VA_Data_Comparisons

2024-10-15

This code examines the differences between environmental data for the VA site, located at Deep Water Shoal, James River, Virginia, as part of the CViMVP project. The two data sources are:

1. VIMS Water Quality Data, which were downloaded by Madeline Eppley on 15 September 2023
2. NOAA National Buoy Data Center (NDBC), Chesapeake Bay Interpretive Buoy System downloaded by myself (Nicole Mongillo) on 1 October 2024.

I will plot the salinity and temperature data from each data set against each other to see how similar the data are. I will then apply a correction factor to the NDBC data based on the difference between NDBC and VIMS data.

All labels for objects with data from NOAA will begin with NDBC, and all labels for objects with data from VIMS will be labeled VIMS.

```
setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
```

```
library("dplyr") #Used for working with data frames
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

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

```
library("lubridate") #Used for time-date conversions
```

```
##
## Attaching package: 'lubridate'

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

```
library("readr") #Used to read the CSV file
library("ggplot2")
library("stringr")
```

#VIMS Data Upload and Cleaning

```r
#Environmental data from the NDBC could only be downloaded by year, so first we need to merge the yearly

VIMS_raw <- read_csv("../../data/envr_of_origin/raw_envr_data/VA1-VIMS-raw.csv")
```

```
## Rows: 220 Columns: 7
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): DateDeployed, DateRetrieved
## dbl (5): StationID, Year, AverageSpat, WaterTemperature, Salinity
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
# View how the data are stored. Note the variable names and the format and units that the data are stor
summary(VIMS_raw)
```

```
##    StationID        Year      DateDeployed       DateRetrieved
##  Min.   :435   Min.   :2010   Length:220         Length:220
##  1st Qu.:435   1st Qu.:2013   Class :character   Class :character
##  Median :435   Median :2016   Mode  :character   Mode  :character
##  Mean   :435   Mean   :2016
##  3rd Qu.:435   3rd Qu.:2020
##  Max.   :435   Max.   :2023
##
##   AverageSpat     WaterTemperature    Salinity
##  Min.   : 0.000   Min.   :17.40    Min.   : 3.80
##  1st Qu.: 0.325   1st Qu.:25.15    1st Qu.:12.70
##  Median : 1.450   Median :26.55    Median :15.70
##  Mean   : 6.179   Mean   :26.21    Mean   :15.05
##  3rd Qu.: 5.400   3rd Qu.:27.80    3rd Qu.:18.00
##  Max.   :81.700   Max.   :29.90    Max.   :21.60
##  NA's   :22
```

```r
#Convert to POSIXct format. Store it into a column named datetime in the data frame.
VIMS_raw$datetime <- as.POSIXct(VIMS_raw$DateRetrieved, "%d-%b-%y", tz = "")

#Print the new data frame and examine to make sure the new datetime column is in the correct format.
head(VIMS_raw)
```

```
## # A tibble: 6 x 8
##   StationID  Year DateDeployed DateRetrieved AverageSpat WaterTemperature
##       <dbl> <dbl> <chr>        <chr>               <dbl>            <dbl>
## 1       435  2023 01-Jun-23    01-Jun-23            NA               19.6
## 2       435  2023 01-Jun-23    15-Jun-23             0               23
## 3       435  2023 15-Jun-23    22-Jun-23             1               23.1
## 4       435  2023 22-Jun-23    29-Jun-23             7.7             24.4
## 5       435  2023 29-Jun-23    06-Jul-23            14.2             28.1
## 6       435  2023 06-Jul-23    13-Jul-23            35.5             28.7
## # i 2 more variables: Salinity <dbl>, datetime <dttm>
```

```r
#rename columns
VIMS_raw <- VIMS_raw %>% rename("salinity_VIMS" = "Salinity")
VIMS_raw <- VIMS_raw  %>% rename("temp_VIMS" = "WaterTemperature")

#Filter the data between the values of 0 and 40 for both salinity and temperature.
VIMS_filtered <- VIMS_raw %>%
    filter(between(salinity_VIMS, 0, 40))

VIMS_filtered <- VIMS_raw %>%
    filter(between(temp_VIMS, 0, 40))

# Sanity check - print the ranges to ensure values are filtered properly. We can see that the ranges fo
print(summary(VIMS_filtered$salinity_VIMS))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    3.80   12.70   15.70   15.05   18.00   21.60
```

```r
print(summary(VIMS_filtered$temp_VIMS))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   17.40   25.15   26.55   26.21   27.80   29.90
```

```r
VIMS_filtered$date <- as.POSIXct(VIMS_filtered$datetime, "%Y-%m-%d", tz = "")
```

