0	0.0123
6861	6862	12/06/22	05:15:00	PM	0.0	22.45	0.0	0.0123
6862	6863	12/06/22	05:30:00	PM	0.0	22.48	0.0	0.0123
6863	6864	12/06/22	05:45:00	PM	0.0	22.51	0.0	0.0123
6864	6865	12/06/22	06:00:00	PM	0.0	22.39	0.0	0.0123
6865	6866	12/06/22	06:15:00	PM	0.0	22.43	0.0	0.0123
6866	6867	12/06/22	06:30:00	PM	0.0	22.49	0.0	0.0123
6867	6868	12/06/22	06:45:00	PM	0.0	22.39	0.0	0.0123
6868	6869	12/06/22	07:00:00	PM	0.0	22.49	0.0	0.0123
6869	6870	12/06/22	07:15:00	PM	0.0	22.52	0.0	0.0123
6870	6871	12/06/22	07:30:00	PM	0.0	22.40	0.0	0.0123
6871	6872	12/06/22	07:45:00	PM	0.0	22.48	0.0	0.0123
6872	6873	12/06/22	08:00:00	PM	0.0	22.50	0.0	0.0123
6873	6874	12/06/22	08:15:00	PM	0.0	22.38	0.0	0.0123
6874	6875	12/06/22	08:30:00	PM	0.0	22.48	0.0	0.0123
6875	6876	12/06/22	08:45:00	PM	0.0	22.52	0.0	0.0123
6876	6877	12/06/22	09:00:00	PM	0.0	22.41	0.0	0.0123
6877	6878	12/06/22	09:15:00	PM	0.0	22.50	0.0	0.0123
6878	6879	12/06/22	09:30:00	PM	0.0	22.50	0.0	0.0123
6879	6880	12/06/22	09:45:00	PM	0.0	22.40	0.0	0.0123
6880	6881	12/06/22	10:00:00	PM	0.0	22.51	0.0	0.0123
6881	6882	12/06/22	10:15:00	PM	0.0	22.52	0.0	0.0123
6882	6883	12/06/22	10:30:00	PM	0.0	22.45	0.0	0.0123
6883	6884	12/06/22	10:45:00	PM	0.0	22.48	0.0	0.0123
6884	6885	12/06/22	11:00:00	PM	0.0	22.46	0.0	0.0123
6885	6886	12/06/22	11:15:00	PM	0.0	22.39	0.0	0.0123
6886	6887	12/06/22	11:30:00	PM	0.0	22.50	0.0	0.0123
6887	6888	12/06/22	11:45:00	PM	0.0	22.50	0.0	0.0123
6888	6889	12/07/22	12:00:00	AM	0.0	22.44	0.0	0.0123
6889	6890	12/07/22	12:15:00	AM	0.0	22.51	0.0	0.0123
6890	6891	12/07/22	12:30:00	AM	0.0	22.49	0.0	0.0123
6891	6892	12/07/22	12:45:00	AM	0.0	22.50	0.0	0.0123
6892	6893	12/07/22	01:00:00	AM	0.0	22.59	0.0	0.0123
6893	6894	12/07/22	01:15:00	AM	0.0	22.56	0.0	0.0123
6894	6895	12/07/22	01:30:00	AM	0.0	22.65	0.0	0.0123
6895	6896	12/07/22	01:45:00	AM	0.0	22.67	0.0	0.0123
6896	6897	12/07/22	02:00:00	AM	0.0	22.59	0.0	0.0123
6897	6898	12/07/22	02:15:00	AM	0.0	22.67	0.0	0.0123
6898	6899	12/07/22	02:30:00	AM	0.0	22.66	0.0	0.0123
6899	6900	12/07/22	02:45:00	AM	0.0	22.71	0.0	0.0123
6900	6901	12/07/22	03:00:00	AM	0.0	22.72	0.0	0.0123

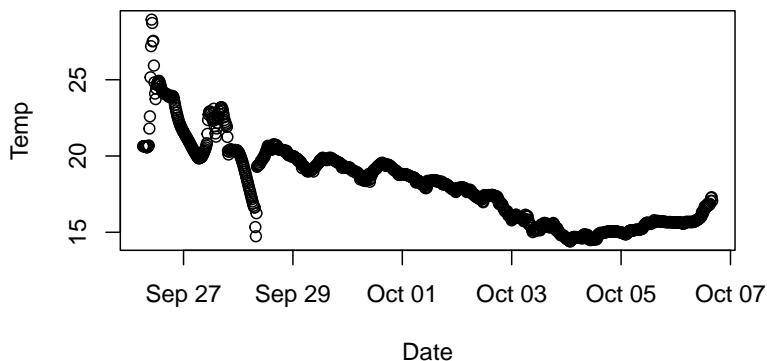
6901	6902	12/07/22	03:15:00	AM	0.0	22.60	0.0	0.0123
6902	6903	12/07/22	03:30:00	AM	0.0	22.68	0.0	0.0123
6903	6904	12/07/22	03:45:00	AM	0.0	22.68	0.0	0.0123
6904	6905	12/07/22	04:00:00	AM	0.0	22.63	0.0	0.0123
6905	6906	12/07/22	04:15:00	AM	0.0	22.74	0.0	0.0123
6906	6907	12/07/22	04:30:00	AM	0.0	22.59	0.0	0.0123
6907	6908	12/07/22	04:45:00	AM	0.0	22.65	0.0	0.0123
6908	6909	12/07/22	05:00:00	AM	0.0	22.71	0.0	0.0123
6909	6910	12/07/22	05:15:00	AM	0.0	22.59	0.0	0.0123
6910	6911	12/07/22	05:30:00	AM	0.0	22.71	0.0	0.0123
6911	6912	12/07/22	05:45:00	AM	0.0	22.62	0.0	0.0123
6912	6913	12/07/22	06:00:00	AM	0.0	22.68	0.0	0.0123
6913	6914	12/07/22	06:15:00	AM	0.0	22.70	0.0	0.0123
6914	6915	12/07/22	06:30:00	AM	0.0	22.57	0.0	0.0123
6915	6916	12/07/22	06:45:00	AM	0.0	22.67	0.0	0.0123
6916	6917	12/07/22	07:00:00	AM	0.0	22.66	0.0	0.0123
6917	6918	12/07/22	07:15:00	AM	0.0	22.68	0.0	0.0123
6918	6919	12/07/22	07:30:00	AM	0.0	22.72	0.0	0.0123
6919	6920	12/07/22	07:45:00	AM	0.0	22.61	0.0	0.0123
6920	6921	12/07/22	08:00:00	AM	0.0	22.70	0.0	0.0123
6921	6922	12/07/22	08:15:00	AM	0.0	22.61	0.0	0.0123
6922	6923	12/07/22	08:30:00	AM	0.0	22.67	0.0	0.0123
6923	6924	12/07/22	08:45:00	AM	0.0	22.77	0.0	0.0123
6924	6925	12/07/22	09:00:00	AM	0.0	22.68	0.0	0.0123
6925	6926	12/07/22	09:15:00	AM	0.0	22.76	0.0	0.0123
6926	6927	12/07/22	09:30:00	AM	0.0	22.61	0.0	0.0123
6927	6928	12/07/22	09:45:00	AM	0.0	22.71	0.0	0.0123
6928	6929	12/07/22	10:00:00	AM	0.0	18.93	0.0	0.0123
6929	6930	12/07/22	10:15:00	AM	0.0	16.36	0.0	0.0123
6930	6931	12/07/22	10:30:00	AM	0.0	15.60	0.0	0.0123
6931	6932	12/07/22	10:45:00	AM	0.0	15.27	0.0	0.0123
6932	6933	12/07/22	11:00:00	AM	0.0	15.12	0.0	0.0123
6933	6934	12/07/22	11:15:00	AM	0.0	15.03	0.0	0.0123
6934	6935	12/07/22	11:30:00	AM	73.4	14.98	115.1	0.0585
6935	6936	12/07/22	11:45:00	AM	90.7	14.96	142.2	0.0708
6936	6937	12/07/22	12:00:00	PM	105.3	14.95	165.2	0.0813
6937	6938	12/07/22	12:15:00	PM	124.7	14.94	195.7	0.0955
6938	6939	12/07/22	12:30:00	PM	138.7	14.90	217.8	0.1059
6939	6940	12/07/22	12:45:00	PM	151.6	14.91	238.0	0.1154
6940	6941	12/07/22	01:00:00	PM	158.0	14.89	248.2	0.1202
6941	6942	12/07/22	01:15:00	PM	164.9	14.86	259.2	0.1254

