

COMPASS_SynopticCB_PW_SO4_Cl_202311_Template

Stephanie J. Wilson

2025-08-29

Contents

0.1	Import Sample Data	4
0.2	Assessing Standard Curves	4
0.3	Assess Check Standards	6
0.4	Assess Blanks	7
0.5	Assess Duplicates	8
0.6	Calculate mmol/L concentrations & salinity, add dilutions	10
0.7	Assess Analytical Spikes	11
0.8	Check if samples within the range of the standard curve	11
0.9	Check to see if samples run match metadata & merge info	12
0.10	Visualize Data by Plot	12
0.11	Export Processed Data	13

```
##Add Required Packages
##Keep the R Markdown output within the PDF margins
##Setup - Change things here & write any notes
```

```
# identify section
cat("Setup Information")
```

```
## Setup Information
```

```
##### Run information - PLEASE CHANGE
Date_Run = "2023-09-05" #Date that instrument was run
Run_by = "Unknown" #Instrument user
Script_run_by = "Zoe Read" #Code user
run_notes = "Samples missing from metadata:
  MSM_202311_UP_LYSC_45CM
  MSM_202311_WC_LYSB_45CM
  MSM_202311_WC_LYSC_45CM
  MSM_202311_UP_RHZ_SF_TREE_1
  MSM_202311_UP_RHZ_SF_TREE_2
  MSM_202311_UP_RHZ_SF_TREE_3
  MSM_202311_UP_RHZ_SF_TREE_4
  MSM_202311_UP_RHZ_SF_TREE_6
  MSM_202311_UP_RHZ_SF_TREE_7
  MSM_202311_UP_RHZ_SF_TREE_8
  MSM_202311_TR_RHZ_SF_TREE_
  MSM_202311_TR_RHZ_SF_TREE_2
  MSM_202311_TR_RHZ_SF_TREE_3
  MSM_202311_TR_RHZ_SF_TREE_4
  MSM_202311_TR_RHZ_SF_TREE_5
  MSM_202311_TR_RHZ_SF_TREE_6
  MSM_202311_TR_RHZ_SF_TREE_7
  MSM_202311_TR_RHZ_SF_TREE_8
  MSM_202311_WC_RHZ_SF_COLLAR_1
  MSM_202311_WC_RHZ_SF_COLLAR_2
  MSM_202311_WC_RHZ_SF_COLLAR_3
  MSM_202311_WC_RHZ_SF_COLLAR_4
  MSM_202311_WC_RHZ_SF_COLLAR_5
  MSM_202311_WC_RHZ_LYSA
  MSM_202311_WC_RHZ_LYSC.
  118_MSM_202311_TR_RHZ_SF_Tree_1 value above cal curve for S04, but only slightly."
# any notes from the run

samples <- c("GCW", "GWI", "MSM", "SWH") #whatever identifies your samples within the same names
samples_pattern <- paste(samples, collapse = "|")
# samples_pattern <- 'GCW' #use this instead of the line
# above if you have only one site code
chks_name = "Check Standard" #what did you name your check standards?

##### File Names - PLEASE CHANGE file path and name for
##### raw summary data file raw_file_name_cl =
##### 'Porewater/Sulfate_Chloride/Synoptic_CB/2023/Raw
##### Data/COMPASS_Synoptic_CB_MonMon_202311_C1.txt'
```

```
##### raw_file_name_so4 =
##### 'Porewater/Sulfate_Chloride/Synoptic_CB/2023/Raw
##### Data/COMPASS_Synoptic_CB_MonMon_202311_SO4.txt'

raw_file_name_cl = "Raw Data/COMPASS_Synoptic_CB_MonMon_202311_Cl.txt"
raw_file_name_so4 = "Raw Data/COMPASS_Synoptic_CB_MonMon_202311_SO4.txt"

# file path and name of processed data file
# processed_file_name =
# 'Porewater/Sulfate_Chloride/Synoptic_CB/2023/Processed
# Data/COMPASS_SynopticCB_PW_Processed_Cl_SO4_202311.csv'
processed_file_name = "Processed Data/COMPASS_SynopticCB_PW_Processed_Cl_SO4_202311.csv"

##### Log Files - PLEASE CHECK downloaded metadata csv -
##### downloaded from Google drive as csv for this year
##### Raw_Metadata =
##### 'Porewater/Sulfate_Chloride/Synoptic_CB/2023/Raw
##### Data/COMPASS_SynopticCB_PW_SampleLog_2023.csv'
Raw_Metadata = "Raw Data/COMPASS_SynopticCB_PW_SampleLog_2023.csv"

# qaqc log file path for this year Log_path =
# 'Porewater/Sulfate_Chloride/Synoptic_CB/2023/Raw
# Data/COMPASS_Synoptic_Cl_SO4_QAQClog_2023.csv'
Log_path = "Raw Data/COMPASS_Synoptic_Cl_SO4_QAQClog_2023.csv"
```