#NDBC Data Upload and Cleaning

```r
#Environmental data from the NDBC could only be downloaded by year, so first we need to merge the yearl

getwd()
```

```
## [1] "/Users/nicolemongillo/Desktop/GitHub/MVP-H2F-HatcheryField/src/envr_data"
```

```r
#set working directory to files location
setwd("../../data/envr_of_origin/raw_envr_data/VA_NDBC_Data_44041")

#merge files into one
NDBC_raw <- list.files(path=".") %>%
  lapply(read.csv) %>%
  bind_rows

NDBC_raw <- subset(NDBC_raw, select = c(X.YY, MM, DD, hh, mm, OTMP, SAL)) #keep year, month, day, hour,

# View how the data are stored. Note the variable names and the format and units that the data are stor
summary(NDBC_raw)
```

```
##       X.YY            MM               DD              hh
##  Min.   :2008   Min.   : 1.000   Min.   : 1.00   Min.   : 0.00
##  1st Qu.:2012   1st Qu.: 5.000   1st Qu.: 8.00   1st Qu.: 5.00
##  Median :2016   Median : 8.000   Median :16.00   Median :11.00
##  Mean   :2015   Mean   : 7.222   Mean   :16.01   Mean   :11.49
```

```
##  3rd Qu.:2019   3rd Qu.:10.000   3rd Qu.:24.00   3rd Qu.:18.00
##  Max.   :2019   Max.   :12.000   Max.   :31.00   Max.   :23.00
##  NA's   :350    NA's   :350      NA's   :350     NA's   :350
##        mm              OTMP            SAL
##  Min.   : 0.000   Min.   : 0.00   Min.   : 0.000
##  1st Qu.: 0.000   1st Qu.:11.80   1st Qu.: 0.400
##  Median : 0.000   Median :21.50   Median : 2.600
##  Mean   : 9.928   Mean   :19.97   Mean   : 3.268
##  3rd Qu.:18.000   3rd Qu.:27.60   3rd Qu.: 5.300
##  Max.   :54.000   Max.   :99.00   Max.   :99.000
##  NA's   :350      NA's   :350     NA's   :350
```