6942	6943	12/07/22	01:30:00	PM	166.2	14.84	261.4	0.1264
6943	6944	12/07/22	01:45:00	PM	164.9	14.80	259.6	0.1256
6944	6945	12/07/22	02:00:00	PM	164.9	14.76	259.8	0.1257
6945	6946	12/07/22	02:15:00	PM	166.6	14.78	262.4	0.1269
6946	6947	12/07/22	02:30:00	PM	171.4	14.78	269.9	0.1305
6947	6948	12/07/22	02:45:00	PM	176.2	14.75	277.7	0.1342
6948	6949	12/07/22	03:00:00	PM	176.6	14.73	278.5	0.1345
6949	6950	12/07/22	03:15:00	PM	175.7	14.69	277.3	0.1340
6950	6951	12/07/22	03:30:00	PM	174.4	14.65	275.5	0.1331
6951	6952	12/07/22	03:45:00	PM	174.4	14.63	275.7	0.1332
6952	6953	12/07/22	04:00:00	PM	176.2	14.63	278.5	0.1345
6953	6954	12/07/22	04:15:00	PM	179.6	14.63	283.9	0.1371
6954	6955	12/07/22	04:30:00	PM	182.2	14.63	288.0	0.1391
6955	6956	12/07/22	04:45:00	PM	179.2	14.58	283.6	0.1370
6956	6957	12/07/22	05:00:00	PM	181.8	14.60	287.6	0.1389
6957	6958	12/07/22	05:15:00	PM	189.2	14.65	298.9	0.1443
6958	6959	12/07/22	05:30:00	PM	190.5	14.68	300.7	0.1452
6959	6960	12/07/22	05:45:00	PM	197.1	14.67	311.2	0.1502
6960	6961	12/07/22	06:00:00	PM	202.3	14.63	319.8	0.1543
6961	6962	12/07/22	06:15:00	PM	205.4	14.56	325.2	0.1569
6962	6963	12/07/22	06:30:00	PM	205.9	14.47	326.7	0.1576
6963	6964	12/07/22	06:45:00	PM	204.5	14.36	325.3	0.1569
6964	6965	12/07/22	07:00:00	PM	202.3	14.24	322.8	0.1557
6965	6966	12/07/22	07:15:00	PM	200.1	14.11	320.2	0.1545
6966	6967	12/07/22	07:30:00	PM	195.8	14.04	313.9	0.1515
6967	6968	12/07/22	07:45:00	PM	224.0	14.30	356.9	0.1721
6968	6969	12/07/22	08:00:00	PM	223.1	14.30	355.4	0.1714
6969	6970	12/07/22	08:15:00	PM	224.4	14.30	357.5	0.1724
6970	6971	12/07/22	08:30:00	PM	227.1	14.30	361.8	0.1745
6971	6972	12/07/22	08:45:00	PM	229.3	14.31	365.2	0.1762
6972	6973	12/07/22	09:00:00	PM	231.5	14.30	368.8	0.1779
6973	6974	12/07/22	09:15:00	PM	232.8	14.29	371.0	0.1789
6974	6975	12/07/22	09:30:00	PM	232.8	14.26	371.2	0.1791
6975	6976	12/07/22	09:45:00	PM	233.3	14.22	372.4	0.1796
6976	6977	12/07/22	10:00:00	PM	232.8	14.20	371.8	0.1793
6977	6978	12/07/22	10:15:00	PM	231.5	14.16	370.1	0.1785
6978	6979	12/07/22	10:30:00	PM	227.9	14.08	365.0	0.1761
6979	6980	12/07/22	10:45:00	PM	224.0	13.99	359.5	0.1734
6980	6981	12/07/22	11:00:00	PM	220.0	13.88	354.1	0.1708
6981	6982	12/07/22	11:15:00	PM	215.5	13.73	348.1	0.1679
6982	6983	12/07/22	11:30:00	PM	212.0	13.57	343.8	0.1658

6983	6984	12/07/22	11:45:00	PM	209.8	13.44	341.3	0.1646
6984	6985	12/08/22	12:00:00	AM	210.7	13.37	343.3	0.1656
6985	6986	12/08/22	12:15:00	AM	211.1	13.32	344.4	0.1661
6986	6987	12/08/22	12:30:00	AM	210.7	13.26	344.2	0.1660
6987	6988	12/08/22	12:45:00	AM	210.3	13.22	343.9	0.1659
6988	6989	12/08/22	01:00:00	AM	209.8	13.17	343.5	0.1657
6989	6990	12/08/22	01:15:00	AM	208.5	13.10	342.0	0.1649
6990	6991	12/08/22	01:30:00	AM	205.9	12.96	338.9	0.1634
6991	6992	12/08/22	01:45:00	AM	203.2	12.85	335.3	0.1617
6992	6993	12/08/22	02:00:00	AM	205.4	12.78	339.5	0.1638
6993	6994	12/08/22	02:15:00	AM	212.9	12.78	351.9	0.1697
6994	6995	12/08/22	02:30:00	AM	220.9	12.77	365.3	0.1762
6995	6996	12/08/22	02:45:00	AM	224.8	12.77	371.7	0.1793
6996	6997	12/08/22	03:00:00	AM	226.2	12.73	374.4	0.1806
6997	6998	12/08/22	03:15:00	AM	224.8	12.68	372.5	0.1797
6998	6999	12/08/22	03:30:00	AM	222.2	12.61	368.9	0.1779
6999	7000	12/08/22	03:45:00	AM	217.3	12.46	362.1	0.1746
7000	7001	12/08/22	04:00:00	AM	216.0	12.30	361.3	0.1743
7001	7002	12/08/22	04:15:00	AM	217.8	12.14	365.8	0.1764
7002	7003	12/08/22	04:30:00	AM	218.2	12.02	367.6	0.1773
7003	7004	12/08/22	04:45:00	AM	215.1	11.85	363.9	0.1755
7004	7005	12/08/22	05:00:00	AM	212.9	11.67	361.8	0.1745
7005	7006	12/08/22	05:15:00	AM	212.0	11.49	361.9	0.1745
7006	7007	12/08/22	05:30:00	AM	211.6	11.32	362.8	0.1750
7007	7008	12/08/22	05:45:00	AM	207.6	11.12	357.7	0.1725
7008	7009	12/08/22	06:00:00	AM	206.7	10.91	358.1	0.1727
7009	7010	12/08/22	06:15:00	AM	204.1	10.71	355.4	0.1714
7010	7011	12/08/22	06:30:00	AM	201.5	10.50	352.7	0.1701
7011	7012	12/08/22	06:45:00	AM	199.7	10.29	351.5	0.1695
7012	7013	12/08/22	07:00:00	AM	199.3	10.09	352.6	0.1700
7013	7014	12/08/22	07:15:00	AM	197.1	9.91	350.3	0.1690
7014	7015	12/08/22	07:30:00	AM	194.9	9.70	348.3	0.1680
7015	7016	12/08/22	07:45:00	AM	194.9	9.49	350.2	0.1689
7016	7017	12/08/22	08:00:00	AM	194.0	9.33	350.0	0.1688
7017	7018	12/08/22	08:15:00	AM	193.6	9.15	351.0	0.1693
7018	7019	12/08/22	08:30:00	AM	194.0	9.00	353.1	0.1703
7019	7020	12/08/22	08:45:00	AM	195.8	8.89	357.4	0.1724
7020	7021	12/08/22	09:00:00	AM	199.3	8.84	364.2	0.1757
7021	7022	12/08/22	09:15:00	AM	205.9	8.87	376.0	0.1814
7022	7023	12/08/22	09:30:00	AM	210.3	8.90	383.7	0.1851
7023	7024	12/08/22	09:45:00	AM	215.1	8.99	391.6	0.1889

