2022\_Rating\_Curve

Jake Cavaiani

9/2/2022

# Load packages #  
library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 ──

## ✓ ggplot2 3.3.5 ✓ purrr 0.3.4  
## ✓ tibble 3.1.6 ✓ dplyr 1.0.7  
## ✓ tidyr 1.1.3 ✓ stringr 1.4.0  
## ✓ readr 1.4.0 ✓ forcats 0.5.1

## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(lubridate)

##   
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':  
##   
## date, intersect, setdiff, union

library(data.table)

##   
## Attaching package: 'data.table'

## The following objects are masked from 'package:lubridate':  
##   
## hour, isoweek, mday, minute, month, quarter, second, wday, week,  
## yday, year

## The following objects are masked from 'package:dplyr':  
##   
## between, first, last

## The following object is masked from 'package:purrr':  
##   
## transpose

library(rio)  
library(ggplot2)  
library(scales)

##   
## Attaching package: 'scales'

## The following object is masked from 'package:purrr':  
##   
## discard

## The following object is masked from 'package:readr':  
##   
## col\_factor

library(psych)

##   
## Attaching package: 'psych'

## The following objects are masked from 'package:scales':  
##   
## alpha, rescale

## The following objects are masked from 'package:ggplot2':  
##   
## %+%, alpha

library(here)

## here() starts at /Users/jakecavaiani/Documents/DoD\_Discharge

library(googledrive)  
library(readxl)  
library(cowplot)

##   
## Attaching package: 'cowplot'

## The following object is masked from 'package:lubridate':  
##   
## stamp

library(zoo)

##   
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric

library(readr)  
library(dplyr)  
library(RColorBrewer)  
library(gridExtra)

##   
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':  
##   
## combine

library(ggpmisc)

## Loading required package: ggpp

##   
## Attaching package: 'ggpp'

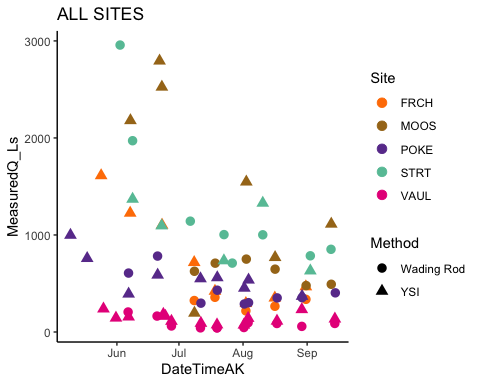
## The following object is masked from 'package:ggplot2':  
##   
## annotate

library(birk)

### ALL Sites

# Import data from google drive #  
discharge.2022 <- "https://docs.google.com/spreadsheets/d/e/2PACX-1vQR6HHHDpnxc6DmNHfdNLR9-dgDHR5Imt0Ve4\_t2DzIF18\_8D3O2da5zcWQzJUSoFQfaetPZDeXZ610/pub?gid=0&single=true&output=csv"  
QSummary.2022 <- read.csv(url(discharge.2022))  
  
### Format Time ###  
QSummary.2022$Date <- mdy(QSummary.2022$Date)  
QSummary.2022$DateTimeAK <- as.POSIXct(paste(QSummary.2022$Date, QSummary.2022$Time), format = "%Y-%m-%d %H:%M", tz = "America/Anchorage")  
QSummary.2022$DateTimeAK <- lubridate::round\_date(QSummary.2022$DateTimeAK, "60 minutes")  
  
### ALL Sites ###  
ggplot(QSummary.2022) +  
 geom\_point(aes(x=DateTimeAK, y=MeasuredQ\_Ls, color=Site, shape=Method), size=3) +  
 theme\_classic() +  
 scale\_color\_manual(values=c("#FF7F00","#A6761D", "#6A3D9A", "#66C2A5", "#E7298A")) +  
 ggtitle("ALL SITES")

## Warning: Removed 10 rows containing missing values (geom\_point).

 Here is all the sites up discrete discharge measurements up until 8/31/2022

eielson.atmo.2022.url <- "https://docs.google.com/spreadsheets/d/e/2PACX-1vRNaeqSnSANyg1akwelIH50Oh3undYyZkhoMku-n0Y9JTpLOY3jK7BJwQlA2ktP40MFZ3OzYdjnGmFm/pub?output=csv"  
  
eielson.atmo.2022 <- read.csv(url(eielson.atmo.2022.url), skip = 6)   
eielson.atmo.2022 <- eielson.atmo.2022[-1,]  
  
names(eielson.atmo.2022) <- c("Site", "DateTimeAK", "TempC", "sea\_level\_pressure", "AirPressure")  
  
eielson.atmo.2022$DateTimeAK <- mdy\_hm(eielson.atmo.2022$DateTimeAK)  
  
eielson.atmo.2022$DateTimeAK <- lubridate::round\_date(eielson.atmo.2022$DateTimeAK, "15 minutes")  
  
eielson.atmo.2022$DateTimeAK <- force\_tz(eielson.atmo.2022$DateTimeAK, "America/Anchorage")  
  
eielson.atmo.2022$AirPressure <- as.numeric(eielson.atmo.2022$AirPressure)

## Warning: NAs introduced by coercion

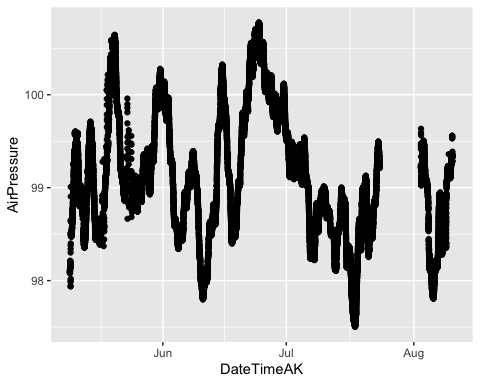
eielson.atmo.2022$AirPressure <- eielson.atmo.2022$AirPressure\*3.38639 # converting from inHG to kPa

### MOOS

This is the raw atmospheric pressure that is located at MOOS…..We only have data through 8/10 at this site because we havent downloaded data off of it since 8/10 field visit. The PT is currently deployed we just dont have it downloaded yet. Due to battery issues for some of the PTs we have had to do some musical chairs with the PTs so this is our most complete atmospheric data with our gear so I will use this sensor to compensate for the rest of the water PTs. I am going to clean this up a little bit:

# cleaning erroneous points   
moos.atmo.2022 <- moos.atmo.2022 %>%  
 mutate(across(c(AirPressure),   
 ~ifelse(DateTimeAK >= "2022-05-01" & DateTimeAK <= "2022-10-01" & AirPressure < 97.5, NA, .)))  
  
moos.atmo.2022 <- moos.atmo.2022 %>%  
 mutate(across(c(AirPressure),   
 ~ifelse(DateTimeAK >= "2022-05-15" & DateTimeAK <= "2022-05-20" & AirPressure < 98.2, NA, .)))  
  
moos.atmo.2022 <- moos.atmo.2022 %>%  
 mutate(across(c(AirPressure),   
 ~ifelse(DateTimeAK >= "2022-05-18" & DateTimeAK <= "2022-05-20" & AirPressure < 99.1, NA, .)))  
  
moos.atmo.2022 <- moos.atmo.2022 %>%  
 mutate(across(c(AirPressure),   
 ~ifelse(DateTimeAK >= "2022-08-10" & DateTimeAK <= "2022-08-12" & AirPressure > 99.5, NA, .)))  
  
moos.atmo.2022 <- moos.atmo.2022 %>%  
 mutate(across(c(AirPressure),   
 ~ifelse(DateTimeAK >= "2022-08-10" & DateTimeAK <= "2022-08-12" & AirPressure < 99, NA, .)))  
  
  
ggplot(moos.atmo.2022, aes(x = DateTimeAK, y = AirPressure)) +  
 geom\_point()

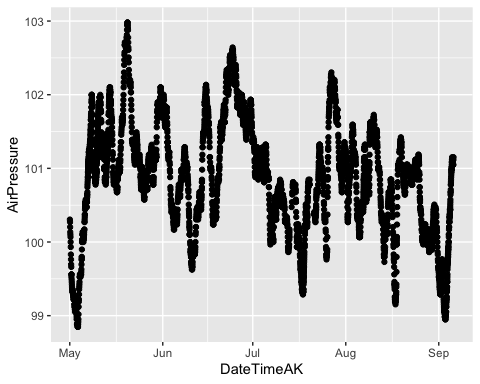
## Warning: Removed 197 rows containing missing values (geom\_point).