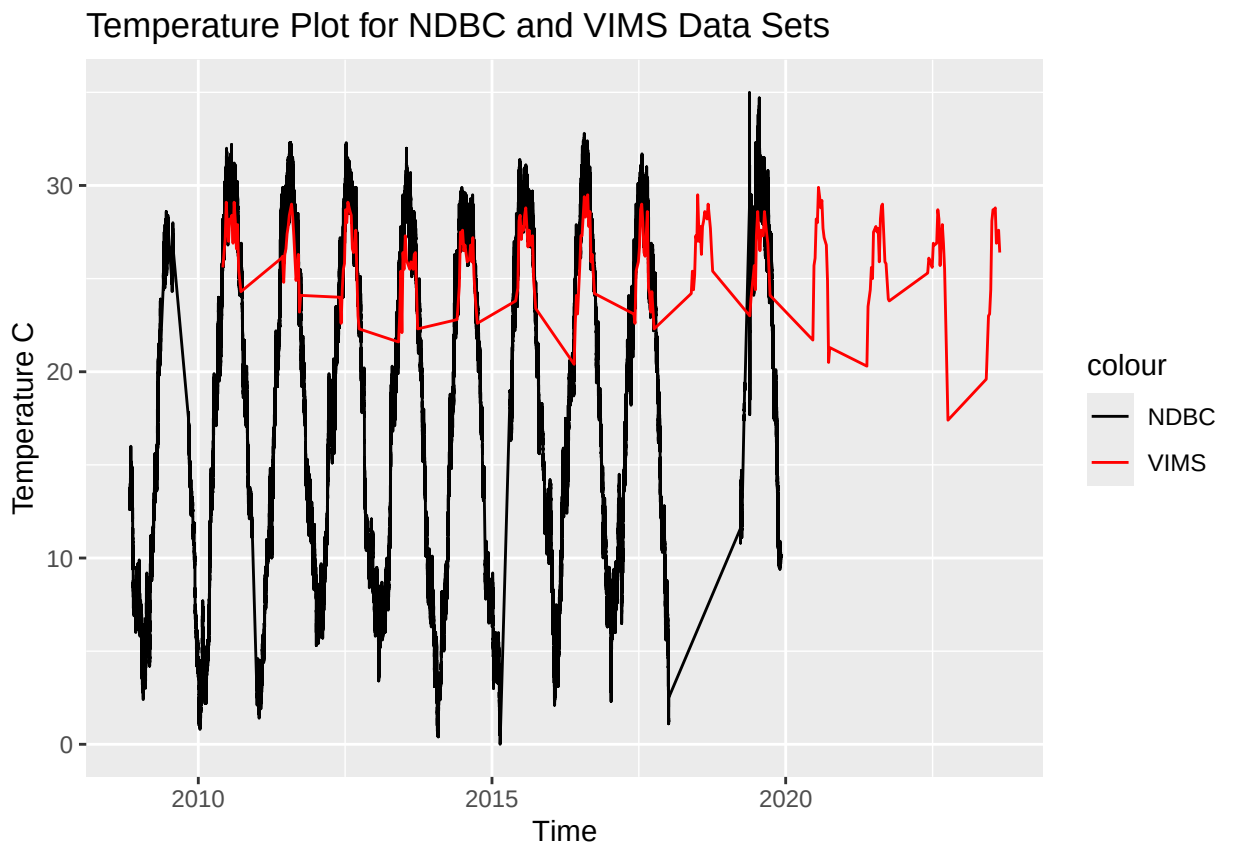
```
#make one single datetime column in POSIXct format
NDBC_raw$datetime <- as.POSIXct(paste(NDBC_raw$X.YY, NDBC_raw$MM, NDBC_raw$DD, NDBC_raw$hh, NDBC_raw$mm

#remove unmerged date-time columns
NDBC_raw<- subset(NDBC_raw, select = c(OTMP, SAL, datetime))

#reorder and rename columns
NDBC_raw <- NDBC_raw[ , c(3,1,2)]

colnames(NDBC_raw) <- c("datetime", "temp_NDBC", "salinity_NDBC")

#Filter the data between the values of 0 and 40 for both salinity and temperature.
NDBC_filtered1 <- NDBC_raw %>%
    filter(salinity_NDBC >=0 & salinity_NDBC <= 40)
# Sanity check - print the ranges to ensure values are filtered properly. We can see that the ranges fo
print(summary(NDBC_filtered1$salinity_NDBC))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.400   2.600   3.086   5.300  12.000
```