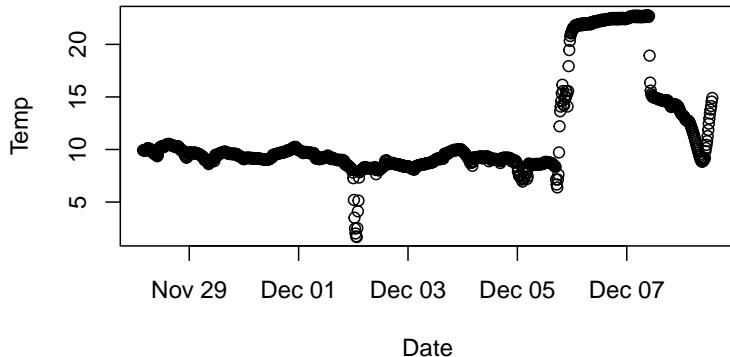
7024	7025	12/08/22	10:00:00 AM	216.9	9.08	393.9	0.1901
7025	7026	12/08/22	10:15:00 AM	217.3	9.21	393.3	0.1898
7026	7027	12/08/22	10:30:00 AM	219.1	9.84	390.1	0.1882
7027	7028	12/08/22	10:45:00 AM	219.5	10.14	387.8	0.1871
7028	7029	12/08/22	11:00:00 AM	222.2	10.44	389.6	0.1879
7029	7030	12/08/22	11:15:00 AM	226.2	10.80	392.9	0.1896
7030	7031	12/08/22	11:30:00 AM	231.9	11.28	398.0	0.1920
7031	7032	12/08/22	11:45:00 AM	238.6	11.86	403.5	0.1947
7032	7033	12/08/22	12:00:00 PM	246.7	12.46	411.1	0.1984
7033	7034	12/08/22	12:15:00 PM	252.9	12.95	416.3	0.2009
7034	7035	12/08/22	12:30:00 PM	258.8	13.42	421.2	0.2033
7035	7036	12/08/22	12:45:00 PM	262.8	13.80	423.7	0.2046
7036	7037	12/08/22	01:00:00 PM	234.2	14.13	374.6	0.1807
7037	7038	12/08/22	01:15:00 PM	250.2	14.56	396.1	0.1911
7038	7039	12/08/22	01:30:00 PM	261.0	14.90	409.9	0.1978

```
Eel_cond$Date <- as.POSIXct(strptime(Eel_cond$Date,
                                     format = "%m/%d/%y %I:%M:%S %p", tz = "EST"))
#
#
h <- head(Eel_cond, 1000)
plot(data = h, Temp~Date)
```



```
Eel_cond <- Eel_cond[Eel_cond$Date > "2022-09-29",]
```

```
t <- tail(Eel_cond, 1000)
plot(data = t, Temp~Date)
```



```
Eel_cond <- Eel_cond[Eel_cond$Date < "2022-12-05 12:00:00",]

gg3 <- gg2+
  geom_point(data = Eel_cond, aes(x=Date,y=Temp), color = "purple", cex = 1)
gg3
```

Warning: Removed 64 rows containing missing values (`geom_point()`).

Warning: Removed 62 rows containing missing values (`geom_line()`).

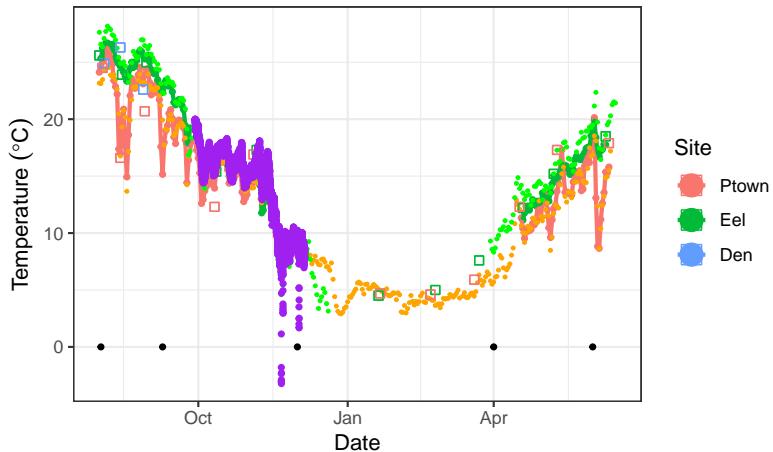
Warning: Removed 1 rows containing missing values (`geom_point()`).

Warning: Removed 284 rows containing missing values (`geom_point()`).

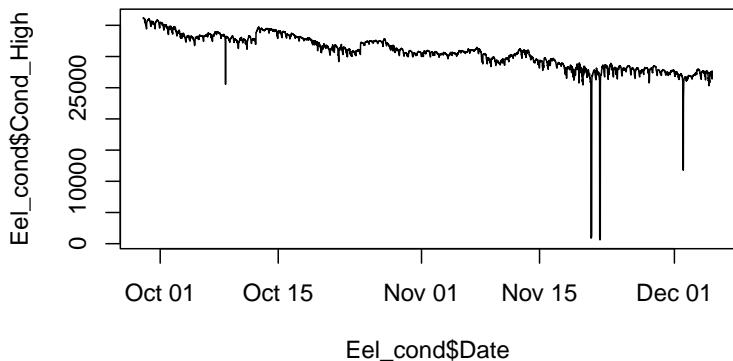
Warning: Removed 361 rows containing missing values (`geom_point()`).

Warning: Removed 24 rows containing missing values (`geom_point()`).

A. Temp



```
plot(Eel_cond$Date, Eel_cond$Cond_High, type = "l")
```

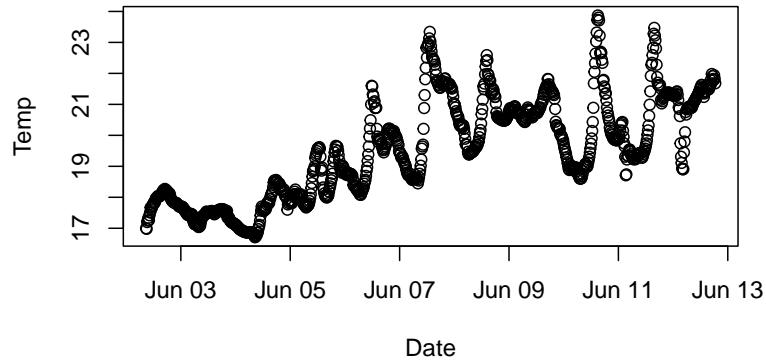


```
data_dir <- "~/GitHub/EAD-ASEB-Ssolidissima-0A/projects/Seawater data/data/Conductivity/"
Ptown_cond <- read.csv(paste(data_dir,"20674247_Ptown.csv",sep=""), skip = 2, stringsAsFactors = FALSE)
names(Ptown_cond) <- c("n","Date","Cond_High","Temp","Other")
str(Ptown_cond)
```

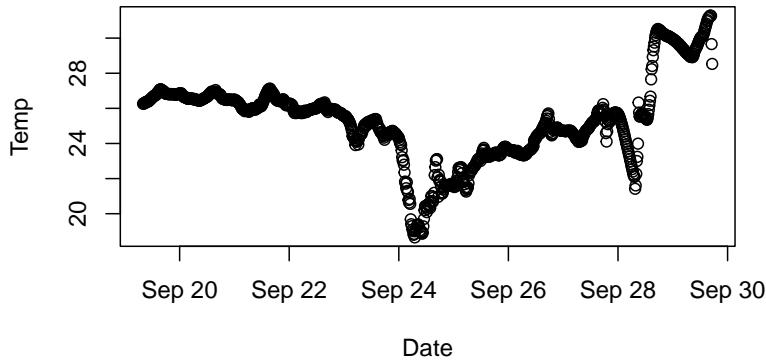
```
'data.frame': 11461 obs. of 9 variables:
 $ n       : int 4 5 6 7 8 9 10 11 12 13 ...
 $ Date    : chr "6/2/2022 8:45" "6/2/2022 9:00" "6/2/2022 9:15" "6/2/2022 9:30" ...
```

```
$ Cond_High: num  0 32496 32904 32947 32965 ...
$ Temp      : num  17 17 17.2 17.2 17.3 ...
$ Other     : logi  NA NA NA NA NA NA ...
$ NA        : chr  "" "" "" ...
```

```
Ptown_cond$Date <- as.POSIXct(Ptown_cond$Date, format = "%m/%d/%Y %H:%M")
#
#
h <- head(Ptown_cond,1000)
plot(data = h,Temp~Date)
```



```
#Ptown_cond <- Ptown_cond[Ptown_cond$Date > "2022-09-29",]
t <- tail(Ptown_cond,1000)
plot(data = t,Temp~Date)
```



```
Ptown_cond <- Ptown_cond[Ptown_cond$Date < "2022-09-27 17:00:00",]
#
str(Ptown_cond)

'data.frame': 11265 obs. of 9 variables:
 $ n      : int  4 5 6 7 8 9 10 11 12 13 ...
 $ Date    : POSIXct, format: "2022-06-02 08:45:00" "2022-06-02 09:00:00" ...
 $ Cond_High: num  0 32496 32904 32947 32965 ...
 $ Temp    : num  17 17 17.2 17.2 17.3 ...
 $ Other   : logi  NA NA NA NA NA NA ...
 $ NA      : chr  "" "" "" ...
```

```
Ptown.1 <- Ptown_cond[,c("Date","Cond_High","Temp")]

gg3 <- gg2+
  geom_point(data = Ptown_cond, aes(x=Date,y=Temp), color = "purple", cex = 1)
gg3
```