##Set Up Code

```
#Link to the protocol used for analysis
#steph will add this soon

#Coefficients / constants that are needed for calculations
cl_mw <- 35.45      #molecular weight of Chloride, g/mol
s_mw <- 32.06       #molecular weight of sulfur, g/mol
Con1 <- 1000000     #conversion factor value for spike volumes (uL -> L)

#Flag cutoffs
r2_cutoff = 0.98    #this is the level below which we want to rerun or consider a curve
chk_flag_std_s = 10  #this is the maximum cv allowed for sulfate check standards
chk_flag_std_cl = 5  #this is the maximum cv allowed for chloride check standards
chk_flag_std_perc = 15 #this is the maximum perc diff allowed for check standards
chk_flag_dups = 10   #this is the maximum cv allowed for duplicates
high_recovery_cutoff = 120 #this is the maximum percent recovery of SO4 allowed in spiked samples
low_recovery_cutoff = 80  #this is the minimum percent recovery of SO4 allowed in spiked samples
chks_flag = 0.80      #if less than this percent of samples pass a check, a flag is added

#Standard concentrations - Update if running different standard curve:
standards <- tibble(
  sample_ID = c("Standard 1", "Standard 2", "Standard 3", "Standard 4", "Standard 5"),
  SO4_std_conc = c(0.5, 1.0, 2.0, 10, 20), #ug/mL
  Cl_std_conc = c(5, 10, 20, 100, 200))    #ug/mL

#Spike concentration calc
#spike for these samples was 10uL of the 250 ug/mL standard
```

```

spk_std <- (250/s_mw)      # mM of SO4 calculated from 250 ug/mL SO4 spike solution
spkvol <- 10               # uL volume of spike added
spkvol <- spkvol/Con1      # L volume of spike added
spk_Conc <- (spk_std)*spkvol # mmoles of SO4 added to each spiked sample

#Top standard Concentrations- Update if running different standard curve:
top_std_cl = 200          #ug/mL
top_std_so4 = 20          #ug/mL

#Set time zone
common_tz = "Etc/GMT+5"
Sys.setenv(TZ = "America/New_York")

```

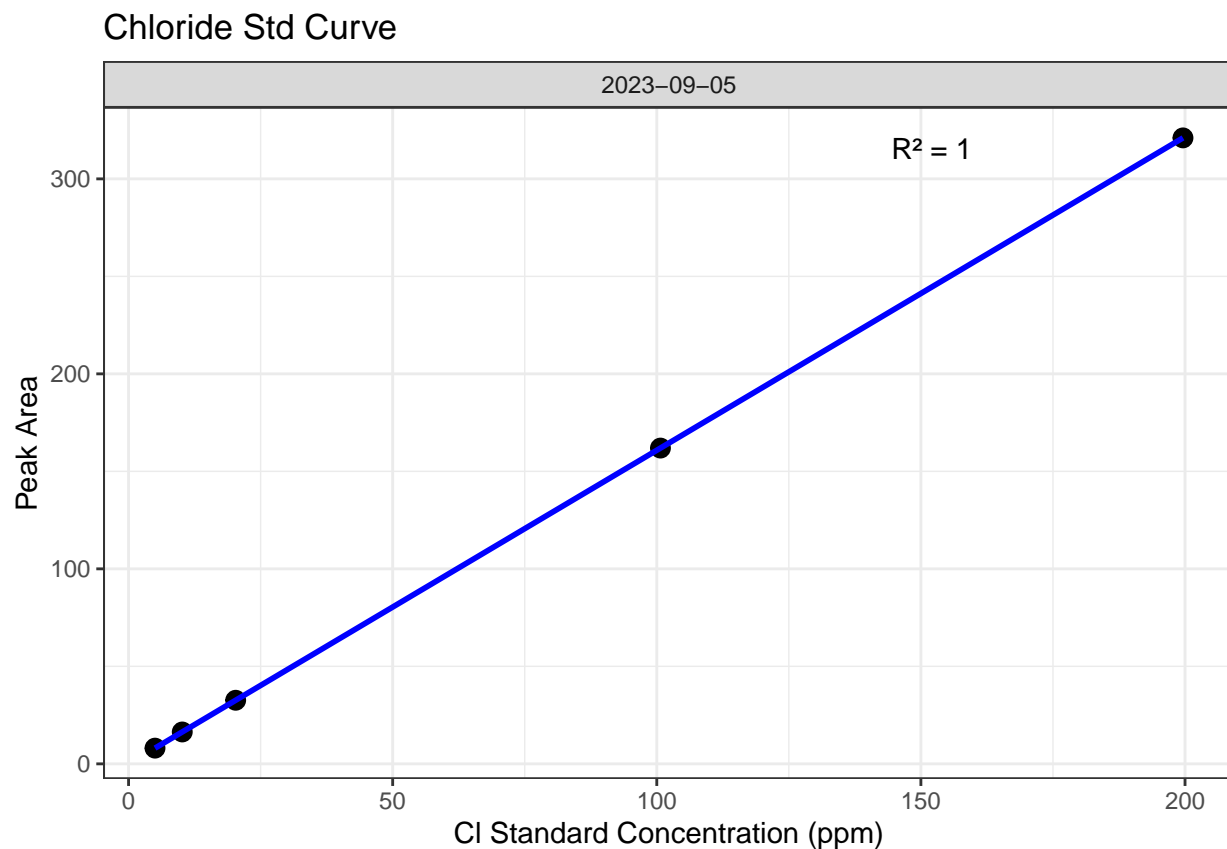
```
##Read in metadata and create similar sample IDs for matching to samples
```