 This is a little better

# Eielson Atmospheric Pressure

ggplot(eielson.atmo.2022, aes(x = DateTimeAK, y = AirPressure)) +  
 geom\_point()

## Warning: Removed 1 rows containing missing values (geom\_point).

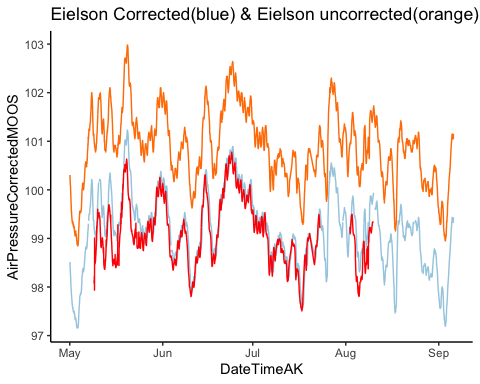
 This is the air pressure record I am going to use when doing the correction

# MOOS elevation is 574 feet  
# POKE 760.96  
# STRT 820 ft  
# FRCH 601 ft   
# VAUL 688.8 ft  
  
  
# compare Eielson and MOOS atmospheric pressure  
eielson.atmo.2022.compare <- eielson.atmo.2022  
eielson.atmo.2022.compare$sea\_level\_pressure <- as.numeric(eielson.atmo.2022.compare$sea\_level\_pressure)  
eielson.atmo.2022.compare$mmHG <- eielson.atmo.2022.compare$sea\_level\_pressure \* 25.44 # converting to mmHG  
  
# conversion to elevation at each site   
# MOOS   
eielson.atmo.2022.compare$mmHGcorrectedMOOS <- eielson.atmo.2022.compare$mmHG - (2.5\*574/100)   
  
eielson.atmo.2022.compare$AirPressureCorrectedMOOS <- eielson.atmo.2022.compare$mmHGcorrectedMOOS \* 0.133322 # converting this to kPA to compare with MOOS PT  
  
# POKE   
eielson.atmo.2022.compare$mmHGcorrectedPOKE <- eielson.atmo.2022.compare$mmHG - (2.5\*760.96/100)   
  
eielson.atmo.2022.compare$AirPressureCorrectedPOKE <- eielson.atmo.2022.compare$mmHGcorrectedPOKE \* 0.133322 # converting this to kPA   
  
# STRT  
eielson.atmo.2022.compare$mmHGcorrectedSTRT <- eielson.atmo.2022.compare$mmHG - (2.5\*820/100)   
  
eielson.atmo.2022.compare$AirPressureCorrectedSTRT <- eielson.atmo.2022.compare$mmHGcorrectedSTRT \* 0.133322 # converting this to kPA   
  
# FRCH  
eielson.atmo.2022.compare$mmHGcorrectedFRCH <- eielson.atmo.2022.compare$mmHG - (2.5\*601/100)   
  
eielson.atmo.2022.compare$AirPressureCorrectedFRCH <- eielson.atmo.2022.compare$mmHGcorrectedFRCH \* 0.133322 # converting this to kPA   
  
# VAUL  
eielson.atmo.2022.compare$mmHGcorrectedVAUL <- eielson.atmo.2022.compare$mmHG - (2.5\*688.8/100)   
  
eielson.atmo.2022.compare$AirPressureCorrectedVAUL <- eielson.atmo.2022.compare$mmHGcorrectedVAUL \* 0.133322 # converting this to kPA   
  
  
moos.atmo.2022.compare <- moos.atmo.2022

This is the uncorrected air pressure comparison between Eielson and MOOS

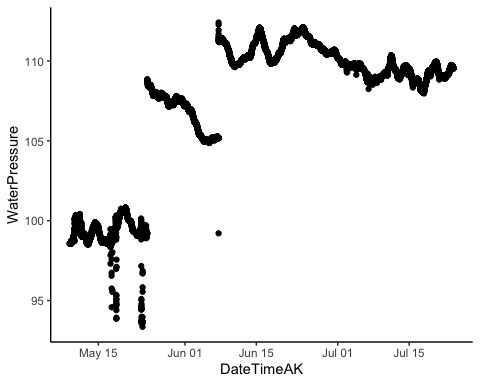
atmo.pt.2022.1 <- left\_join(eielson.atmo.2022.compare, moos.atmo.2022.compare, by = "DateTimeAK")  
  
ggplot(aes(x = DateTimeAK, y = AirPressureCorrectedMOOS), data = atmo.pt.2022.1) +  
 geom\_line(aes(x=DateTimeAK, y=AirPressureCorrectedMOOS), data = atmo.pt.2022.1, color="#A6CEE3") +  
 geom\_line(aes(x=DateTimeAK, y=AirPressure.x), data = atmo.pt.2022.1, color="#FF7F00") +  
 geom\_line(aes(x=DateTimeAK, y=AirPressure.y), data = atmo.pt.2022.1, color = "red") +  
 theme\_classic() +  
 ggtitle("Eielson Corrected(blue) & Eielson uncorrected(orange) & MOOS(red) Atmo P")

## Warning: Removed 1255 row(s) containing missing values (geom\_path).

 Looks like we have to use the corrected Pressure to get what we want for our rating curves

ggplot(moos.stream.one.2022, aes(x = DateTimeAK, y = WaterPressure)) +  
 geom\_point() +  
 theme\_classic()

## Warning: Removed 3 rows containing missing values (geom\_point).

 This is PT1 at MOOS raw. The step changes are from downloading days where the PVC is cleaned and might be at a different height due to sedimentation

I need to clip off the beginning of the record where it wasnt in the water and then I need to adjust the step changes

# adjusting moos.stream.one   
moos.stream.one.2022 <- moos.stream.one.2022 %>%  
 mutate(across(c(WaterPressure),   
 ~ifelse(DateTimeAK >= "2022-05-09" & DateTimeAK <= "2022-05-25", NA, .)))  
  
moos.stream.one.2022 <- moos.stream.one.2022 %>%  
 mutate(across(c(WaterPressure),   
 ~ifelse(DateTimeAK >= "2022-05-09" & WaterPressure < 103, NA, .)))  
  
# shfiting the step change   
moos.1.before <- moos.stream.one.2022[c(1:8404), ] # 6/7 @ 14:15 we cleaned the PVC housing so we are shifting the previous data up to match after the cleaning   
  
moos.1.after <- moos.stream.one.2022[c(8405:21686), ]  
  
  
moos.stream.one.2022[8413, 2] - moos.stream.one.2022[8404, 2] # 6.423

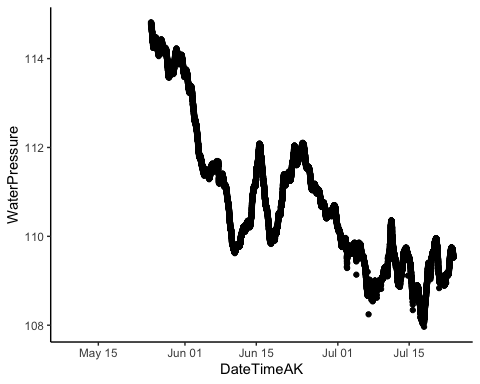
## [1] 6.423

moos.1.before$WaterPressure <- moos.1.before[, 2] + 6.423  
moos.stream.one.2022 <- full\_join(moos.1.before, moos.1.after)

## Joining, by = c("Site", "WaterPressure", "TempC", "DateTimeAK")

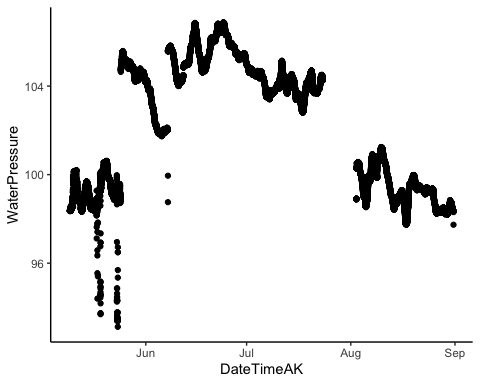
moos.stream.one.2022 <- moos.stream.one.2022 %>%  
 mutate(across(c(WaterPressure),   
 ~ifelse(DateTimeAK >= "2022-06-07" & DateTimeAK <= "2022-06-08" & WaterPressure > 111.9, NA, .)))  
  
  
ggplot(moos.stream.one.2022, aes(x = DateTimeAK, y = WaterPressure)) +  
 geom\_point() +  
 theme\_classic()

## Warning: Removed 4592 rows containing missing values (geom\_point).

 This looks much better

ggplot(moos.stream.two.2022, aes(x = DateTimeAK, y = WaterPressure)) +  
 geom\_point() +  
 theme\_classic()

## Warning: Removed 6 rows containing missing values (geom\_point).

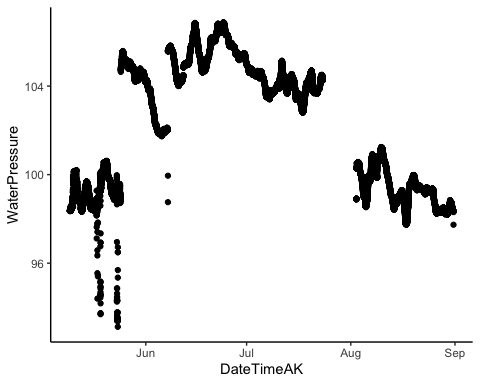


This is PT2 that has the step changes due to the cleaning and then it has a gap at the end of July to beginning of August was because the memory filled up

Let me adjust it now

# adjusting/cleaning moos.stream.two  
  
ggplot(moos.stream.two.2022, aes(x = DateTimeAK, y = WaterPressure)) +  
 geom\_point() +  
 theme\_classic()

## Warning: Removed 6 rows containing missing values (geom\_point).



moos.stream.two.2022 <- moos.stream.two.2022 %>%  
 mutate(across(c(WaterPressure),   
 ~ifelse(DateTimeAK >= "2022-05-09" & DateTimeAK <= "2022-05-25", NA, .))) # PT wasnt placed into the stream until 5/25  
  
moos.stream.two.2022 <- moos.stream.two.2022 %>%  
 mutate(across(c(WaterPressure),   
 ~ifelse(DateTimeAK >= "2022-06-01" & DateTimeAK <= "2022-07-01" & WaterPressure < 101, NA, .)))  
  