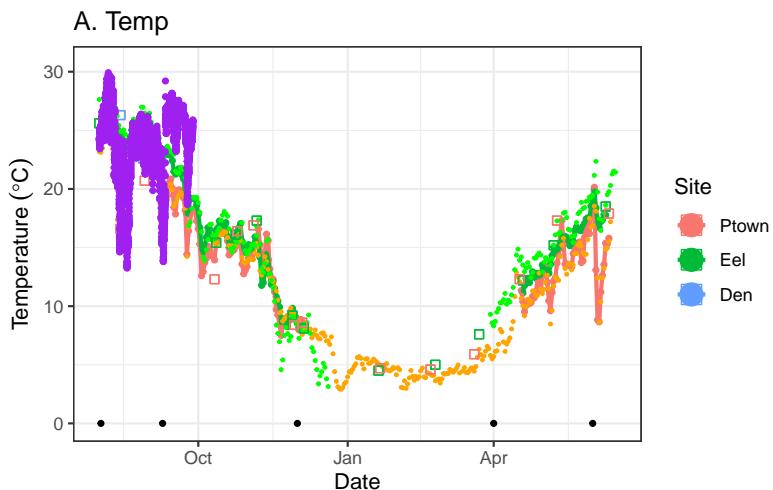
Warning: Removed 64 rows containing missing values (`geom_point()`).

Warning: Removed 62 rows containing missing values (`geom_line()`).

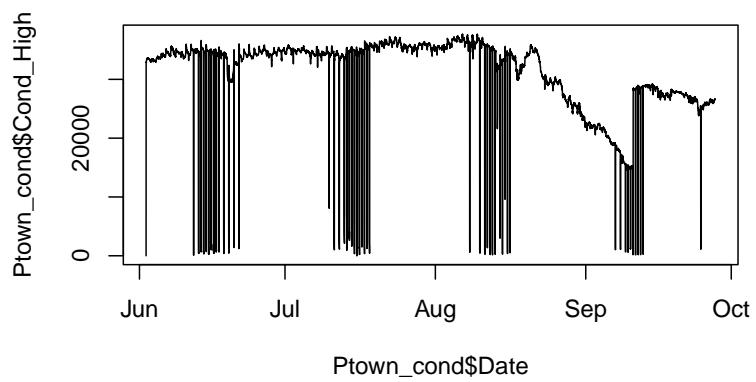
```

Warning: Removed 1 rows containing missing values (`geom_point()`).
Warning: Removed 284 rows containing missing values (`geom_point()`).
Warning: Removed 361 rows containing missing values (`geom_point()`).
Warning: Removed 24 rows containing missing values (`geom_point()`).
Warning: Removed 5725 rows containing missing values (`geom_point()`).

```



```
plot(Ptown_cond$Date, Ptown_cond$Cond_High, type = "l")
```



```

data_dir <- "~/GitHub/EAD-ASEB-Ssolidissima-0A/projects/Seawater data/data/Conductivity/"
Ptown_cond <- read.csv(paste(data_dir,"21446049_Ptown_Dec22.csv",sep=""), skip = 2, stringsAsFactors=TRUE)
names(Ptown_cond) <- c("n","Date","Cond_Low","Cond_High","Temp","Other")
str(Ptown_cond)

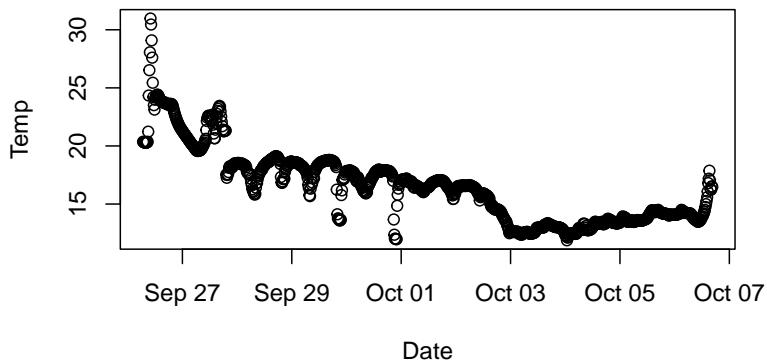
'data.frame': 7040 obs. of 10 variables:
$ n      : int 4 5 6 7 8 9 10 11 12 13 ...
$ Date   : chr "9/26/2022 6:45" "9/26/2022 7:00" "9/26/2022 7:15" "9/26/2022 7:30" ...
$ Cond_Low: num 0 0 0 0 0 0 0 0 0 0 ...
$ Cond_High: num 0 0 0 0 0 0 0 0 0 0 ...
$ Temp   : num 20.4 20.3 20.3 20.3 20.3 ...
$ Other   : chr "" "" "" ...
$ NA     : chr ...

```

```

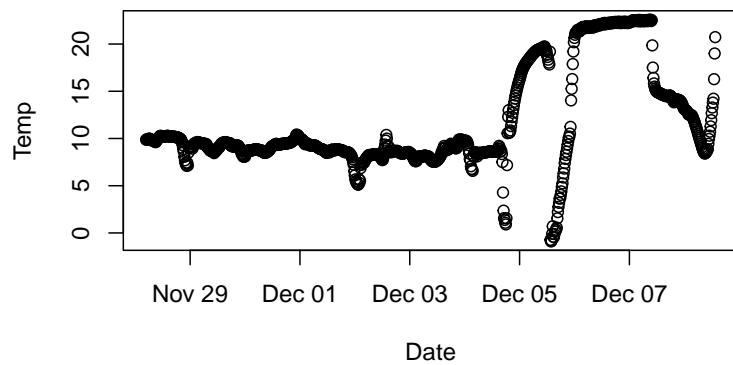
Ptown_cond$Date <- as.POSIXct(Ptown_cond$Date, format = "%m/%d/%Y %H:%M")
#
#
h <- head(Ptown_cond,1000)
plot(data = h,Temp~Date)

```



```
Ptown_cond <- Ptown_cond[Ptown_cond$Date > "2022-09-27 20:00:00",]

t <- tail(Ptown_cond,1000)
plot(data = t,Temp~Date)
```



```
Ptown_cond <- Ptown_cond[Ptown_cond$Date < "2022-12-04 13:00:00",]

#
str(Ptown_cond)
```

```
'data.frame': 6499 obs. of 10 variables:
 $ n      : int  154 155 156 157 158 159 160 161 162 163 ...
 $ Date   : POSIXct, format: "2022-09-27 20:15:00" "2022-09-27 20:30:00" ...
 $ Cond_Low: num  28182 28357 28409 28265 28159 ...
 $ Cond_High: num  38034 38333 38336 38165 38040 ...
 $ Temp   : num  17.8 18.1 18.2 18.2 18.2 ...
 $ Other   : chr  "" "" "" ...
 $ NA      : chr  "" "" ...
```

```
Ptown.2 <- Ptown_cond[,c("Date","Cond_High","Temp")]

gg3 <- gg2+
  geom_point(data = Ptown_cond, aes(x=Date,y=Temp), color = "purple", cex = 1)
gg3
```

Warning: Removed 64 rows containing missing values (`geom_point()`).

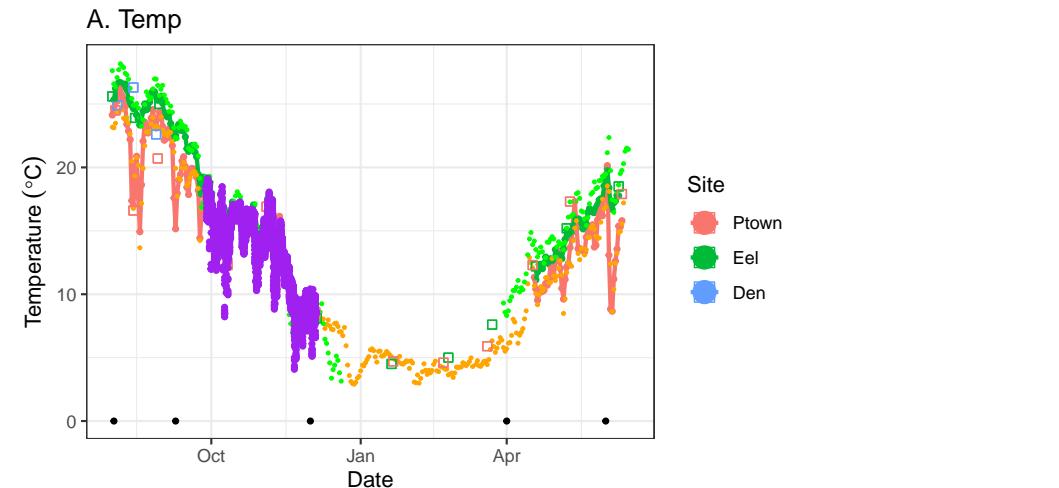
Warning: Removed 62 rows containing missing values (`geom_line()`).