0.1 Import Sample Data

0.2 Assessing Standard Curves

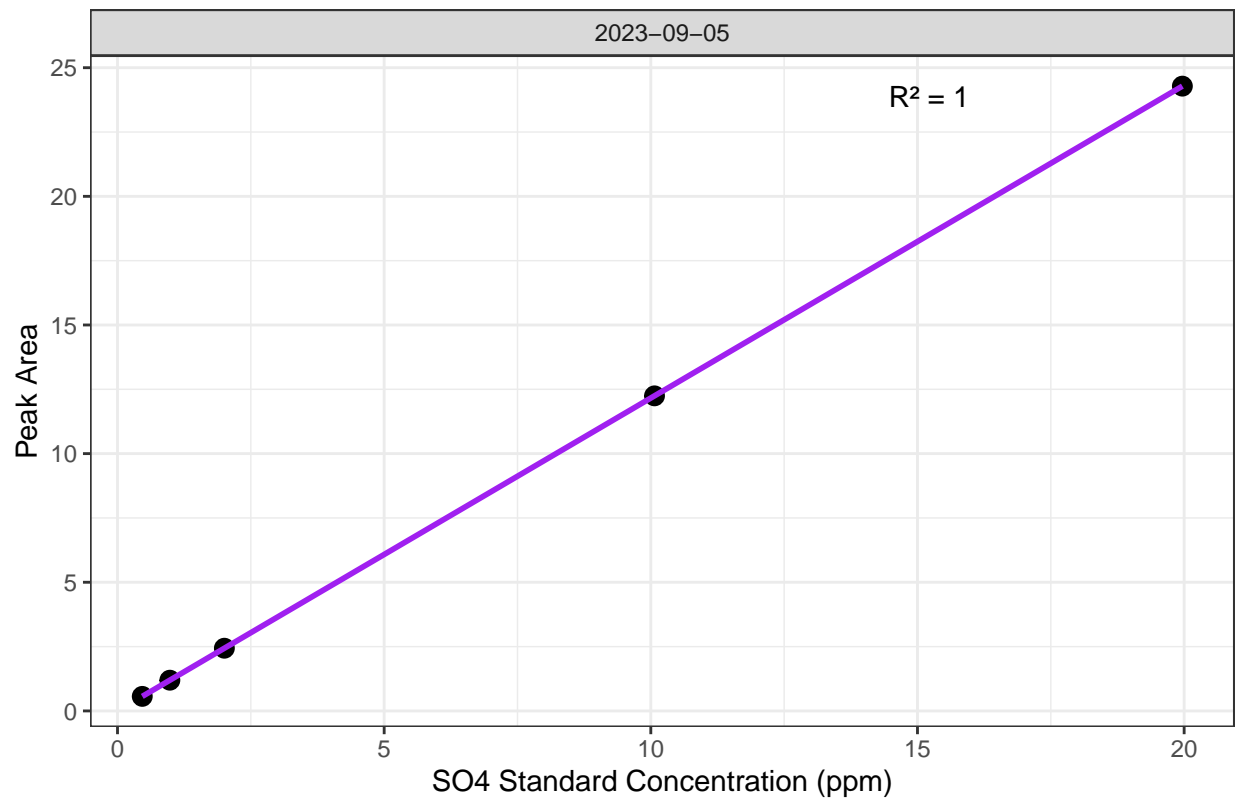
```
## Assess the Standard Curves
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```



```
## 'geom_smooth()' using formula = 'y ~ x'
```

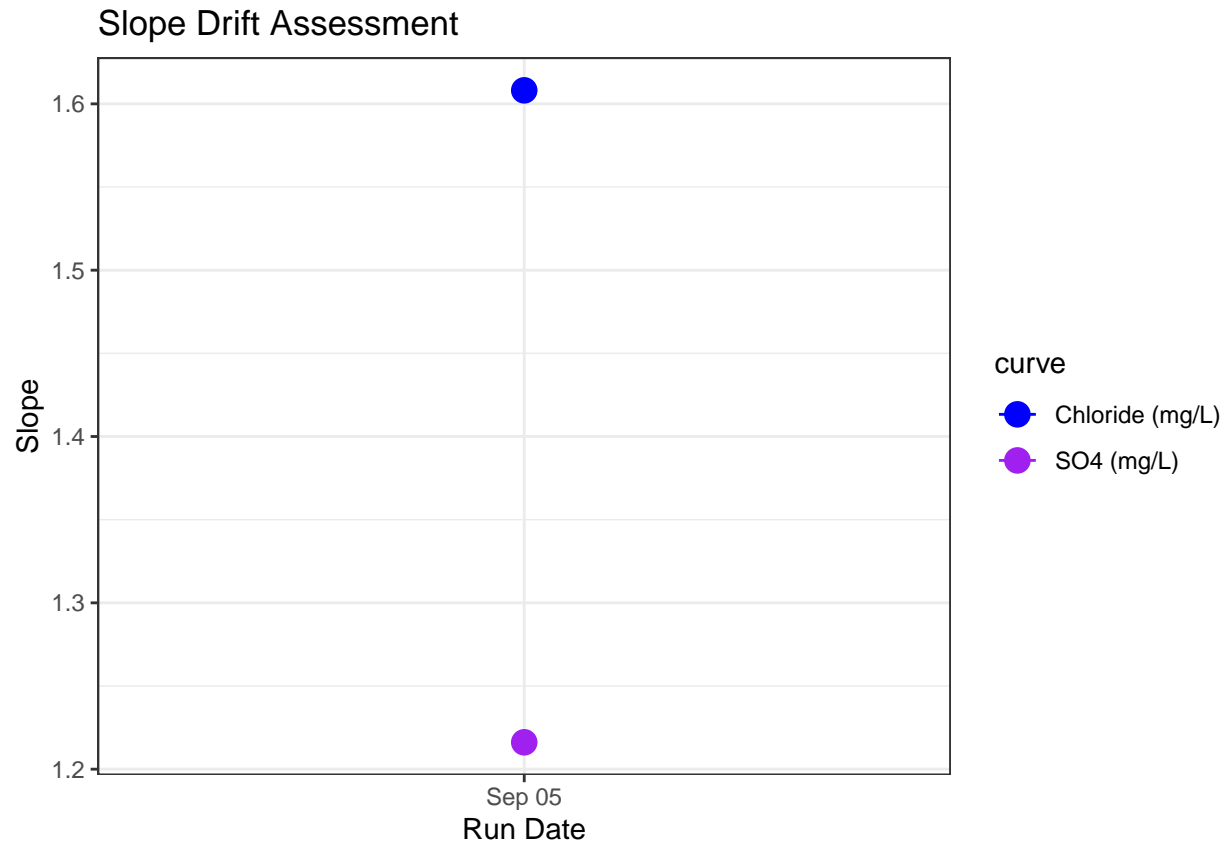
Sulfate Std Curve