# shfiting the step change   
moos.2.before <- moos.stream.two.2022[c(1:8402), ] # 6/7 @ 14:05 we cleaned the PVC housing so we are shifting the previous data up to match after the cleaning   
  
moos.2.after <- moos.stream.two.2022[c(8403:30023), ]  
  
  
moos.stream.two.2022[8408, 2] - moos.stream.two.2022[8402, 2] # 3.573

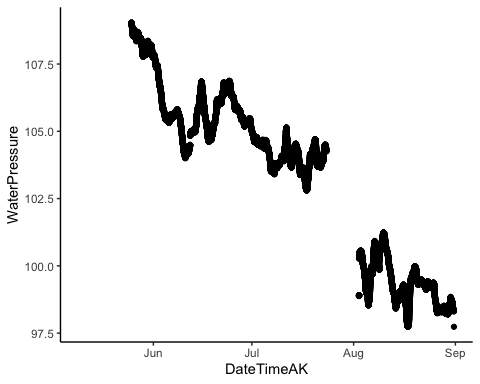
## [1] 3.573

moos.2.before$WaterPressure <- moos.2.before[, 2] + 3.573  
moos.stream.two.2022 <- full\_join(moos.2.before, moos.2.after)

## Joining, by = c("Site", "WaterPressure", "TempC", "DateTimeAK")

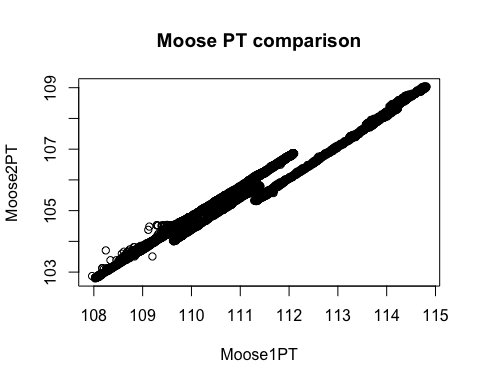
ggplot(moos.stream.two.2022, aes(x = DateTimeAK, y = WaterPressure)) +  
 geom\_point() +  
 theme\_classic()

## Warning: Removed 4593 rows containing missing values (geom\_point).

 This looks much better

# joining two cleaned water pressure together   
moos.final.pressure.2022 <- full\_join(moos.stream.one.2022, moos.stream.two.2022, by = c("DateTimeAK"))  
moos.final.pressure.2022$MeanPressure <- rowMeans(moos.final.pressure.2022[,c(2,6)], na.rm = TRUE)  
  
moos.final.pressure.2022 <- moos.final.pressure.2022[,-c(3,5,7)] # removing columns that are unnecessary. I only need both water pressures and mean pressure with a DateTime column   
  
  
# eielson.atmo.2022 <- eielson.atmo.2022[,-c(1,3,4)] # i dont need site, Temp and sea\_level\_pressure column  
  
eielson.atmo.2022.compare <- eielson.atmo.2022.compare[ , -which(names(eielson.atmo.2022.compare) %in% c("Site","TempC", "sea\_level\_pressure", "AirPressure", "mmHG", "mmHGcorrectedMOOS","mmHGcorrectedPOKE","mmHGcorrectedSTRT","mmHGcorrectedFRCH","mmHGcorrectedVAUL"))] # removing columns I do not need  
  
  
#join the two atmospheric and water pressure together  
MOOS.2022 <- left\_join(eielson.atmo.2022.compare, moos.final.pressure.2022, by = "DateTimeAK")  
  
# Water pressure - atmospheric pressure  
MOOS.2022$difference <- MOOS.2022$MeanPressure - MOOS.2022$AirPressureCorrectedMOOS  
  
names(MOOS.2022)[names(MOOS.2022) == 'Site.x'] <- 'Site'  
MOOS.2022$Site <- "MOOS"

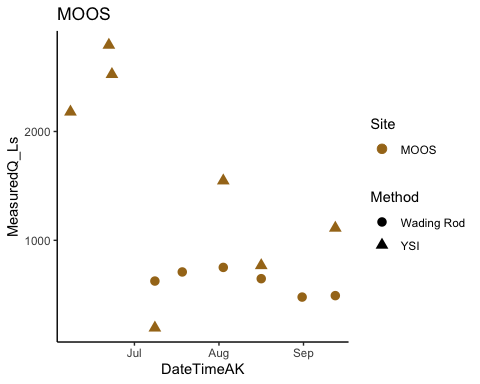
# Checking closeness between two PTs  
moos.stream.two.2022 <- moos.stream.two.2022[1:nrow(moos.stream.one.2022),]  
moos.stream.one.2022$Site <- "MOOS1" #add column identifier  
moos.stream.two.2022$Site <- "MOOS2"  
moos.pt.2022 <- bind\_rows(moos.stream.one.2022, moos.stream.two.2022)  
  
plot(x = moos.stream.one.2022$WaterPressure, y = moos.stream.two.2022$WaterPressure, main = "Moose PT comparison",  
 xlab = "Moose1PT",   
 ylab = "Moose2PT")

 This is the closeness between the two PTs so they are both pretty similar to each other

gap from 7/23/22 @9:15 - 8/2 @ 9:00

### Filter MOOS ###  
QSummary.MO.2022 <- QSummary.2022 %>% filter(Site =="MOOS")  
  
ggplot(QSummary.MO.2022) +  
 geom\_point(aes(x = DateTimeAK, y = MeasuredQ\_Ls, shape = Method, color = Site), size=3) +  
 theme\_classic() +  
 scale\_color\_manual(values=c("#A6761D")) +   
 ggtitle("MOOS")

## Warning: Removed 2 rows containing missing values (geom\_point).

 There should be 6 wading rod points

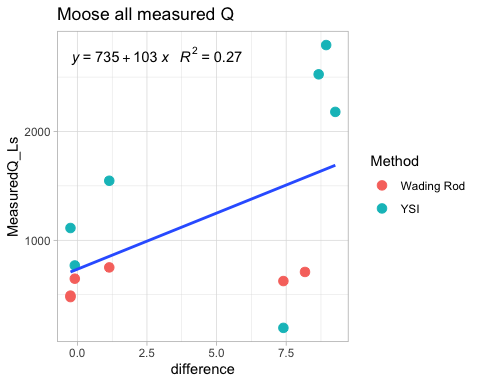
# trying to merge by nearest date if we have an offset point   
MOOS.2022.dt <- setDT(MOOS.2022)  
MOOS.2022.dt <- subset(MOOS.2022.dt, DateTimeAK < "2022-08-31 06:00:00") # removing rows that had dates corresponding to end of record that messed up the rolling nearest function   
QSummary.MO.2022.dt <- QSummary.MO.2022  
  
Moose1comb.2022 <- MOOS.2022.dt[QSummary.MO.2022.dt, on = "DateTimeAK", roll = 'nearest']  
  
MOOS1.lm.2022 <- lm(Moose1comb.2022$MeasuredQ\_Ls ~ Moose1comb.2022$difference)

# plot rating curve   
moos.formula = y~x  
  
ggplot(aes(x = difference, y = MeasuredQ\_Ls), data = Moose1comb.2022) +  
 geom\_point(aes(color = Method), size = 3) +  
 geom\_smooth(method = "lm", se=FALSE, formula = moos.formula) +  
 stat\_poly\_eq(formula = moos.formula,  
 aes(label = paste(..eq.label.., ..rr.label.., sep = "~~~")),  
 parse = TRUE) +  
 #xlim(216, 216.4) +  
 #ylim(0,1500) +  
 theme\_light() +  
 ggtitle("Moose all measured Q")

## Warning: Removed 2 rows containing non-finite values (stat\_smooth).

## Warning: Removed 2 rows containing non-finite values (stat\_poly\_eq).

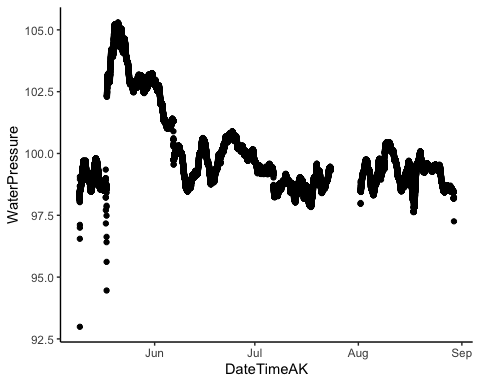
## Warning: Removed 2 rows containing missing values (geom\_point).