Warning: Removed 1 rows containing missing values (`geom_point()`).

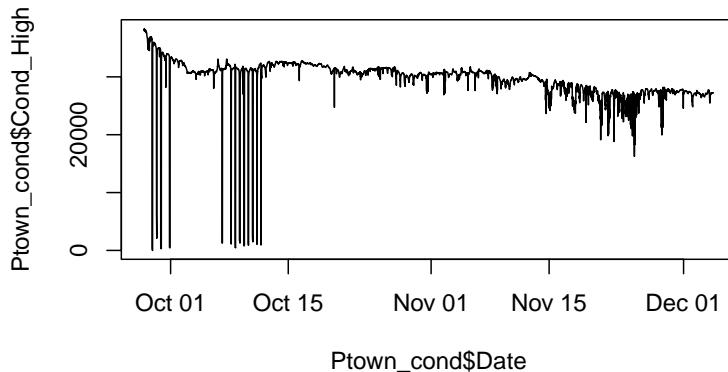
Warning: Removed 284 rows containing missing values (`geom_point()`).

Warning: Removed 361 rows containing missing values (`geom_point()`).

Warning: Removed 24 rows containing missing values (`geom_point()`).



```
plot(Ptown_cond$Date, Ptown_cond$Cond_High, type = "l")
```



```

data_dir <- "~/GitHub/EAD-ASEB-Ssolidissima-0A/projects/Seawater data/data/Conductivity/"
Ptown_cond <- read.csv(paste(data_dir,"21446049_Ptown_Jul23.csv",sep=""), skip = 2, stringsAsFactors=TRUE)
names(Ptown_cond) <- c("n","Date","Cond_Low","Cond_High","Temp","Other")
str(Ptown_cond)

```

```

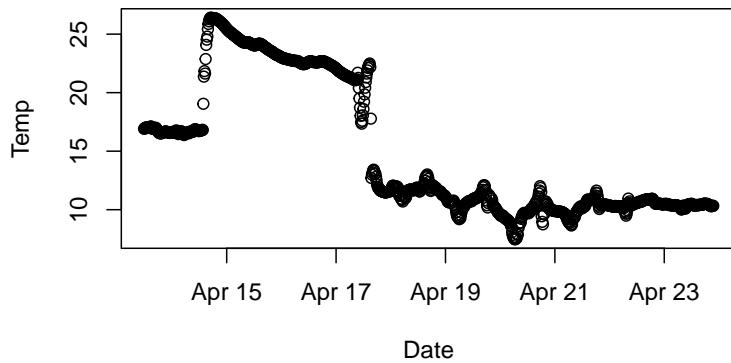
'data.frame':   10480 obs. of  10 variables:
 $ n      : int  4 5 6 7 8 9 10 11 12 13 ...
 $ Date   : chr  "4/13/2023 11:45" "4/13/2023 12:00" "4/13/2023 12:15" "4/13/2023 12:30" ...
 $ Cond_Low: num  18392 18491 18481 18548 18540 ...
 $ Cond_High: num  26717 26735 26754 26776 26776 ...
 $ Temp    : num  16.9 16.9 17 17 17 ...
 $ Other   : chr  "" "" "" "" ...
 $ NA      : chr  "" "" "" "" ...
 $ NA      : chr  "" "" "" "" ...
 $ NA      : chr  "" "" "" "" ...
 $ NA      : chr  "" "" "" "" ...

```

```

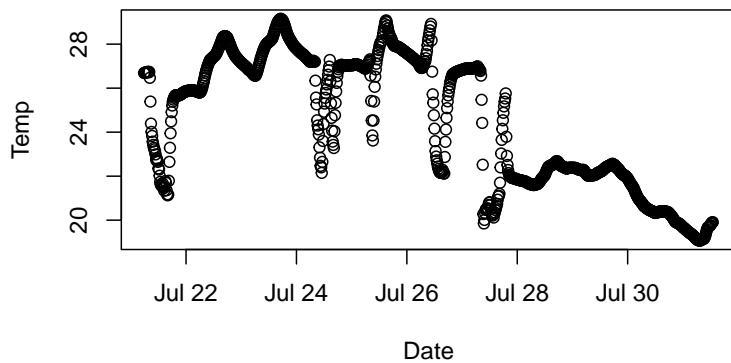
Ptown_cond$date <- as.POSIXct(Ptown_cond$date, format = "%m/%d/%Y %H:%M")
#
#
h <- head(Ptown_cond,1000)
plot(data = h,Temp~Date)

```



```
Ptown_cond <- Ptown_cond[Ptown_cond$Date > "2023-04-17 15:00:00",]

t <- tail(Ptown_cond, 1000)
plot(data = t, Temp~Date)
```



```
Ptown_cond <- Ptown_cond[Ptown_cond$Date < "2023-07-19 12:00:00",]

#
str(Ptown_cond)
```

```

'data.frame': 8915 obs. of 10 variables:
$ n      : int 402 403 404 405 406 407 408 409 410 411 ...
$ Date   : POSIXct, format: "2023-04-17 15:15:00" "2023-04-17 15:30:00" ...
$ Cond_Low : num 96.5 13103.5 12916.4 12072.3 12203.6 ...
$ Cond_High: num 145 15426 15276 14200 14352 ...
$ Temp    : num 17.8 12.7 13 13.2 13.4 ...
$ Other   : chr "" "" "" ...
$ NA      : chr "" "" "" ...

```

```

Ptown.3 <- Ptown_cond[,c("Date","Cond_High","Temp")]

Ptown_cond <- rbind(Ptown.1, Ptown.2, Ptown.3)

gg3 <- gg2+
  geom_point(data = Ptown_cond, aes(x=Date,y=Temp), color = "purple", cex = 1)
gg3

```

Warning: Removed 64 rows containing missing values (`geom_point()`).

Warning: Removed 62 rows containing missing values (`geom_line()`).

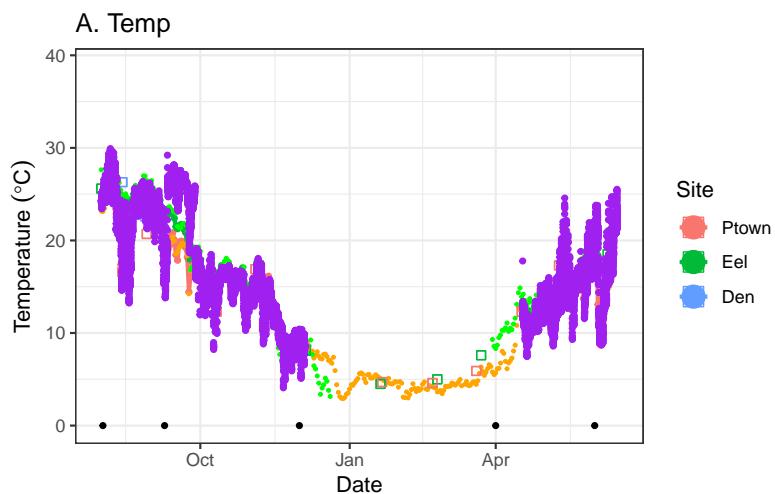
Warning: Removed 1 rows containing missing values (`geom_point()`).

Warning: Removed 284 rows containing missing values (`geom_point()`).

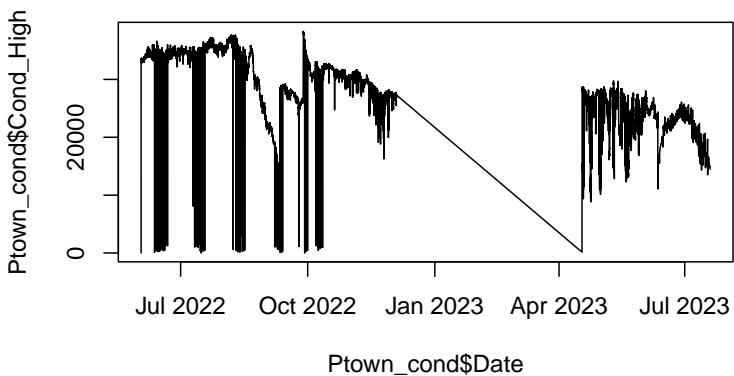
Warning: Removed 361 rows containing missing values (`geom_point()`).

Warning: Removed 24 rows containing missing values (`geom_point()`).