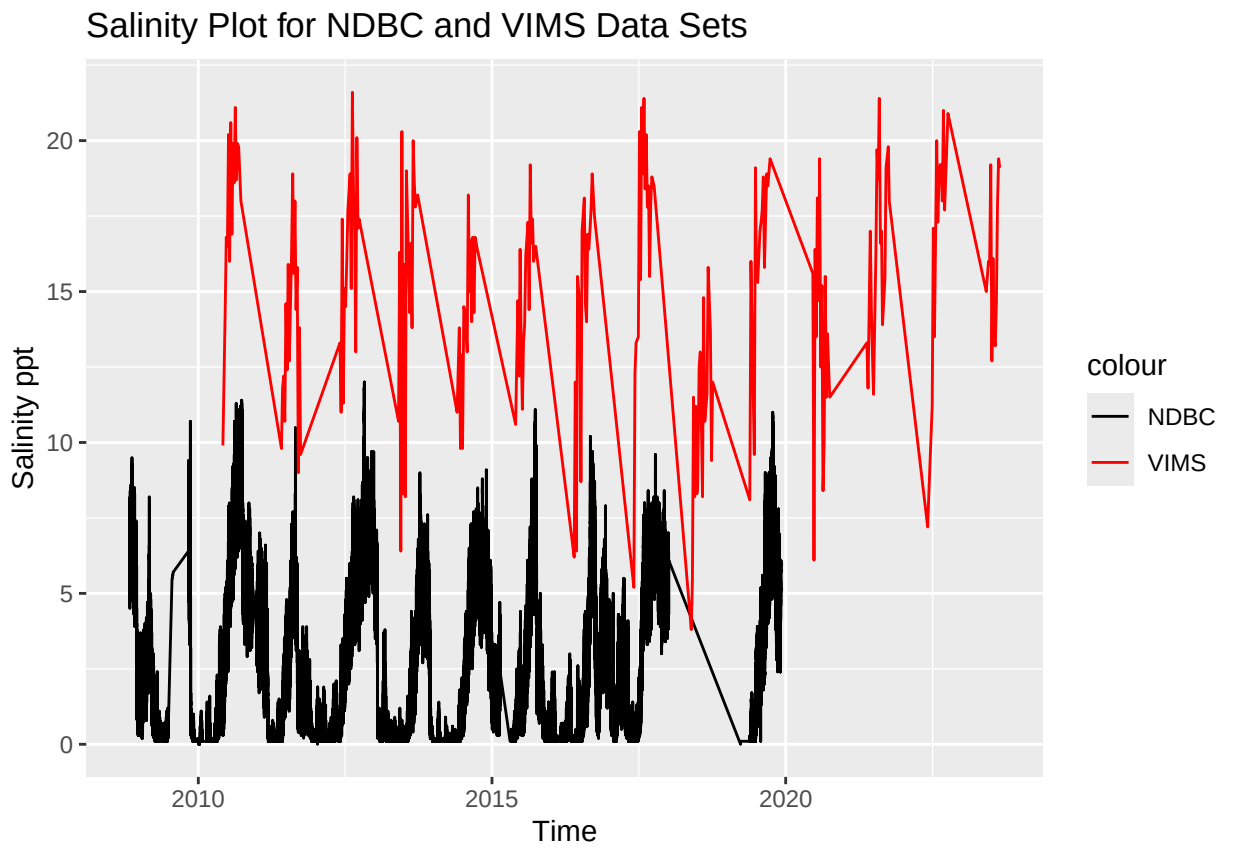
```
#Repeat for temperature
NDBC_filtered <- NDBC_filtered1 %>%
    filter(between(temp_NDBC, 0, 40))
print(summary(NDBC_filtered$temp_NDBC))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   11.80   21.50   19.84   27.60   35.00
```

```
tempplot <- ggplot(NDBC_filtered, aes(x=datetime, y = temp_NDBC, color = "NDBC")) +
    geom_line()+
    geom_line(data = VIMS_filtered, aes(x=datetime, y = temp_VIMS, color = "VIMS"))+
  scale_color_manual(values=c("black", "red"))+
  labs(x = "Time", y = "Temperature C", title = "Temperature Plot for NDBC and VIMS Data Sets")

tempplot
```




```
salplot <- ggplot(NDBC_filtered, aes(x=datetime, y = salinity_NDBC, color = "NDBC")) +
    geom_line()+
    geom_line(data = VIMS_filtered, aes(x=datetime, y = salinity_VIMS, color = "VIMS"))+
  scale_color_manual(values=c("black", "red"))+
  labs(x = "Time", y = "Salinity ppt", title = "Salinity Plot for NDBC and VIMS Data Sets")

salplot
```