 We have all 6 wading rod points here

### POKE ####  
poke.stream.one.2022.url <- "https://docs.google.com/spreadsheets/d/e/2PACX-1vR8p1pdbkQBMo54XaZinGwxyzojCnlkXeZid3EFvmt9v31PFvpsa1DlWClj8aG0kkJIuU57WXl-cq7Q/pub?output=csv"  
poke.stream.two.2022.url <- "https://docs.google.com/spreadsheets/d/e/2PACX-1vR-wJjf3rO8eC7bMpQZgwqFLHFQpf5qCInyWtUF1PqaCB8Z\_EoM1cTHRUmWypjftREIx1rP0V6zYJxL/pub?output=csv"  
  
# load in url  
poke.stream.one.2022 <- read.csv(url(poke.stream.one.2022.url), skip = 1)   
poke.stream.two.2022 <- read.csv(url(poke.stream.two.2022.url), skip = 1)   
  
# cleaning df to be able to interpret and merge  
poke.stream.one.2022 <- poke.stream.one.2022[, -c(5:18)] # removing columns that arent date/abs pressure and temp  
  
poke.stream.two.2022 <- poke.stream.two.2022[, -c(5:9)] # removing columns that arent date/abs pressure and temp  
  
# changing to AK time (It reads in as GMT but it is actually AKST )  
poke.stream.one.2022$DateTimeAK <- mdy\_hms(poke.stream.one.2022$Date.Time..GMT.08.00)  
  
poke.stream.two.2022$DateTimeAK <- mdy\_hms(poke.stream.two.2022$Date.Time..GMT.08.00)  
  
# round date to 5 minute intervals  
poke.stream.one.2022$DateTimeAK <- lubridate::round\_date(poke.stream.one.2022$DateTimeAK, "5 minutes")  
poke.stream.two.2022$DateTimeAK <- lubridate::round\_date(poke.stream.two.2022$DateTimeAK, "5 minutes")  
  
# cleaning off original datetime   
poke.stream.one.2022 <- poke.stream.one.2022[, -c(2)] # removing columns that aren't date/abs pressure and temp  
poke.stream.two.2022 <- poke.stream.two.2022[, -c(2)] # removing columns that arent   
  
  
names(poke.stream.one.2022) <- c("Site", "WaterPressure", "TempC", "DateTimeAK")  
names(poke.stream.two.2022) <- c("Site", "WaterPressure", "TempC", "DateTimeAK")

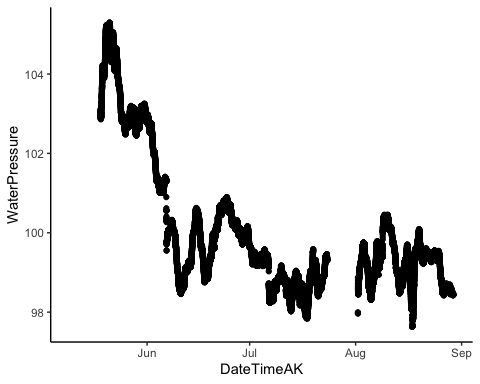
ggplot(poke.stream.one.2022, aes(x = DateTimeAK, y = WaterPressure))+  
 geom\_point() +  
 theme\_classic()

## Warning: Removed 6 rows containing missing values (geom\_point).

 This is PT1….step change is air pressure and then placed in the stream. Let me clip that out

#adjusting/cleaning poke.stream.one  
poke.stream.one.2022 <- poke.stream.one.2022 %>%  
 mutate(across(c(WaterPressure),   
 ~ifelse(DateTimeAK >= "2022-05-09" & DateTimeAK <= "2022-05-18", NA, .))) # PT wasnt placed into the stream until 5/17  
  
poke.stream.one.2022 <- poke.stream.one.2022 %>%  
 mutate(across(c(WaterPressure),   
 ~ifelse(DateTimeAK >= "2022-08-27" & DateTimeAK <= "2022-08-31" & WaterPressure < 98.3, NA, .)))  
  
ggplot(poke.stream.one.2022, aes(x = DateTimeAK, y = WaterPressure))+  
 geom\_point() +  
 theme\_classic()

## Warning: Removed 2598 rows containing missing values (geom\_point).

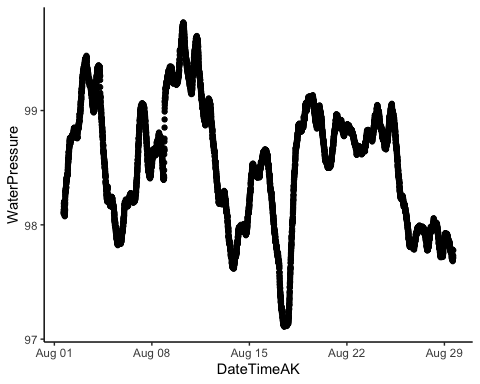


# plot(poke.stream.one.2022$DateTime, poke.stream.one.2022$WaterPressure,  
# xlim = as.POSIXct(c("2022-08-27 00:00:00", "2022-08-31 00:00:00")),  
# ylim = c(97, 99))

Out of water points were clipped off

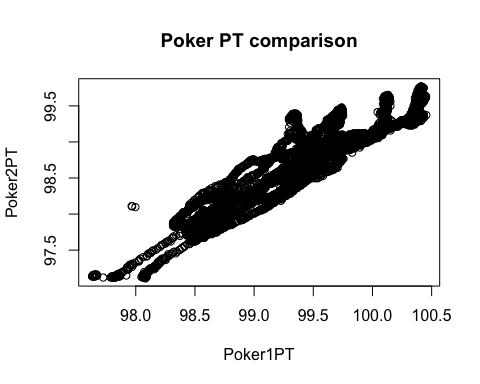
ggplot(poke.stream.two.2022, aes(x = DateTimeAK, y = WaterPressure))+  
 geom\_point() +  
 theme\_classic()

## Warning: Removed 3 rows containing missing values (geom\_point).

 This is PT2…we only have data from 8/1 because of our musical chairs situation. We got two PTs back from onset so we deployed a second PT at POKE

# merge to one   
poke.final.pressure.2022 <- left\_join(poke.stream.one.2022, poke.stream.two.2022, by = c("DateTimeAK"))  
poke.final.pressure.2022$MeanPressure <- rowMeans(poke.final.pressure.2022[,c(2,6)], na.rm = TRUE)

# Checking closeness between two PTs  
poke.pt.2022 <- right\_join(poke.stream.one.2022, poke.stream.two.2022, by = "DateTimeAK")  
# poke.stream.one.2022 <- poke.stream.one.2022[1:nrow(poke.stream.two.2022),]  
# poke.stream.one.2022$Site <- "POKE1" #add column identifier  
# poke.stream.two.2022$Site <- "POKE2"  
#poke.pt.2022 <- bind\_rows(poke.stream.one.2022, poke.stream.two.2022)  
  
plot(x = poke.pt.2022$WaterPressure.x, y = poke.pt.2022$WaterPressure.y, main = "Poker PT comparison",  
 xlab = "Poker1PT",   
 ylab = "Poker2PT")

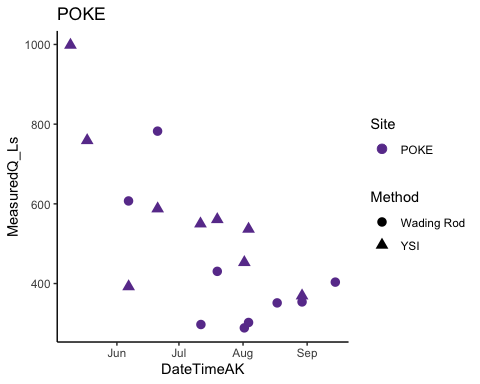


# join the two atmospheric and water pressure together

POKE.2022 <- left\_join(eielson.atmo.2022.compare, poke.final.pressure.2022, by = "DateTimeAK")  
  
# Water pressure - atmospheric pressure   
POKE.2022$difference <- POKE.2022$MeanPressure - POKE.2022$AirPressureCorrectedPOKE  
  