Warning: Removed 9036 rows containing missing values (`geom_point()`).



```
plot(Ptown_cond$Date, Ptown_cond$Cond_High, type = "l")
```



SW temperature

```
data_dir <- "~/GitHub/EAD-ASEB-Ssolidissima-0A/projects/Seawater data/data/SW temp HOBO/"
Den_cond <- read.csv(paste(data_dir,"10709184_East_Dennis.csv",sep=""), skip = 2, stringsAsFactors = TRUE)
names(Den_cond) <- c("n","Date","Temp","Other")
str(Den_cond)
```

```
'data.frame': 10549 obs. of 8 variables:
```

```

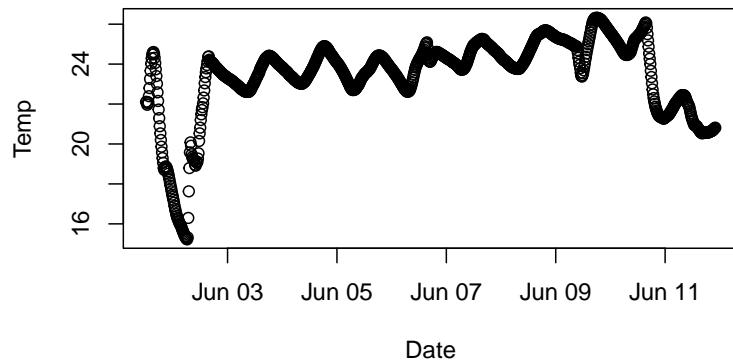
$ n      : int  2 3 4 5 6 7 8 9 10 11 ...
$ Date   : chr  "6/1/2022 12:15" "6/1/2022 12:30" "6/1/2022 12:45" "6/1/2022 13:00" ...
$ Temp    : num  22.1 22.1 22 22 22.1 ...
$ Other   : logi  NA NA NA NA NA NA ...
$ NA      : chr  "" "" "" "" ...
$ NA      : chr  "" "" "" "" ...
$ NA      : chr  "" "" "" "" ...
$ NA      : chr  "" "" "" "" ...

```

```

Den_cond$Date <- as.POSIXct(Den_cond$Date, format = "%m/%d/%Y %H:%M")
#
#
h <- head(Den_cond,1000)
plot(data = h,Temp~Date)

```

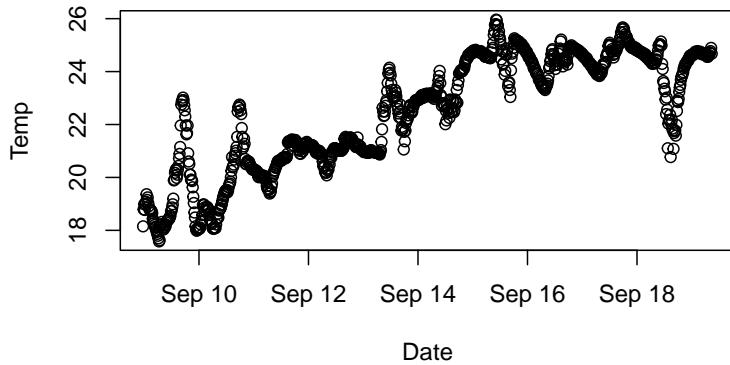


```

Den_cond <- Den_cond[Den_cond$Date > "2022-06-17 12:00:00",]

t <- tail(Den_cond,1000)
plot(data = t,Temp~Date)

```



```

Den_cond <- Den_cond[Den_cond$Date < "2023-09-12",]

#
str(Den_cond)

'data.frame': 9013 obs. of 8 variables:
 $ n      : int 1538 1539 1540 1541 1542 1543 1544 1545 1546 1547 ...
 $ Date   : POSIXct, format: "2022-06-17 12:15:00" "2022-06-17 12:30:00" ...
 $ Temp   : num 25.4 25.6 25.6 25.7 25.9 ...
 $ Other   : logi NA NA NA NA NA NA ...
 $ NA     : chr "" "" "" ...
gg3 <- gg2+
  geom_point(data = Den_cond, aes(x=Date,y=Temp), color = "purple", cex = 1)
gg3

```

Warning: Removed 64 rows containing missing values (`geom_point()`).

Warning: Removed 62 rows containing missing values (`geom_line()`).

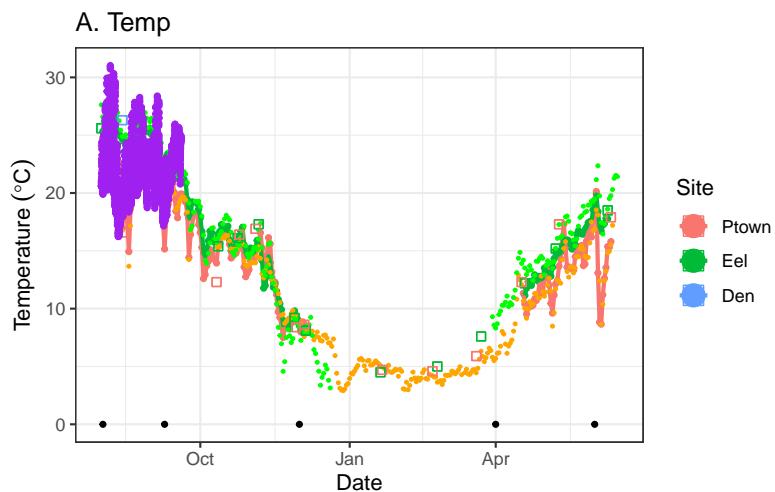
Warning: Removed 1 rows containing missing values (`geom_point()`).

Warning: Removed 284 rows containing missing values (`geom_point()`).

Warning: Removed 361 rows containing missing values (`geom_point()`).

Warning: Removed 24 rows containing missing values (`geom_point()`).

Warning: Removed 4273 rows containing missing values (`geom_point()`).



0.7 Flood and ebb tide analysis

Issues: The tide groups aren't right. They jump around. Also not all the sediment temps get assigned a tide group because I need to assign everything to the same timeseries (0,15,30,45) etc.

```
library(lemon)
```

Warning: package 'lemon' was built under R version 4.3.2

```
library(zoo)
```

Warning: package 'zoo' was built under R version 4.3.2

Attaching package: 'zoo'

The following objects are masked from 'package:base':

```
as.Date, as.Date.numeric
```

```
library(lubridate)
```

```
Ptown_sediment <- Ptown_sediment[,c("Date","Temp")]
```

```
# to.minutes15(Ptown)[,4] # interpolate for each 15 min interval
```

```
# format(round_date(as.POSIXct(paste("1900-01-01 ", x)), unit="15 mins"), "%M")  
str(Ptown)
```

'data.frame': 16534 obs. of 17 variables:

```
$ Site          : Factor w/ 3 levels "Den","Eel","Ptown": NA  
$ Date          : POSIXct, format: NA NA ...  
$ Temp          : num  NA NA NA NA NA NA NA NA NA ...  
$ pH            : num  NA NA NA NA NA NA NA NA NA ...  
$ Tris          : num  NA NA NA NA NA NA NA NA NA ...  
$ Tris_temp     : num  NA NA NA NA NA NA NA NA NA ...  
$ Serial_number: num  NA NA NA NA NA NA NA NA NA ...  
$ offset        : num  NA NA NA NA NA NA NA NA NA ...  
$ pH_corr       : num  NA NA NA NA NA NA NA NA NA ...  
$ H_plus_conc  : num  NA NA NA NA NA NA NA NA NA ...  
$ date          : chr  NA NA NA NA ...  
$ day           : chr  NA NA NA NA ...  
$ Time          : chr  NA NA NA NA ...  
$ Tideheight    : num  NA NA NA NA NA NA NA NA NA ...  
$ state         : chr  NA NA NA NA ...  
$ previous_Tideheight: num  NA NA NA NA NA NA NA NA NA ...  
$ tide_group    : int  NA NA NA NA NA NA NA NA NA ...
```

```
Ptown$Date <- round_date(Ptown$Date, unit="15 mins")  
tail(Eel$Date)
```

[1] "2023-06-09 07:45:00 -05" "2023-06-09 08:00:00 -05"

[3] "2023-06-09 08:15:00 -05" "2023-06-09 08:30:00 -05"