```
## [1] "QAQC log file exists and has been read into the code."
```

```
## 'geom_line()': Each group consists of only one observation.
```

```
## i Do you need to adjust the group aesthetic?
```

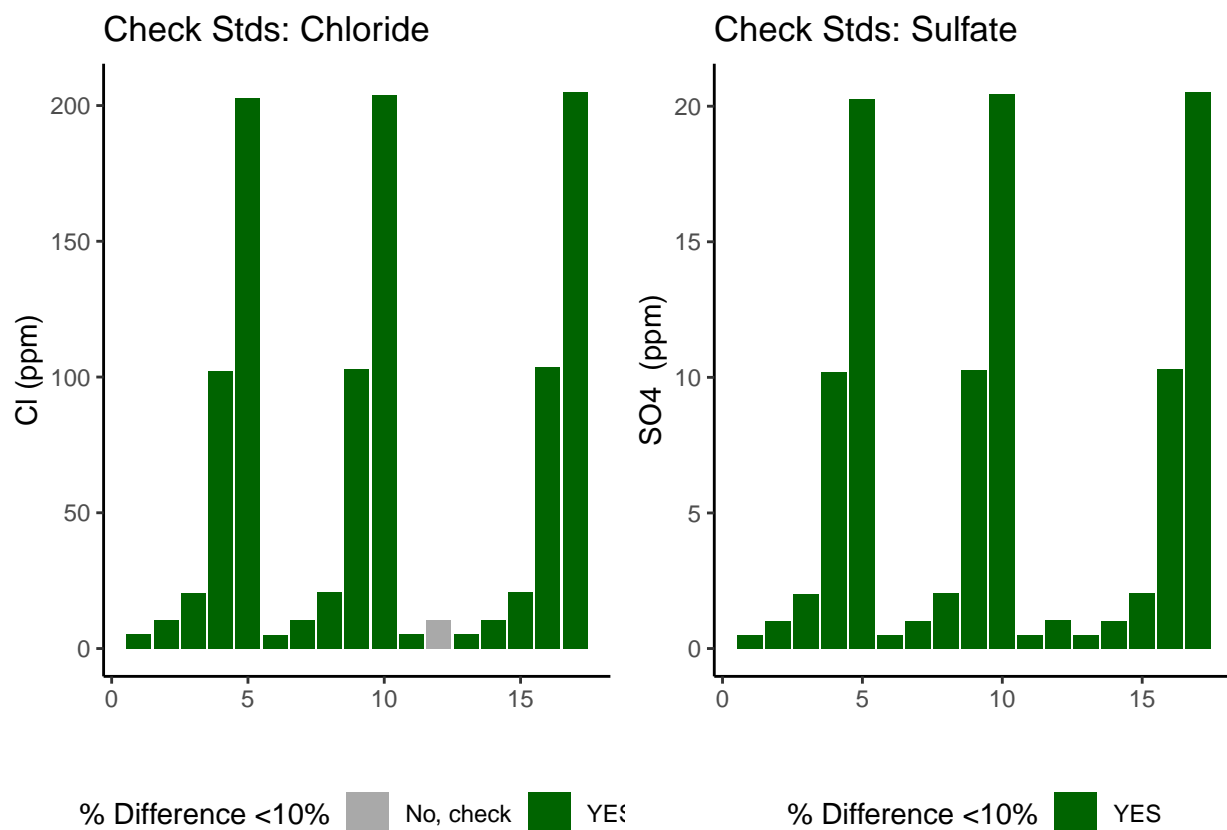