POKE.2022 <- POKE.2022[ , -which(names(POKE.2022) %in% c("WaterPressure.x","TempC.x", "Site.y", "WaterPressure.y", "TempC.y"))] # removing columns I do not need  
  
names(POKE.2022)[names(POKE.2022) == 'Site.x'] <- 'Site'  
POKE.2022$Site <- "POKE"

### Filter POKE ###  
QSummary.PO.2022 <- QSummary.2022 %>% filter(Site =="POKE")  
  
ggplot(QSummary.PO.2022) +  
 geom\_point(aes(x = DateTimeAK, y = MeasuredQ\_Ls, shape = Method, color = Site), size=3) +  
 theme\_classic() +  
 scale\_color\_manual(values=c("#6A3D9A")) +   
 ggtitle("POKE")

## Warning: Removed 2 rows containing missing values (geom\_point).

 There looks like we should have 9 wading rod points

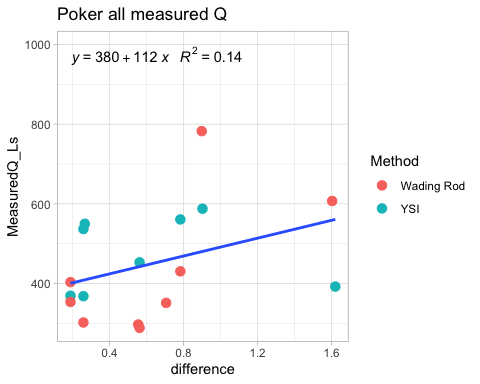
# trying to merge by nearest date if we have an offset point   
POKE.2022.dt <- setDT(POKE.2022)  
POKE.2022.dt <- subset(POKE.2022.dt, DateTimeAK < "2022-08-29 07:45:00") # removing rows that had dates corresponding to end of record that messed up the rolling nearest function   
QSummary.PO.2022.dt <- QSummary.PO.2022  
  
Poker1comb.2022 <- POKE.2022.dt[QSummary.PO.2022.dt, on = "DateTimeAK", roll = 'nearest']  
  
Poker1comb.2022[15,9] <- 0.2580529  
  
POKE1.lm.2022 <- lm(Poker1comb.2022$MeasuredQ\_Ls ~ Poker1comb.2022$difference)

### Rating Curve ###  
poke.formula <- y ~ x  
ggplot(aes(x = difference, y = MeasuredQ\_Ls), data = Poker1comb.2022) +  
 geom\_point(aes(color = Method), size = 3) +  
 geom\_smooth(method = "lm", se=FALSE, formula = poke.formula) +  
 stat\_poly\_eq(formula = poke.formula,  
 aes(label = paste(..eq.label.., ..rr.label.., sep = "~~~")),  
 parse = TRUE) +  
 # xlim(-2, 2) +  
 #ylim(0,1500) +  
 theme\_light() +  
 ggtitle("Poker all measured Q")

## Warning: Removed 3 rows containing non-finite values (stat\_smooth).

## Warning: Removed 3 rows containing non-finite values (stat\_poly\_eq).

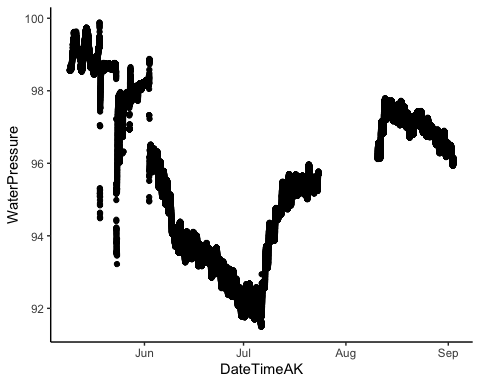
## Warning: Removed 3 rows containing missing values (geom\_point).

 This is the rating curve. We are missing 5/9 and 5/17 YSI because we didn’t have our Pressure Transducers in yet. rolling date join

### STRT ####  
strt.stream.one.2022.url <- "https://docs.google.com/spreadsheets/d/e/2PACX-1vS4XSmuumKD-j1aDhpA1gfyfrpqUiJUTuvcP9UGrgagvIrzlGYWk71hl0zAC8g8GHqOm1ONjWOpO\_So/pub?output=csv" # WL  
strt.stream.two.2022.url <- "https://docs.google.com/spreadsheets/d/e/2PACX-1vRhNn42-Jr1Y3aTFfMPiRxKrNZZjHeBBxkCmVcLVPQEKBf4qg\_1Pw4nyUBr3mDwtEE8NKGIbS7kNSrS/pub?output=csv" #WR  
  
# load in data   
strt.stream.one.2022 <- read.csv(url(strt.stream.one.2022.url), skip = 1)   
strt.stream.two.2022 <- read.csv(url(strt.stream.two.2022.url), skip = 1)   
  
# clean for merging purposes  
strt.stream.one.2022 <- strt.stream.one.2022[, -c(4,6:19)] # removing columns that arent date/abs pressure and temp  
  
strt.stream.two.2022 <- strt.stream.two.2022[, -c(5:12)] # removing columns that arent date/abs pressure and temp  
  
# changing to AK time (It reads in as GMT but it is actually AKST )  
strt.stream.one.2022$DateTimeAK <- mdy\_hms(strt.stream.one.2022$Date.Time..GMT.08.00)  
  
strt.stream.two.2022$DateTimeAK <- mdy\_hms(strt.stream.two.2022$Date.Time..GMT.08.00)  
  
# round date to 5 minute intervals  
strt.stream.one.2022$DateTimeAK <- lubridate::round\_date(strt.stream.one.2022$DateTimeAK, "5 minutes")  
strt.stream.two.2022$DateTimeAK <- lubridate::round\_date(strt.stream.two.2022$DateTimeAK, "5 minutes")  
  
# cleaning off original datetime   
strt.stream.one.2022 <- strt.stream.one.2022[, -c(2)] # removing columns that aren't date/abs pressure and temp  
strt.stream.two.2022 <- strt.stream.two.2022[, -c(2)] # removing columns that arent   
  
  
names(strt.stream.one.2022) <- c("Site","WaterPressure", "TempC", "DateTimeAK")  
  
names(strt.stream.two.2022) <- c("Site","WaterPressure", "TempC", "DateTimeAK")

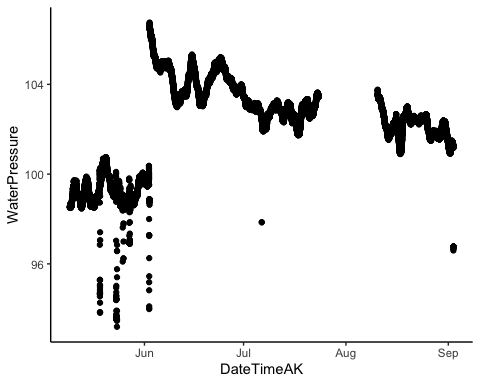
ggplot(strt.stream.one.2022, aes(x = DateTimeAK, y = WaterPressure))+  
 geom\_point() +  
 theme\_classic()

## Warning: Removed 12 rows containing missing values (geom\_point).

 This is PT1 for STRT raw. Looking at how low the PT data is I am NOT going to trust this one and take it out of our analysis

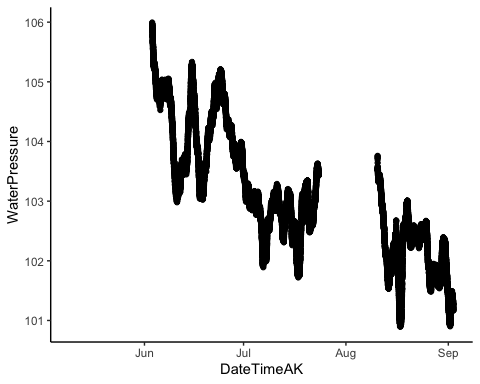
ggplot(strt.stream.two.2022, aes(x = DateTimeAK, y = WaterPressure))+  
 geom\_point() +  
 theme\_classic()

## Warning: Removed 3 rows containing missing values (geom\_point).

 This is raw PT2 for STRT. I need to remove out of water points (anything before 6/2) and clean any erroneous points

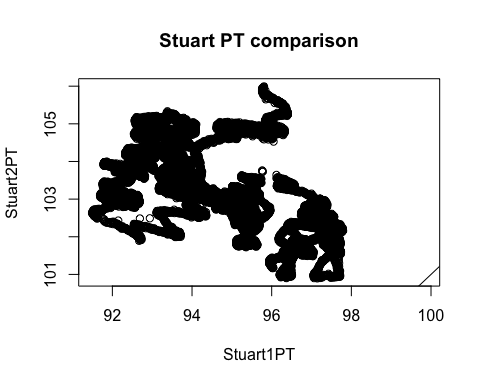
#adjusting/cleaning STRT 2  
strt.stream.two.2022 <- strt.stream.two.2022 %>%  
 mutate(across(c(WaterPressure),   
 ~ifelse(DateTimeAK >= "2022-05-09" & DateTimeAK <= "2022-06-03", NA, .))) # PT wasnt placed into the stream until 6/3  
  
strt.stream.two.2022 <- strt.stream.two.2022 %>%  
 mutate(across(c(WaterPressure),   
 ~ifelse(DateTimeAK >= "2022-05-09" & DateTimeAK <= "2022-10-03" &  
 WaterPressure < 100, NA, .))) # PT wasnt placed into the stream until 6/3  
  
  
ggplot(strt.stream.two.2022, aes(x = DateTimeAK, y = WaterPressure))+  
 geom\_point() +  
 theme\_classic()