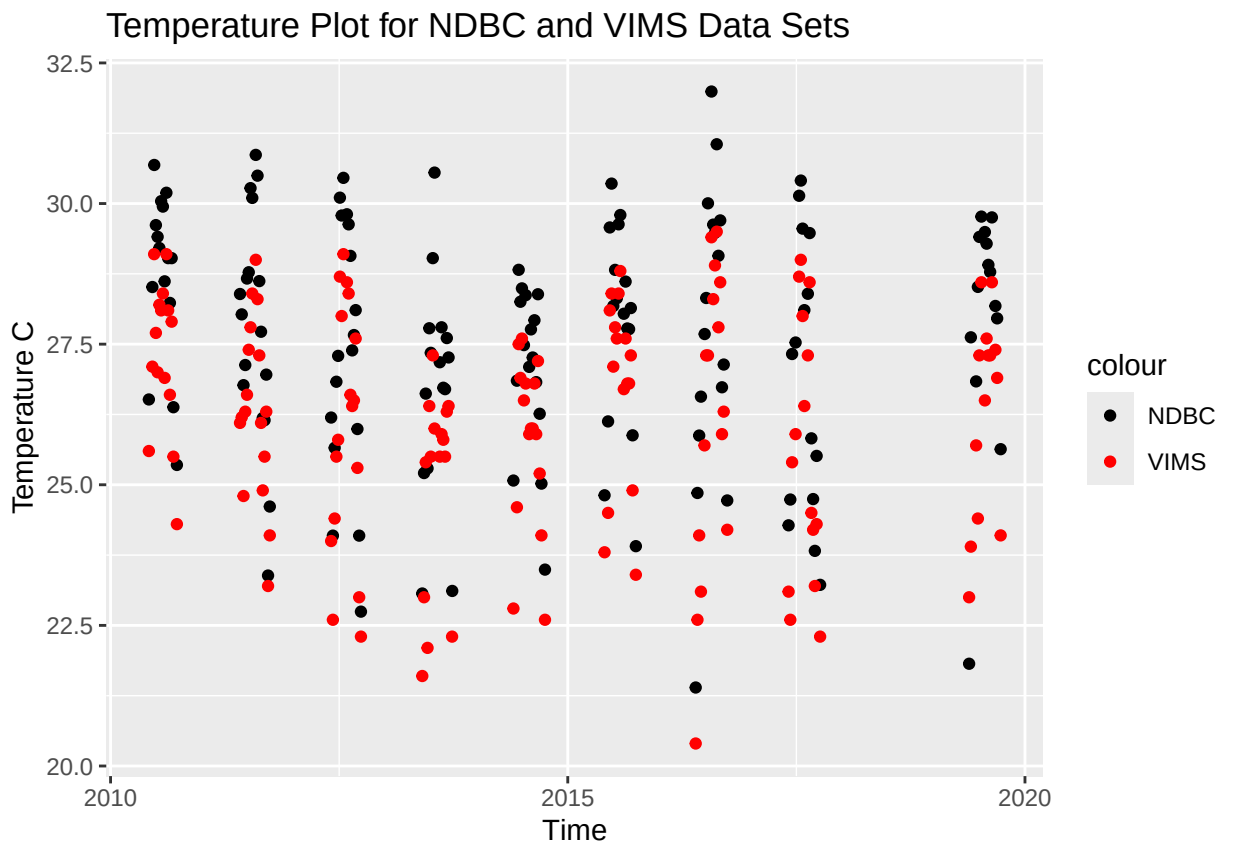
VIMS data are recorded daily, while NDBC are recorded once an hour. I will average NDBC temperature and salinity by day. Then I will select days that appear in both data sets in order to do a correction.

```
#extract date from datetime and make it POSIXct format
NDBC_filtered$date <- format(NDBC_filtered$datetime, "%Y-%m-%d")
NDBC_filtered$date <- as.POSIXct(NDBC_filtered$date, "%Y-%m-%d", tz = "")

#average temperature and salinity by day for NDBC
NDBC_filtered <- NDBC_filtered %>%
  group_by(date) %>%
  mutate(mean_daily_temp_NDBC = mean(temp_NDBC), mean_daily_sal_NDBC = mean(salinity_NDBC))

#filter so each day is represented once
NDBC_day <- NDBC_filtered[match(unique(NDBC_filtered$date), NDBC_filtered$date), ]

NDBC_day_filtered <- NDBC_day[NDBC_day$date %in% VIMS_filtered$datetime, ]

VIMS_day_filtered <- VIMS_filtered[VIMS_filtered$datetime %in% NDBC_day_filtered$date, ]
```