```
## [1] "Cl Curve r2 GOOD"
```

```
## [1] "SO4 Curve r2 GOOD"
```

0.3 Assess Check Standards

```
## Assess the Check Standards
```



```
## [1] ">80% of Chloride Check Standards are within range of expected concentration"
```

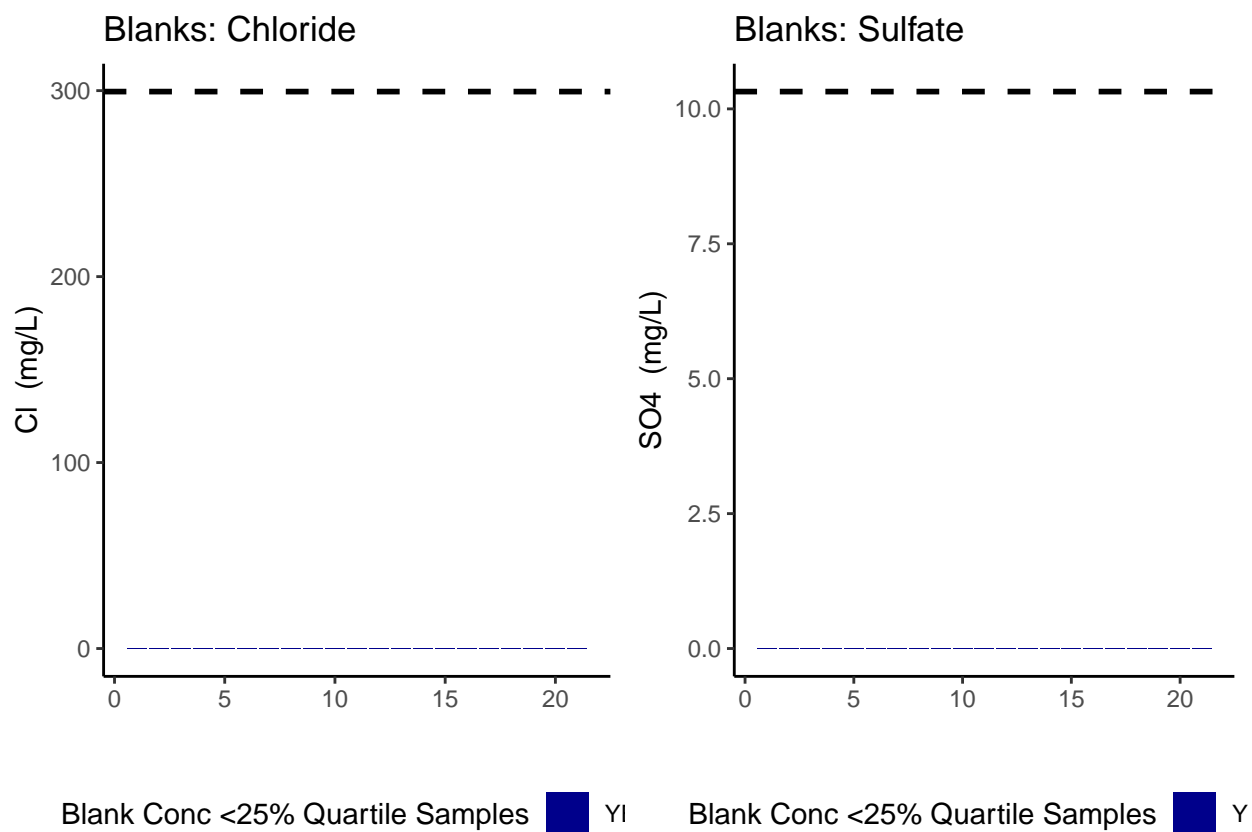
```
## [1] ">80% of Sulfate Check Standards are within range of expected concentration"
```

0.4 Assess Blanks

```
## Assess Blanks
```

```
## [1] ">80% of Chloride Blank concentrations are lower 25% quartile of samples"
```

```
## [1] ">80% of Sulfate Blank concentrations are lower 25% quartile of samples"
```