## Warning: Removed 7194 rows containing missing values (geom\_point).

 This looks better

# merge to one   
strt.final.pressure.2022 <- strt.stream.two.2022 # the first PT is crap

# Checking closeness between two PTs  
strt.stream.two.2022 <- strt.stream.two.2022[1:nrow(strt.stream.one.2022),]  
strt.stream.one.2022$Site <- "STRT1" #add column identifier  
strt.stream.two.2022$Site <- "STRT2"  
strt.pt.2022 <- bind\_rows(strt.stream.one.2022, strt.stream.two.2022)  
  
plot(x = strt.stream.one.2022$WaterPressure, y = strt.stream.two.2022$WaterPressure, main = "Stuart PT comparison",  
 xlab = "Stuart1PT",   
 ylab = "Stuart2PT")  
abline(1,1)

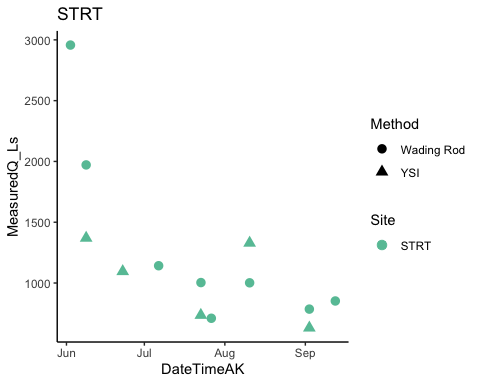


I wouldnt trust the first PT at all TBH

# join the two atmospheric and water pressure together  
STRT.2022 <- left\_join(eielson.atmo.2022.compare, strt.final.pressure.2022, by = "DateTimeAK")  
  
# Water pressure - atmospheric pressure   
STRT.2022$difference <- STRT.2022$WaterPressure - STRT.2022$AirPressureCorrectedSTRT  
  
STRT.2022 <- STRT.2022[ , -which(names(STRT.2022) %in% c("TempC"))] # removing columns I do not need  
STRT.2022$Site <- "STRT"

### Filter STRT ###  
QSummary.ST.2022 <- QSummary.2022 %>% filter(Site =="STRT")  
  
ggplot(QSummary.ST.2022) +  
 geom\_point(aes(x = DateTimeAK, y = MeasuredQ\_Ls, shape = Method, color = Site), size=3) +  
 theme\_classic() +  
 scale\_color\_manual(values=c("#66C2A5")) +   
 ggtitle("STRT")

## Warning: Removed 3 rows containing missing values (geom\_point).



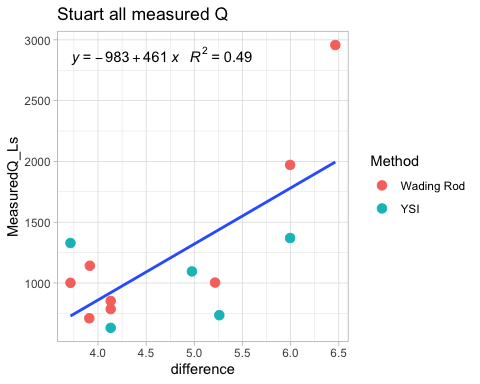
# trying to merge by nearest date if we have an offset point   
STRT.2022.dt <- setDT(STRT.2022)  
STRT.2022.dt <- subset(STRT.2022.dt, DateTimeAK < "2022-09-02 05:00:00") # removing rows that had dates corresponding to end of record that messed up the rolling nearest function   
QSummary.ST.2022.dt <- QSummary.ST.2022  
  
Stuart1comb.2022 <- STRT.2022.dt[QSummary.ST.2022.dt, on = "DateTimeAK", roll = 'nearest']  
Stuart1comb.2022[c(1,2),9] <- 6.463909 # merging by nearest difference value manually  
Stuart1comb.2022[c(10,11),9] <- 3.910157  
Stuart1comb.2022[c(12,13),9] <- 3.714323  
  
  
STRT1.lm.2022 <- lm(Stuart1comb.2022$MeasuredQ\_Ls ~ Stuart1comb.2022$difference)

### Filter STRT ###  
strt.formula <- y ~ x  
  
ggplot(aes(x = difference, y = MeasuredQ\_Ls), data = Stuart1comb.2022) +  
 geom\_point(aes(color = Method), size = 3) +  
 geom\_smooth(method = "lm", se=FALSE, formula = strt.formula) +  
 stat\_poly\_eq(formula = strt.formula,  
 aes(label = paste(..eq.label.., ..rr.label.., sep = "~~~")),  
 parse = TRUE) +  
 #xlim(216, 216.4) +  
 #ylim(0,1500) +  
 theme\_light() +  
 ggtitle("Stuart all measured Q")

## Warning: Removed 3 rows containing non-finite values (stat\_smooth).

## Warning: Removed 3 rows containing non-finite values (stat\_poly\_eq).

## Warning: Removed 3 rows containing missing values (geom\_point).

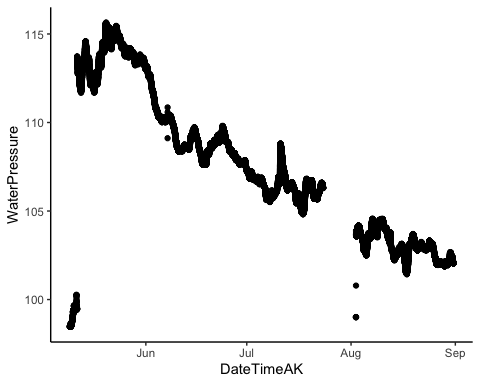


### FRCH

frch.stream.one.2022.url <- "https://docs.google.com/spreadsheets/d/e/2PACX-1vRkT5GIhFY4nLblzPi4Upc8y6Pk2k4vXLSGAkFyMhLowsCmct0fZ5rxm-vd85RfFr7YLrCuuSiOqQbL/pub?output=csv"  
frch.stream.two.2022.url <- "https://docs.google.com/spreadsheets/d/e/2PACX-1vRiDAbzBj\_jHk1Hd3iM7fFvzMrRCSF9fbKTcmQD\_dUFGOjoIQuBZBKV4KdNhfevLOq0Udch0LbWiKXw/pub?output=csv"  
  
# load in url  
frch.stream.one.2022 <- read.csv(url(frch.stream.one.2022.url), skip = 1)   
frch.stream.two.2022 <- read.csv(url(frch.stream.two.2022.url), skip = 1)   
  
# clean for merging purposes  
frch.stream.one.2022 <- frch.stream.one.2022[, -c(5:13)] # removing columns that arent date/abs pressure and temp  
  
frch.stream.two.2022 <- frch.stream.two.2022[, -c(5:14)] # removing columns that arent date/abs pressure and temp  
  
# changing to AK time (It reads in as GMT but it is actually AKST )  
frch.stream.one.2022$DateTimeAK <- mdy\_hms(frch.stream.one.2022$Date.Time..GMT.08.00)  
  
frch.stream.two.2022$DateTimeAK <- mdy\_hms(frch.stream.two.2022$Date.Time..GMT.08.00)  
  
# round date to 5 minute intervals  
frch.stream.one.2022$DateTimeAK <- lubridate::round\_date(frch.stream.one.2022$DateTimeAK, "5 minutes")  
frch.stream.two.2022$DateTimeAK <- lubridate::round\_date(frch.stream.two.2022$DateTimeAK, "5 minutes")  
  
# cleaning off original datetime   
frch.stream.one.2022 <- frch.stream.one.2022[, -c(2)] # removing columns that aren't date/abs pressure and temp  
frch.stream.two.2022 <- frch.stream.two.2022[, -c(2)] # removing columns that arent   
  
names(frch.stream.one.2022) <- c("Site", "WaterPressure", "TempC", "DateTimeAK")  
  
names(frch.stream.two.2022) <- c("Site", "WaterPressure", "TempC", "DateTimeAK")  
  
  
# merge to one   
frch.final.pressure.2022 <- left\_join(frch.stream.one.2022, frch.stream.two.2022, by = c("DateTimeAK"))  
frch.final.pressure.2022$MeanPressure <- rowMeans(frch.final.pressure.2022[,c(2,6)], na.rm = TRUE)

ggplot(frch.stream.one.2022, aes(x = DateTimeAK, y = WaterPressure))+  
 geom\_point() +  
 theme\_classic()

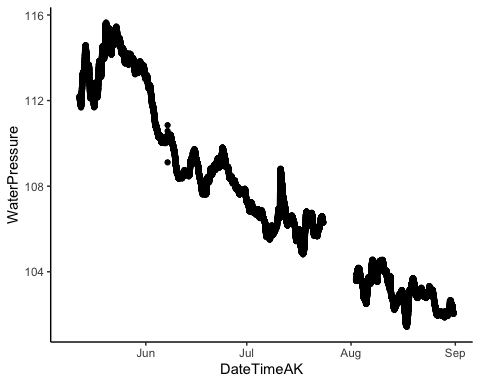
## Warning: Removed 6 rows containing missing values (geom\_point).