[5] "2023-06-09 08:45:00 -05" "2023-06-09 09:00:00 -05"

```
Ptown.sed <- merge(Ptown, Ptown_sediment, by = "Date",  
                     all = TRUE ,  
                     suffixes = c("", ".sediment"))
```

```
Ptown.sed$Season <- cut(lubridate::yday(Ptown.sed$Date - lubridate::days(79)),  
                        breaks = c(0, 93, 187, 276, Inf),  
                        labels = c("Spring", "Summer", "Autumn", "Winter"))
```

```
Ptown.sed <- Ptown.sed[!is.na(Ptown.sed$season),]
```

```
Ptown.sed <- Ptown.sed[Ptown.sed$season!="Winter",]
```

```
head(Ptown_sediment$Date) #The sediment logger is on the 15 minute interval.
```

```

lim.P <- as.POSIXct(c(min(Eel.sed$Date),max(Ptown.sed$Date)))
lim.E <- as.POSIXct(c(min(Eel.sed$Date),max(Ptown.sed$Date)))
carb_dat <- carb_dat[carb_dat$Date>min(Eel.sed$Date) & carb_dat$Date<max(Ptown.sed$Date),]

carb_dat$season <- cut(lubridate::yday(carb_dat$Date - lubridate::days(79)),
  breaks = c(0, 93, 187, 276, Inf),
  labels = c("Spring", "Summer", "Autumn", "Winter"))
carb_dat$season <- factor(carb_dat$season, levels=c("Autumn", "Winter", "Spring", "Summer"))
Ptown.sed$season <- factor(Ptown.sed$season, levels=c("Autumn", "Winter", "Spring", "Summer"))
Eel.sed$season <- factor(Eel.sed$season, levels=c("Autumn", "Winter", "Spring", "Summer"))

carb_dat.P <- carb_dat[carb_dat$Stratum=="Bottom water"&carb_dat$Site=="Ptown",]
carb_dat.E <- carb_dat[carb_dat$Stratum=="Bottom water"&carb_dat$Site=="Eel",]

ggplot(Ptown.sed, aes(y = pH, x = Date))+  

  geom_line(aes(group = tide_group,  

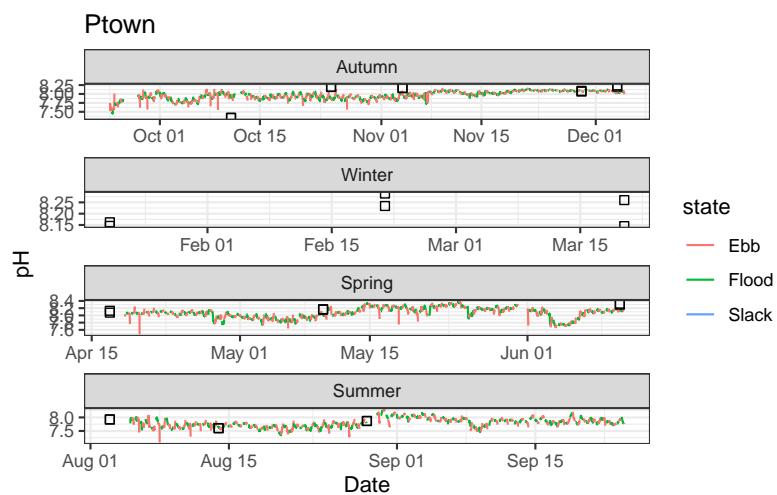
                 color = state))+  

  ggtitle("Ptown")+
  #facet_wrap(vars(season), scales = "free", nrow = 4)+  

  facet_rep_wrap(~season, scales="free", nrow = 4)+  

  geom_point(data = carb_dat.P, aes(x=Date, y = InSitu.pH),cex = 2, pch=22)

```



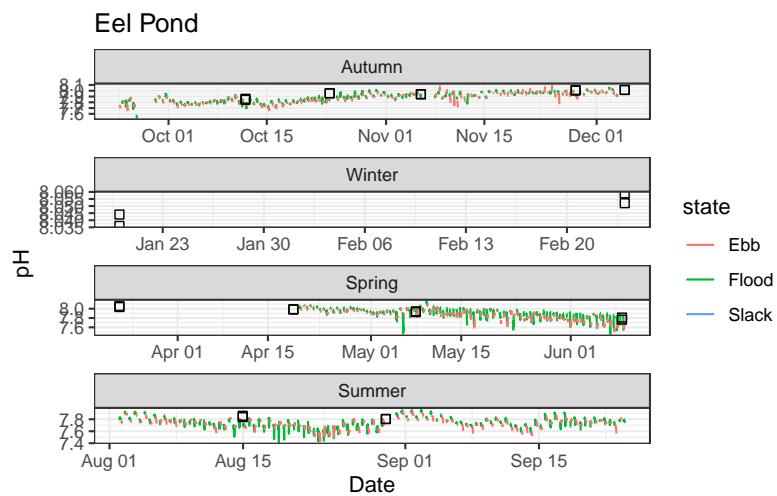
```

ggplot(Eel.sed, aes(y = pH, x = Date))+
  geom_line(aes(group = tide_group, color = state))+  

  ggtitle("Eel Pond")+
  facet_wrap(vars(season), scales = "free", nrow = 4)+  

  geom_point(data = carb_dat.E, aes(x=Date, y = InSitu.pH),cex = 2, pch=22)

```



```

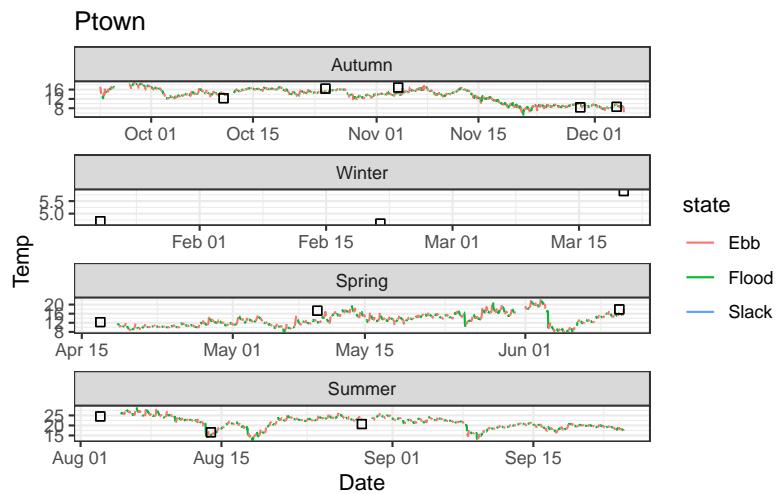
ggplot(Ptown.sed, aes(y = Temp, x = Date))+
  geom_line(aes(group = tide_group,
                color = state))+  

  ggtitle("Ptown")+
  #facet_wrap(vars(season), scales = "free", nrow = 4)+  

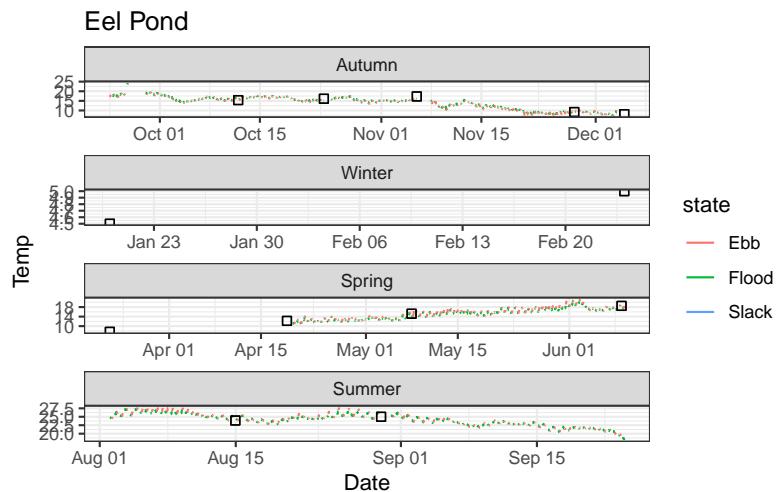
  facet_wrap(~season, scales="free", nrow = 4)+  

  geom_point(data = carb_dat.P, aes(x=Date, y = Water.Sediment..Temp),cex = 2, pch=22)

```

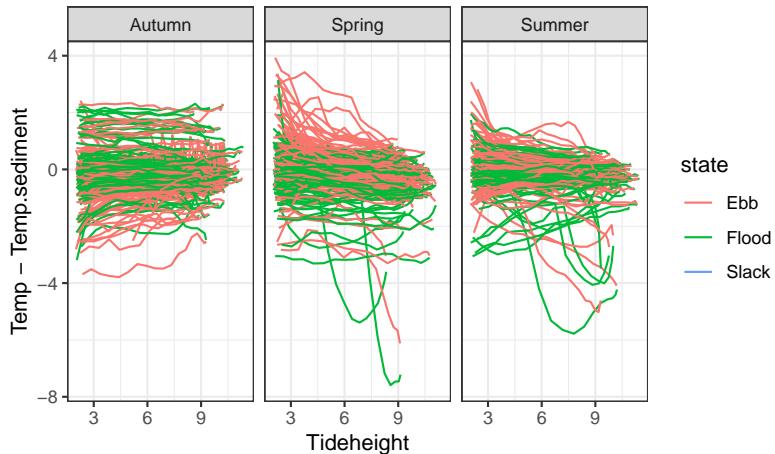


```
ggplot(Eel.sed, aes(y = Temp, x = Date))+
  geom_line(aes(group = tide_group, color = state))+
  ggtitle("Eel Pond")+
  facet_wrap(vars(season), scales = "free", nrow = 4)+
  geom_point(data = carb_dat.E, aes(x=Date, y = Water.Sediment..Temp), cex = 2, pch=22)
```