```
## Chloride blanks mean ppm:
```

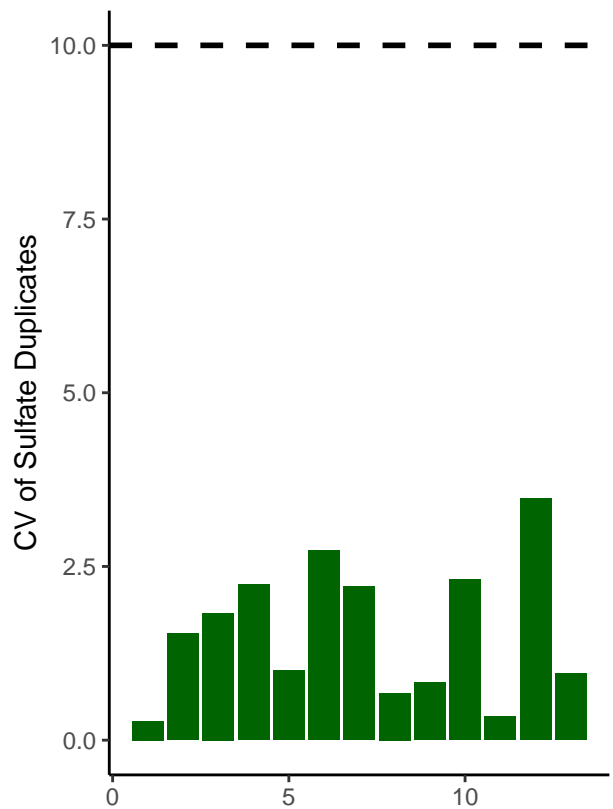
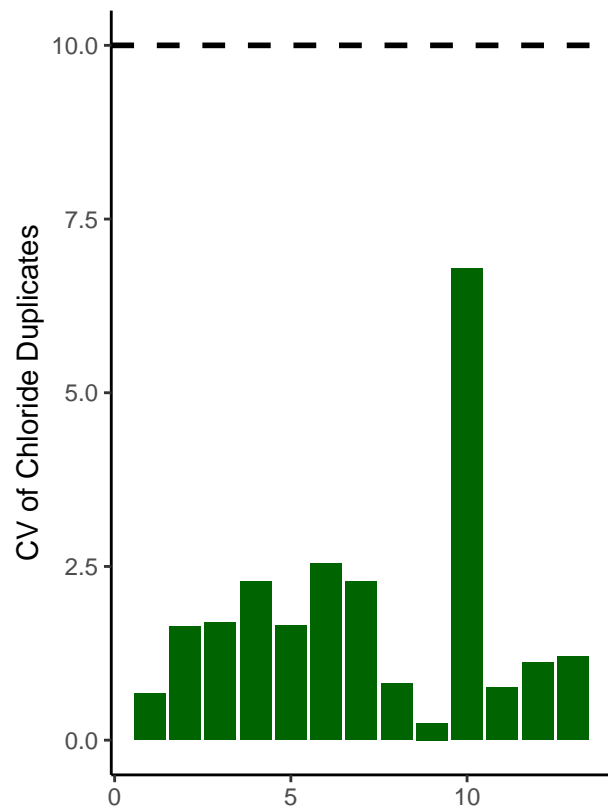
```
## [1] 0.01699524
```

```
## Sulfate blanks mean ppm:
```

```
## [1] 0.004957143
```

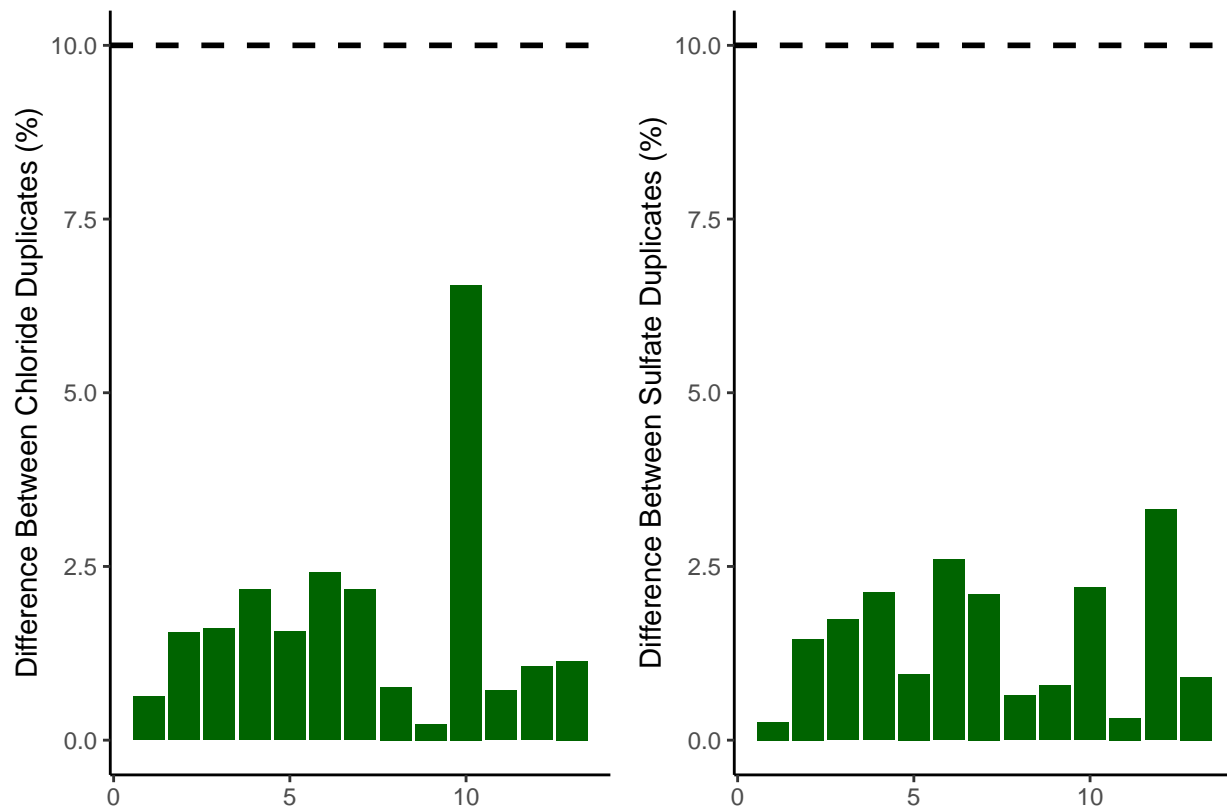
0.5 Assess Duplicates