This looks like the best one yet…we need to remove the beginning out of water points

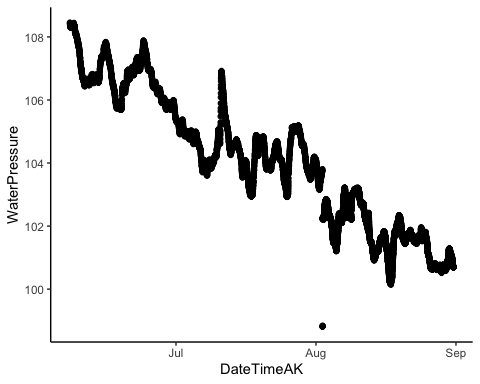
frch.stream.one.2022 <- frch.stream.one.2022 %>%  
 mutate(across(c(WaterPressure),   
 ~ifelse(DateTimeAK >= "2022-05-09" & DateTimeAK <= "2022-05-12", NA, .))) # PT wasnt placed into the stream until 5/12  
  
frch.stream.one.2022 <- frch.stream.one.2022 %>%  
 mutate(across(c(WaterPressure),   
 ~ifelse(DateTimeAK >= "2022-08-01" & DateTimeAK <= "2022-08-15" &  
 WaterPressure < 102, NA, .))) # PT wasnt placed into the stream until 5/12  
  
ggplot(frch.stream.one.2022, aes(x = DateTimeAK, y = WaterPressure))+  
 geom\_point() +  
 theme\_classic()

## Warning: Removed 853 rows containing missing values (geom\_point).

 This looks better

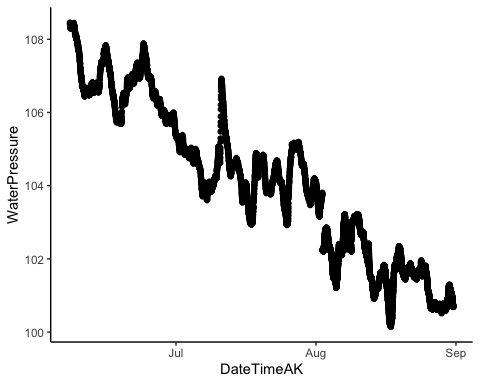
ggplot(frch.stream.two.2022, aes(x = DateTimeAK, y = WaterPressure))+  
 geom\_point() +  
 theme\_classic()

## Warning: Removed 10 rows containing missing values (geom\_point).

 PT2 also looks good…let me remove that one point in august that is below everything else

frch.stream.two.2022 <- frch.stream.two.2022 %>%  
 mutate(across(c(WaterPressure),   
 ~ifelse(DateTimeAK >= "2022-08-01" & DateTimeAK <= "2022-08-15" &  
 WaterPressure < 100, NA, .))) # PT wasnt placed into the stream until 5/12  
  
ggplot(frch.stream.two.2022, aes(x = DateTimeAK, y = WaterPressure))+  
 geom\_point() +  
 theme\_classic()

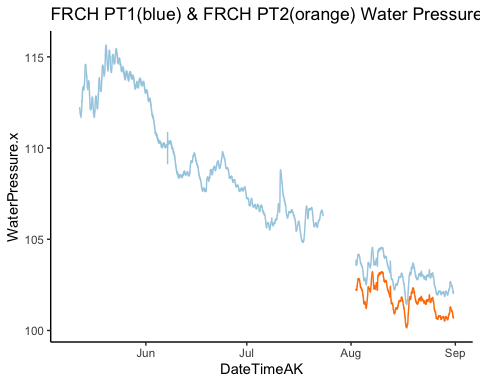
## Warning: Removed 17 rows containing missing values (geom\_point).



# Checking closeness between two PTs  
atmo.pt.frch <- left\_join(frch.stream.one.2022, frch.stream.two.2022, by = "DateTimeAK")  
  
ggplot(aes(x = DateTimeAK, y = WaterPressure.x), data = atmo.pt.frch) +  
 geom\_line(aes(x=DateTimeAK, y=WaterPressure.x), data = atmo.pt.frch, color="#A6CEE3") +  
 geom\_line(aes(x=DateTimeAK, y=WaterPressure.y), data = atmo.pt.frch, color="#FF7F00") +  
 theme\_classic() +  
 ggtitle("FRCH PT1(blue) & FRCH PT2(orange) Water Pressure")

## Warning: Removed 850 row(s) containing missing values (geom\_path).

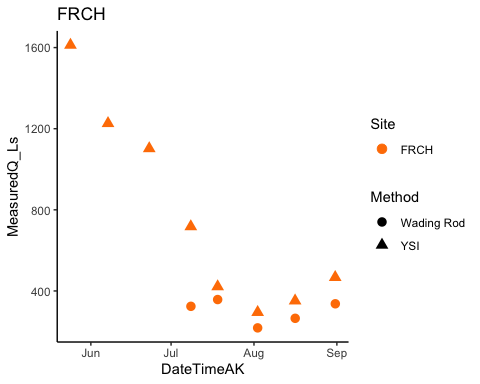
## Warning: Removed 8391 row(s) containing missing values (geom\_path).



# join the two atmospheric and water pressure together  
FRCH.2022 <- left\_join(eielson.atmo.2022.compare, frch.final.pressure.2022, by = "DateTimeAK")  
  
  
# Water pressure - atmospheric pressure   
FRCH.2022$difference <- FRCH.2022$MeanPressure - FRCH.2022$AirPressureCorrectedFRCH  
  
FRCH.2022 <- FRCH.2022[ , -which(names(FRCH.2022) %in% c("WaterPressure.x", "TempC.x", "Site.y", "WaterPressure.y", "TempC.y"))] # removing columns I do not need  
  
names(FRCH.2022)[names(FRCH.2022) == 'Site.x'] <- 'Site'  
  
FRCH.2022$Site <- "FRCH"

### Filter FRCH ###  
QSummary.FR.2022 <- QSummary.2022 %>% filter(Site =="FRCH")  
  
ggplot(QSummary.FR.2022) +  
 geom\_point(aes(x = DateTimeAK, y = MeasuredQ\_Ls, shape = Method, color = Site), size=3) +  
 theme\_classic() +  
 scale\_color\_manual(values=c("#FF7F00")) +   
 ggtitle("FRCH")

## Warning: Removed 1 rows containing missing values (geom\_point).



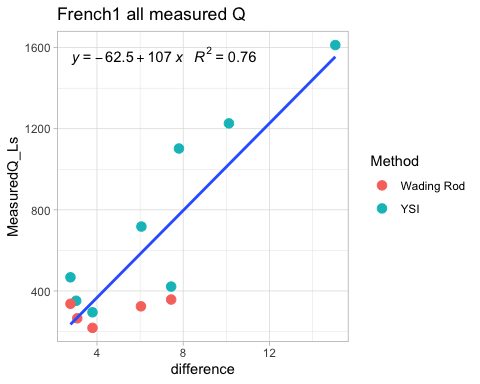
# trying to merge by nearest date if we have an offset point   
FRCH.2022.dt <- setDT(FRCH.2022)  
FRCH.2022.dt <- subset(FRCH.2022.dt, DateTimeAK < "2022-08-31 03:00:00") # removing rows that had dates corresponding to end of record that messed up the rolling nearest function   
QSummary.FR.2022.dt <- QSummary.FR.2022  
  
French1comb.2022 <- FRCH.2022.dt[QSummary.FR.2022.dt, on = "DateTimeAK", roll = 'nearest']  
  
FRCH1.lm.2022 <- lm(French1comb.2022$MeasuredQ\_Ls ~ French1comb.2022$difference)

# rating curve #   
frch.formula <- y ~ x  
  
ggplot(aes(x = difference, y = MeasuredQ\_Ls), data = French1comb.2022) +  
 geom\_point(aes(color = Method), size = 3) +  
 geom\_smooth(method = "lm", se=FALSE, formula = frch.formula) +  
 stat\_poly\_eq(formula = frch.formula,  
 aes(label = paste(..eq.label.., ..rr.label.., sep = "~~~")),  
 parse = TRUE) +  
 #xlim(216, 216.4) +  
 #ylim(0,1500) +  
 theme\_light() +  
 ggtitle("French1 all measured Q")