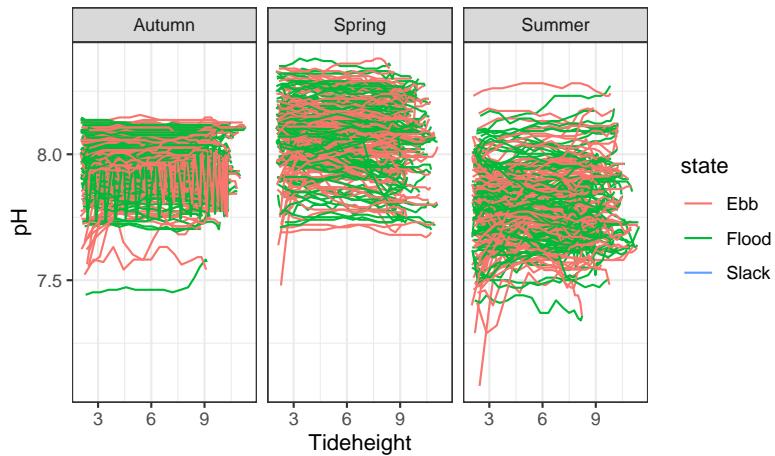
```
ggplot(Ptown.sed, aes(y = Temp-Temp.sediment, x = Tideheight, group = tide_group, color = state))+
  geom_line()+
  facet_wrap(~season)+ ggtitle("Provincetown")
```

Provincetown

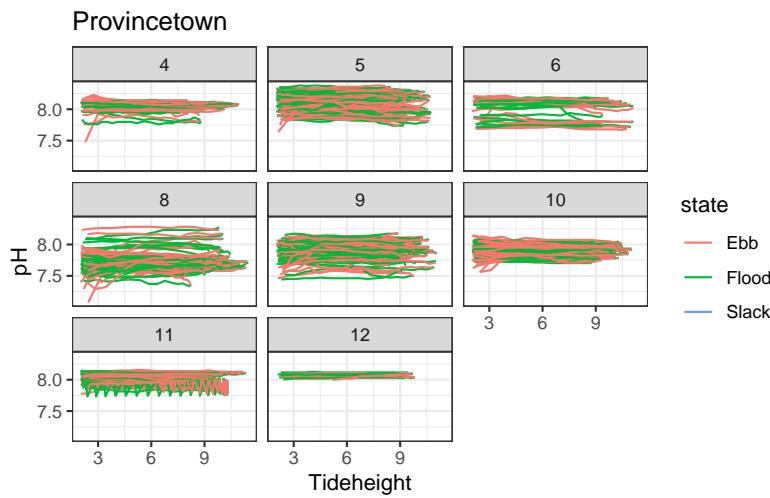


```
ggplot(Ptown.sed, aes(y = pH, x = Tideheight, group = tide_group, color = state))+
  geom_line()+
  facet_wrap(~season)+  ggttitle("Provincetown")
```

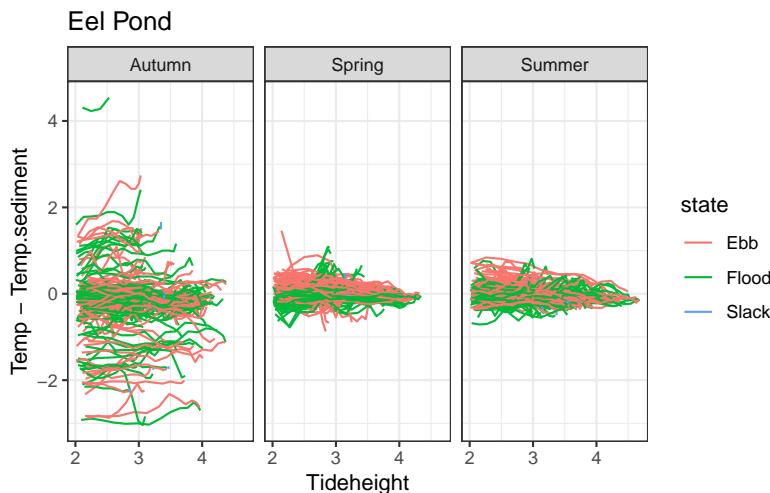
Provincetown



```
ggplot(Ptown.sed, aes(y = pH, x = Tideheight, group = tide_group, color = state))+
  geom_line()+
  facet_wrap(~month(Date))+  ggttitle("Provincetown")
```

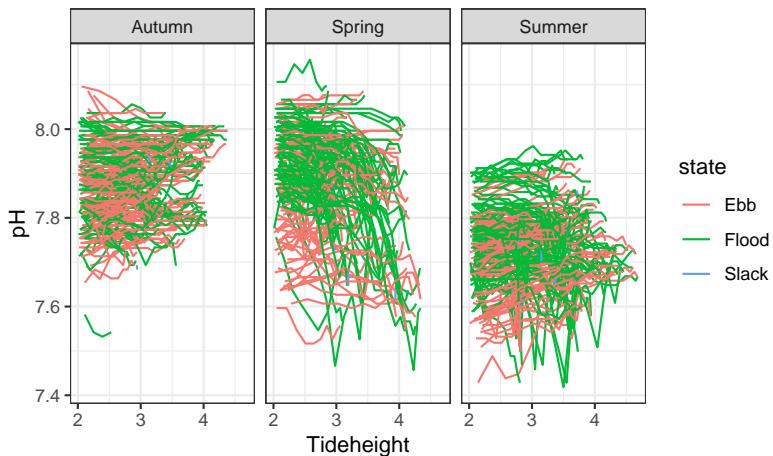


```
Eel.sed$temp.diff <- Eel.sed$Temp-Eel.sed$Temp.sediment
ggplot(Eel.sed, aes(y = Temp-Temp.sediment, x = Tideheight, group = tide_group, color = state))+
  geom_line()+
  facet_wrap(~season)+  ggttitle("Eel Pond")
```



```
ggplot(Eel.sed, aes(y = pH, x = Tideheight, group = tide_group, color = state))+ 
  geom_line()+
  facet_wrap(~season) +  ggttitle("Eel Pond")
```

Eel Pond



```
# result.P <- Ptown.sed %>%
#   group_by(tide_group) %>%
#   reframe(#initial_flood = first(Tideheight[state == "Flood"]),
#          #final_ebb = last(Tideheight[state == "Ebb"]),
#          final_ebb_pH = pH[which(Tideheight ==
#                               last(Tideheight[state == "Ebb"]))],
#          initial_flood_pH = pH[which(Tideheight ==
#                               first(Tideheight[state == "Flood"]))],
#          sediment_temp = mean(Temp.sediment, na.rm = TRUE),
#          #difference = last(Tideheight[state == "Ebb"]) - first(Tideheight[state == "Flood"])
#          )
#
# result.E <- Eel.sed %>%
#   group_by(tide_group) %>%
#   reframe(#initial_flood = first(Tideheight[state == "Flood"]),
#          #final_ebb = last(Tideheight[state == "Ebb"]),
#          final_ebb_pH = pH[which(Tideheight ==
#                               last(Tideheight[state == "Ebb"]))],
#          initial_flood_pH = pH[which(Tideheight ==
#                               first(Tideheight[state == "Flood"]))],
#          sediment_temp = mean(Temp.sediment, na.rm = TRUE),
#          difference = last(Tideheight[state == "Ebb"]) - first(Tideheight[state == "Flood"])
#          )
#
# #
#
# ggplot(result, aes(y = final_ebb_pH, x = initial_flood_pH)) +
```

```
#   geom_point()+
#   ylim(7.5,8.5)+
#   xlim(7.5,8.5)
#
#
# result$change <- result$initial_flood_pH-result$final_ebb_pH
# ggplot(result, aes(y = change, x = sediment_temp))+
#   geom_point()
# #+
# # ylim(7.5,8.5) +
# # xlim(7.5,8.5)
```