```
## Assess Duplicates
```

```
## [1] ">80% of Chloride Duplicates have a CV <10%"
```

```
## [1] ">80% of Sulfate Duplicates have a CV <10%"
```



```
## [1] ">80% of Chloride Duplicates have a percent difference <10%"
```

```
## [1] ">80% of Sulfate Duplicates have a percent difference <10%"
```

0.6 Calculate mmol/L concentrations & salinity, add dilutions

```
cat("Unit Conversion and Salinity Calculation")

# Convert ppm to mmol/L
all_dat$SO4_mM <- (all_dat$SO4_ppm/s_mw)
all_dat$Cl_mM <- (all_dat$Cl_ppm/cl_mw)

# Calculate Salinity calculated using the Knudsen equation
# Salinity = 0.03 + 1.8050 * Chlorinity Ref: A Practical
# Handbook of Seawater Analysis by Strickland & Parsons (P.
# 11) =((1.807*Cl_ppm)+0.026)/1000
all_dat$salinity <- ((1.807 * all_dat$Cl_ppm) + 0.026)/1000

# Need to determine dilution factors for your samples for
# Steph / COMPASS this depends on the site so...
all_dat$Dilution <- 1
all_dat$Dilution <- ifelse(str_detect(all_dat$sample_ID, "MSM") &
  str_detect(all_dat$sample_ID, "UP"), 50, all_dat$Dilution)
```

```

all_dat$Dilution <- ifelse(str_detect(all_dat$sample_ID, "MSM") &
  str_detect(all_dat$sample_ID, "TR"), 50, all_dat$Dilution)
all_dat$Dilution <- ifelse(str_detect(all_dat$sample_ID, "MSM") &
  str_detect(all_dat$sample_ID, "WC"), 100, all_dat$Dilution)
all_dat$Dilution <- ifelse(str_detect(all_dat$sample_ID, "MSM") &
  str_detect(all_dat$sample_ID, "SW"), 100, all_dat$Dilution)

all_dat$Dilution <- ifelse(str_detect(all_dat$sample_ID, "GCW") &
  str_detect(all_dat$sample_ID, "UP"), 50, all_dat$Dilution)
all_dat$Dilution <- ifelse(str_detect(all_dat$sample_ID, "GCW") &
  str_detect(all_dat$sample_ID, "TR"), 50, all_dat$Dilution)
all_dat$Dilution <- ifelse(str_detect(all_dat$sample_ID, "GCW") &
  str_detect(all_dat$sample_ID, "WC"), 100, all_dat$Dilution)
all_dat$Dilution <- ifelse(str_detect(all_dat$sample_ID, "GCW") &
  str_detect(all_dat$sample_ID, "SW"), 100, all_dat$Dilution)

all_dat$Dilution <- ifelse(str_detect(all_dat$sample_ID, "GWI") &
  str_detect(all_dat$sample_ID, "UP"), 100, all_dat$Dilution)
all_dat$Dilution <- ifelse(str_detect(all_dat$sample_ID, "GWI") &
  str_detect(all_dat$sample_ID, "TR"), 100, all_dat$Dilution)
all_dat$Dilution <- ifelse(str_detect(all_dat$sample_ID, "GWI") &
  str_detect(all_dat$sample_ID, "WC"), 200, all_dat$Dilution)
all_dat$Dilution <- ifelse(str_detect(all_dat$sample_ID, "GWI") &
  str_detect(all_dat$sample_ID, "SW"), 200, all_dat$Dilution)

all_dat$Dilution <- ifelse(str_detect(all_dat$sample_ID, "SWH"),
  50, all_dat$Dilution)