## Warning: Removed 1 rows containing non-finite values (stat\_smooth).

## Warning: Removed 1 rows containing non-finite values (stat\_poly\_eq).

## Warning: Removed 1 rows containing missing values (geom\_point).

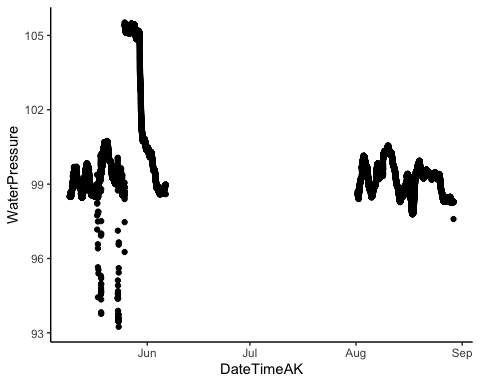
 FRCH looks happy. I am missing 8/31 discharge measurement due to downloading data off before discharge was taken so it is off by a couple of hours and will be recouped during next download

# VAUL

# vaul.stream.one.2022.url <- "https://docs.google.com/spreadsheets/d/e/2PACX-1vRkT5GIhFY4nLblzPi4Upc8y6Pk2k4vXLSGAkFyMhLowsCmct0fZ5rxm-vd85RfFr7YLrCuuSiOqQbL/pub?output=csv"  
vaul.stream.two.2022.url <- "https://docs.google.com/spreadsheets/d/e/2PACX-1vSR7qqdF5BXklT1ocG9bLLUwJ07ha8qmzAVRPaUjm7mUq12VptmViI9NJcW1-jO4cu0KtHBLl90A4DZ/pub?output=csv"  
  
# load in url  
#vaul.stream.one.2022 <- read.csv(url(vaul.stream.one.2022.url), skip = 1)   
vaul.stream.two.2022 <- read.csv(url(vaul.stream.two.2022.url), skip = 1)   
  
# clean for merging purposes  
#vaul.stream.one.2022 <- vaul.stream.one.2022[, -c(5:13)] # removing columns that arent date/abs pressure and temp  
  
vaul.stream.two.2022 <- vaul.stream.two.2022[, -c(5:13)] # removing columns that arent date/abs pressure and temp  
  
# changing to AK time (It reads in as GMT but it is actually AKST )  
#vaul.stream.one.2022$DateTimeAK <- mdy\_hms(vaul.stream.one.2022$Date.Time..GMT.08.00)  
  
vaul.stream.two.2022$DateTimeAK <- mdy\_hms(vaul.stream.two.2022$Date.Time..GMT.08.00)  
  
# round date to 5 minute intervals  
#vaul.stream.one.2022$DateTimeAK <- lubridate::round\_date(vaul.stream.one.2022$DateTimeAK, "5 minutes")  
vaul.stream.two.2022$DateTimeAK <- lubridate::round\_date(vaul.stream.two.2022$DateTimeAK, "5 minutes")  
  
# cleaning off original datetime   
#vaul.stream.one.2022 <- vaul.stream.one.2022[, -c(2)] # removing columns that aren't date/abs pressure and temp  
vaul.stream.two.2022 <- vaul.stream.two.2022[, -c(2)] # removing columns that arent   
  
#names(vaul.stream.one.2022) <- c("Site", "WaterPressure", "TempC", "DateTimeAK")  
  
names(vaul.stream.two.2022) <- c("Site", "WaterPressure", "TempC", "DateTimeAK")  
  
  
# # merge to one   
# frch.final.pressure.2022 <- left\_join(vaul.stream.one.2022, vaul.stream.two.2022, by = c("DateTimeAK"))  
# frch.final.pressure.2022$MeanPressure <- rowMeans(frch.final.pressure.2022[,c(2,6)], na.rm = TRUE)

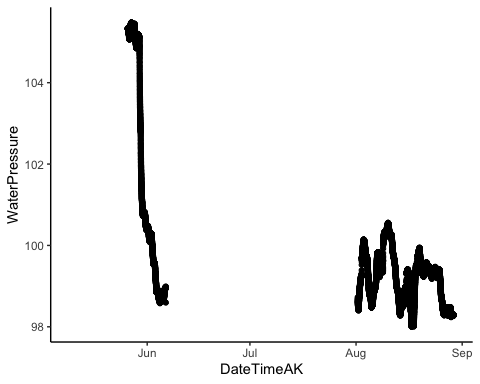
ggplot(vaul.stream.two.2022, aes(x = DateTimeAK, y = WaterPressure))+  
 geom\_point() +  
 theme\_classic()

## Warning: Removed 5 rows containing missing values (geom\_point).

 This is the raw record of VAUL pressure. The beginning is airpressure and then there is a gap as we had a relaunch issue.

vaul.stream.two.2022 <- vaul.stream.two.2022 %>%  
 mutate(across(c(WaterPressure),   
 ~ifelse(DateTimeAK >= "2022-05-09" & DateTimeAK <= "2022-05-26", NA, .))) # PT wasnt placed into the stream until 5/26  
  
vaul.stream.two.2022 <- vaul.stream.two.2022 %>%  
 mutate(across(c(WaterPressure),   
 ~ifelse(DateTimeAK >= "2022-08-15" & DateTimeAK <= "2022-08-31" &  
 WaterPressure < 98, NA, .))) # PT wasnt placed into the stream until 5/12  
  
ggplot(vaul.stream.two.2022, aes(x = DateTimeAK, y = WaterPressure))+  
 geom\_point() +  
 theme\_classic()

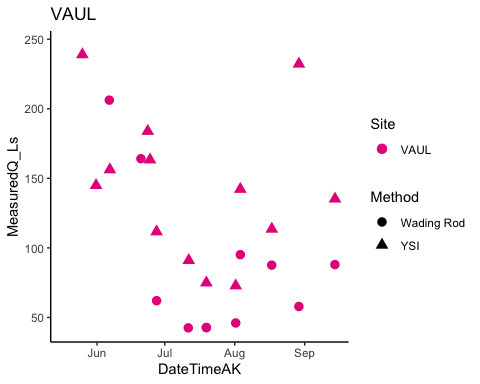
## Warning: Removed 5055 rows containing missing values (geom\_point).

 That is better. Crazy how fast the water level comes down at VAUL

# join the two atmospheric and water pressure together  
vaul.final.pressure.2022 <- vaul.stream.two.2022  
  
VAUL.2022 <- left\_join(eielson.atmo.2022.compare, vaul.final.pressure.2022, by = "DateTimeAK")  
  
# Water pressure - atmospheric pressure   
VAUL.2022$difference <- VAUL.2022$WaterPressure - VAUL.2022$AirPressureCorrectedVAUL  
  
VAUL.2022 <- VAUL.2022[ , -which(names(VAUL.2022) %in% c("TempC"))] # removing columns I do not need  
  
VAUL.2022$Site <- "VAUL"

### Filter VAUL ###  
QSummary.VA.2022 <- QSummary.2022 %>% filter(Site =="VAUL")  
  
ggplot(QSummary.VA.2022) +  
 geom\_point(aes(x = DateTimeAK, y = MeasuredQ\_Ls, shape = Method, color = Site), size=3) +  
 theme\_classic() +  
 scale\_color\_manual(values=c("#E7298A")) +   
 ggtitle("VAUL")

## Warning: Removed 2 rows containing missing values (geom\_point).



# trying to merge by nearest date if we have an offset point   
VAUL.2022.dt <- setDT(VAUL.2022)  
VAUL.2022.dt <- subset(VAUL.2022.dt, DateTimeAK < "2022-08-29 04:00:00") # removing rows that had dates corresponding to end of record that messed up the rolling nearest function   
QSummary.VA.2022.dt <- QSummary.VA.2022  
  
Vault1comb.2022 <- VAUL.2022.dt[QSummary.VA.2022.dt, on = "DateTimeAK", roll = 'nearest']  
  
VAUL1.lm.2022 <- lm(Vault1comb.2022$MeasuredQ\_Ls ~ Vault1comb.2022$difference)

# rating curve #   
vaul.formula <- y ~ x  
  
ggplot(aes(x = difference, y = MeasuredQ\_Ls), data = Vault1comb.2022) +  
 geom\_point(aes(color = Method), size = 3) +  
 geom\_smooth(method = "lm", se=FALSE, formula = vaul.formula) +  
 stat\_poly\_eq(formula = frch.formula,  
 aes(label = paste(..eq.label.., ..rr.label.., sep = "~~~")),  
 parse = TRUE) +  
 #xlim(216, 216.4) +  
 #ylim(0,1500) +  
 theme\_light() +  
 ggtitle("Vault all measured Q")

## Warning: Removed 17 rows containing non-finite values (stat\_smooth).

## Warning: Removed 17 rows containing non-finite values (stat\_poly\_eq).

## Warning: Removed 17 rows containing missing values (geom\_point).