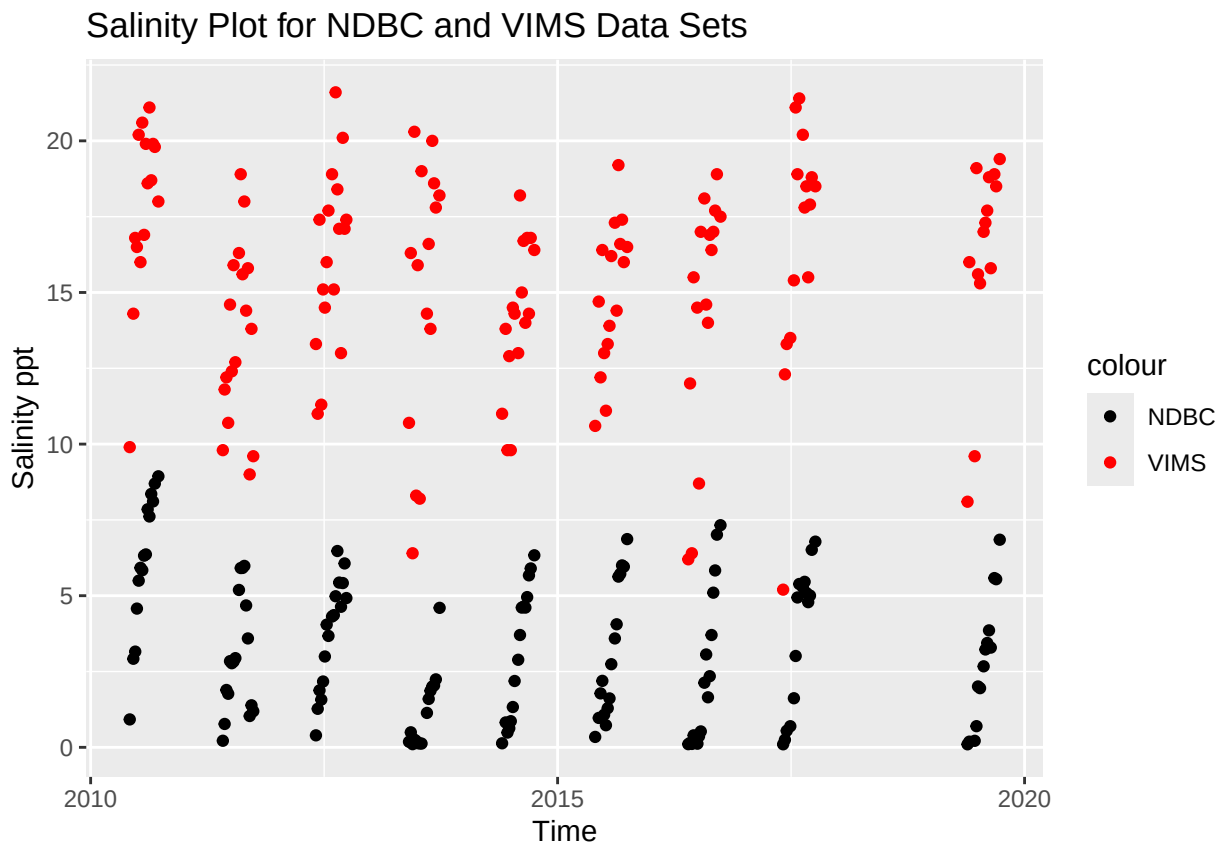
Re-plot temperature and salinity from the two data sets to make sure dates look like they align

```
filter_tempplot <- ggplot(NDBC_day_filtered, aes(x=datetime, y = mean_daily_temp_NDBC, color = "NDBC"))
    geom_point()+
    geom_point(data = VIMS_day_filtered, aes(x=date, y = temp_VIMS, color = "VIMS"))+
  scale_color_manual(values=c("black", "red"))+
  labs(x = "Time", y = "Temperature C", title = "Temperature Plot for NDBC and VIMS Data Sets")
```

```
filter_tempplot
```




```
filtered_salplot <- ggplot(NDBC_day_filtered, aes(x=datetime, y = mean_daily_sal_NDBC, color = "NDBC"))
    geom_point()+
    geom_point(data = VIMS_day_filtered, aes(x=date, y = salinity_VIMS, color = "VIMS"))+
  scale_color_manual(values=c("black", "red"))+
  labs(x = "Time", y = "Salinity ppt", title = "Salinity Plot for NDBC and VIMS Data Sets")

filtered_salplot
```




```
NDBC_VIMS_df <- left_join(VIMS_day_filtered,
          NDBC_day_filtered, by = "date")

#select relevant columns
NDBC_VIMS_df <- NDBC_VIMS_df[ , c("date", "mean_daily_temp_NDBC", "temp_VIMS", "mean_daily_sal_NDBC", "

NDBC_VIMS_df <- NDBC_VIMS_df %>%
  mutate(temp_diff = temp_VIMS-mean_daily_temp_NDBC, sal_diff = salinity_VIMS -mean_daily_sal_NDBC)

temp_correction <- mean(NDBC_VIMS_df$temp_diff)
#On average, VIMS temperature readings are 1.4 °C lower than readings from NDBC. Correct NDBC temperatu

sal_correction <- mean(NDBC_VIMS_df$sal_diff)
#On average, VIMS salinity readings are 12.1 ppt higher than readings from NDBC. Correct NDBC salinity

NDBC_raw$temp_corrected <- NDBC_raw$temp_NDBC-1.4

NDBC_raw$salinity_corrected <- NDBC_raw$salinity_NDBC+12.1

write.csv(NDBC_raw, "/Users/nicolemongillo/Desktop/GitHub/MVP_Chesapeake_VIMS_hatchery/data/envr_raw_da
```