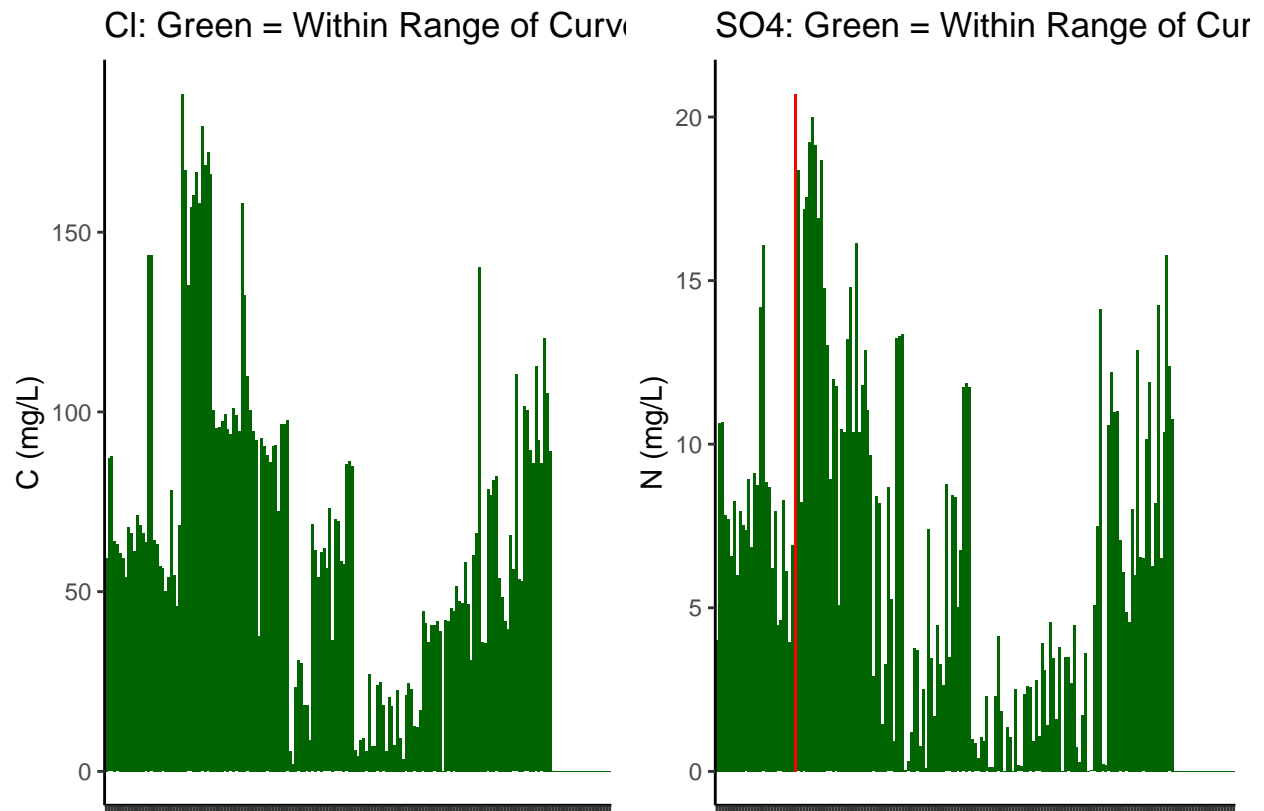
head(all_dat)

```

0.7 Assess Analytical Spikes

0.8 Check if samples within the range of the standard curve

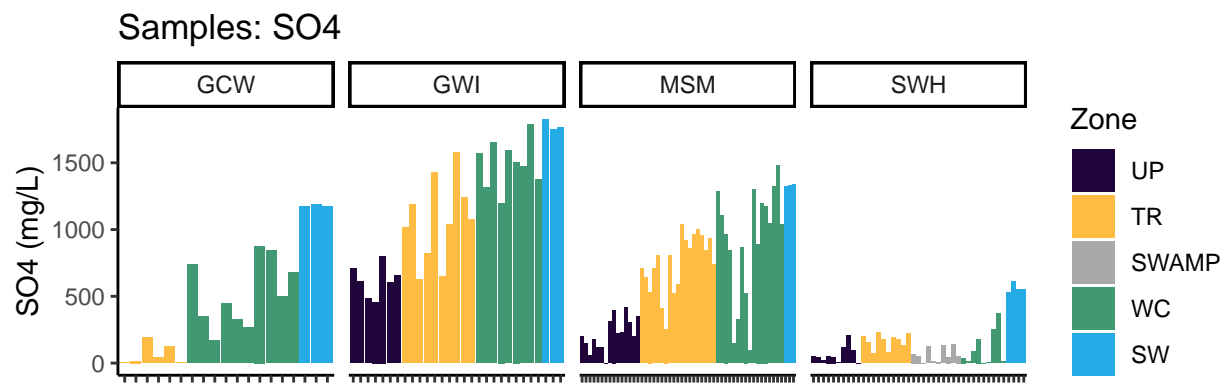
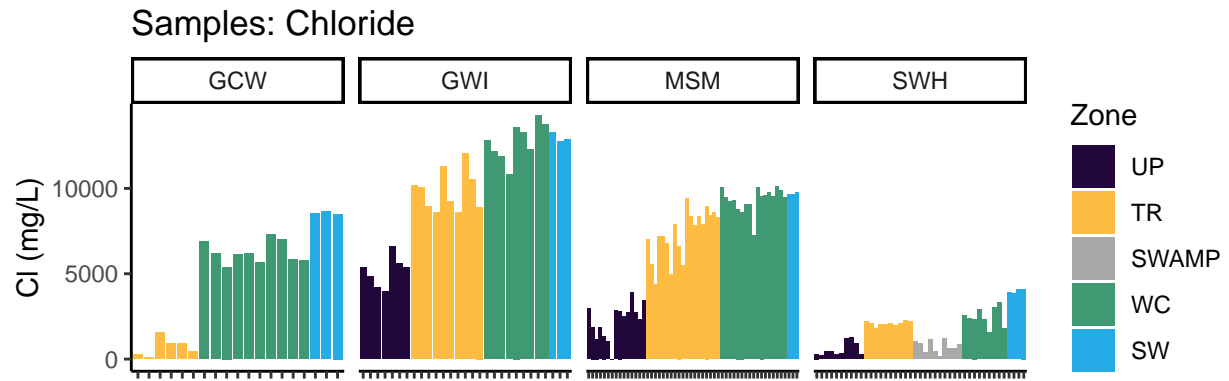
```
## Sample Flagging
```



0.9 Check to see if samples run match metadata & merge info

0.10 Visualize Data by Plot

```
## Visualize Data
```



0.11 Export Processed Data

#end